Mean-Field Last-Passage Percolation: Continuous Sparse Regime*

Scholars: Charles Cooper, Jules Goduco-Bunting, Jiya Patidar, Srujan Roplekar
Project Leader: Daecheol Kim
Faculty Mentor: Partha S. Dey

April 17, 2026

Contents

*Support for this project was provided by the Illinois Mathematics Lab and the Department of Mathematics at the University of Illinois Urbana–Champaign.

1 Introduction

Last Passage Percolation (LPP) is a classical probabilistic model originating from statistical physics, which has gained significant attention due to its deep connections with growth processes, queuing theory, and combinatorial optimization. The fundamental setup of LPP assigns independent and identically distributed (i.i.d.) random weights to edges in a given graph, with the passage time of any path defined as the sum of weights along its edges. The central objective of LPP is to determine the last passage time (LPT)—the maximum total weight achievable among all self-avoiding paths connecting two specified vertices. Historically, the majority of rigorous analysis has concentrated on lattice-based models, particularly in two dimensions, where explicit scaling limits and universal fluctuations such as the Tracy–Widom distribution have been extensively studied and characterized.

In contrast, Mean-Field Last Passage Percolation (MF LPP) shifts the focus from spatially structured lattices to highly symmetric, non-spatial structures, such as complete graphs or Erdős–Rényi random graphs. This transition introduces significant complexity, transforming a traditionally local optimization problem into a global one. Indeed, finding the longest weighted path in complete graphs is a notoriously difficult computational problem, classified as NP-hard. Despite these computational difficulties, MF LPP provides valuable theoretical insights and practical applications in network optimization, stochastic growth processes, and probabilistic combinatorics.

Previously, our work on MF LPP concentrated on the binary sparse regime, where edge weights were Bernoulli-distributed random variables, simplifying the analysis by effectively treating edges as either "present" or "absent." While the binary setting allowed for initial theoretical advances and algorithmic simplifications, it lacked the flexibility to model a wider variety of practical scenarios where edge weights naturally arise from continuous distributions. To overcome this limitation and enhance realism and applicability, the current paper transitions into the continuous sparse regime. Specifically, we investigate MF LPP on Erdős–Rényi random graphs $G(n, \frac{\lambda}{n})$ with edge weights drawn from continuous distributions such as Uniform$(0, 1)$ and Exponential(1).

The motivation for considering continuous distributions stems from real-world network optimization problems, where link strengths, transmission times, and costs are inherently continuous. By introducing continuous edge weights, we can capture richer probabilistic behaviors and complexities not observable in purely binary models. Our analysis relies on recursive distributional equations (RDEs) associated with Galton–Watson branching trees [2], which arise naturally in the sparse Erdős–Rényi setting as local approximations. Through careful study of these RDEs, we provide a rigorous theoretical framework to analyze both subcritical and supercritical regimes, detailing the asymptotic properties of optimal paths and passage times.

2 Model

The primary goal is to analyze the asymptotic behavior of the last passage time W_n on $\mathcal{K}_n$.

Definition 2.1.

- Self-Avoiding Walk (SAW): *A path $\pi = \langle x_0, x_1, \ldots, x_k \rangle$ is called a self-avoiding walk if the vertices $x_0, x_1, \ldots, x_k$ are all distinct.*
- Passage Time: *The passage time of a path π is defined as*

$$\mathcal{W}_\pi := \sum_{e \in \pi} \omega_e,$$

where ω_e is the weight of the edge $e = \langle x_i, x_{i+1} \rangle$ in the path π.

- Last Passage Time (LPT): *The last passage time W_n is defined as*

$$W_n := \sup_{\pi} \mathcal{W}_\pi,$$

where the supremum is taken over all self-avoiding walks π from vertex 1 to vertex n in $\mathcal{K}_n$.

It is well known that the problem of computing LPT is NP hard on complete graphs. Essentially, on a complete graph of n vertices, we are optimizing over $n!$ self-avoiding walks, so to check all SAWs will take factorial time.

Definition 2.2. ***Maximal-weight path cover (MWPC)****: A maximal-weight path cover Π is the optimizer of the Last Passage Time W_n on $\mathcal{K}_n$, that is, $\mathcal{W}_\Pi = W_n$, where Π is a collection of SAWs that partitions the vertices of $\mathcal{K}_n$.*

2.1 A Sparsified Relaxation.

To relax the original problem, we consider the following problem. Let $\lambda > 0$ be a parameter. We choose a threshold $\vartheta_n := \vartheta_n(\lambda)$ such that

$$\mathbb{P}(\omega \geqslant \vartheta_n(\lambda)) = \frac{\lambda}{n}, \text{ or equivalently, } \vartheta_n(\lambda) = \bar{F}^{-1}\left(\frac{\lambda}{n}\right),$$

where $\bar{F}(x) = \mathbb{P}(\omega > x)$ is the tail of ω.

Filtering only the edges with weights greater than $\vartheta_n(\lambda)$, transforms the complete graph with i.i.d. edge weights ω into an Erdős–Rényi random graph $G\left(n, \frac{\lambda}{n}\right)$. The retained edge weights become biased, as they are conditioned to exceed the threshold $\vartheta_n(\lambda)$. However, we can rescale it properly, and for simplicity, we instead consider following modified model:

Last Passage Percolation (LPP) on complete graph $\mathcal{K}_n$ with edge weights $\mathrm{Ber}\,(\lambda/n) \cdot \omega$

Equivalently, this is the *LPP on the Erdős–Rényi random graph $G\left(n, \frac{\lambda}{n}\right)$ with i.i.d. weights ω.*

Example 2.3 (Uniform weights)**.** *Assume that the edge weight distribution $\omega \sim \mathrm{Unif}(0,1)$. Then the threshold is given by $\vartheta(n) = 1 - \frac{\lambda}{n}$, since*

$$\mathbb{P}\left(\mathrm{Unif}(0,1) \geqslant 1 - \frac{\lambda}{n}\right) = \frac{\lambda}{n}.$$

Thus, the edge weights on the complete graph become

$$\mathrm{Ber}\left(\frac{\lambda}{n}\right) \cdot \mathrm{Unif}\left(1 - \frac{\lambda}{n}, 1\right).$$

Equivalently, we analyze the last passage time on the Erdős–Rényi $G\left(n, \frac{\lambda}{n}\right)$ with i.i.d. edge weights distributed as $\mathrm{Unif}\left(1 - \frac{\lambda}{n}, 1\right)$*. For simplicity, we rescale the weights so that they are simply distributed as* $\mathrm{Unif}(0,1)$.

3 Algorithm on Finite Galton-Watson Trees

In Erdős–Rényi random graphs, many components exhibit tree structures due to their sparse connectivity. Analyzing these tree components is crucial for understanding the global structure of the graph and addressing associated optimization problems. In this section, we introduce two algorithms designed for finite trees to identify the maximum total edge weight of a vertex-disjoint path cover, ensuring each vertex appears in at most two edges.

Let $\mathcal{T} = \mathcal{T}_\lambda$ denote a Poisson(λ) branching process with parameter $\lambda \in (0, \infty)$, starting from the root vertex $\varnothing$. Denote the offspring distribution by ξ. For any vertex $v \in \cup k = 0^\infty \mathbb{N}^k$, let $|v|$ represent its distance from the root vertex $\varnothing$.

We have developed the following algorithms to compute the maximal total weight and to construct the corresponding optimal path cover on finite trees explicitly.

Algorithm 1 Maximum total length of the cover by vertex-disjoint path

Input: Rooted finite tree $(\mathcal{T}, \varnothing)$.

Step 1: Arrange all other vertices of the tree based on their distance from the root.

Step 2: Assign to each vertex v a tuple $(W(v), Z(v))$ as follows:

Step 2-1. Initialize at leaves: For a leaf vertex v, set

$$W(v) = 0, \quad Z(v) = \omega(v) := \text{weight of the parent edge}$$

Step 2-2. Recursive Updates: For a non-leaf vertex v with k many children, say

$$v_1, v_2, \ldots, v_k, \quad \text{where} \quad Z(v_1) \geqslant Z(v_2) \geqslant \cdots \geqslant Z(v_k).$$

Set

$$W(v) := \sum_{i=1}^{k} W(v_i) + Z(v_1)_+ + Z(v_2)_+, \tag{1}$$

$$Z(v) := \omega_v - Z(v_2) \cdot \mathbb{1}_{\{Z(v_2)>0,\, k \geqslant 2\}}. \tag{2}$$

Output: $W(\varnothing)$.

Algorithm 2 Maximal weight path cover recovery

Input: Finite tree $\mathcal{T}$ with all $(W(v), Z(v))$ updated.
Step 1: Set $Z(\varnothing) = 0$
Step 2: Recursively recover edges from the root to the leaves as follows:
Step 2-1: Perform a breadth-first traversal of $\mathcal{T}$. For each non-leaf vertex $v \in \mathcal{T}$, without loss of generality, assume that it has k children

$$v_1, \ldots, v_k, \quad \text{where} \quad Z(v_1) \geqslant Z(v_2) \geqslant \cdots \geqslant Z(v_k).$$

Step 2-2: For each node v, let $\ell = |\{v_i : Z(v_i) > 0\}|$ be the number of children with positive Z-value. Then there are two cases:

- Case 1) $Z(v) > 0$
  - If $\ell \geqslant 1$, i.e., $Z(v_1) > 0$, then add the edge (v, v_1).
  - If $\ell = 0$, i.e., $Z(v_1) \leqslant 0$, then no edges are used.
- Case 2) $Z(v) \leqslant 0$
  - If $\ell \geqslant 2$, i.e., $Z(v_2) \geqslant 0$, then add two edges $(v, v_1), (v, v_2)$.
  - If $\ell = 1$, i.e., $Z(v_1) > 0$ and $Z(v_2) \leqslant 0$, then add the edge (v, v_1).
  - If $\ell = 0$, i.e., $Z(v_1) \leqslant 0$, then no edges are used.

Output: MWPC of $\mathcal{T}$.

To establish the validity of these algorithms, we provide the following propositions:

Proposition 3.1. *Let $\mathcal{T}$ be a finite tree. Then, the MP algorithm 1 applied to $(\mathcal{T}, v)$ for any vertex $v \in V(\mathcal{T})$ produces the same value.* i.e., *the output does not depend on the choice of the root.*

Proposition 3.2. *Let $(\mathcal{T}, \varnothing)$ be a rooted finite tree. Then the output of MP algorithm 1 is the maximal total weight of the the tree $(\mathcal{T}, \varnothing)$ and 2 applied to $(\mathcal{T}, \varnothing)$ is the maximal vertex-disjoint path recovery.*

The proofs of correctness for these algorithms are provided in the Appendix.

In order to verify our theoretical findings, we conducted various simulations using Python with the NumPy and NetworkX packages, which we present in the remainder of this paper. The code for the algorithms and simulations presented in this paper can be found here. The implementation of algorithms 1 and 2 is located in `Fall 2024/utility.py` within the `path_cover` function.

4 Analysis for Distribution of Z

Why study Z?

Once the distribution of Z is understood, we can analyze the LPT on a given tree, as demonstrated in the algorithms 1 and 2

Aim of this section. We shall establish three pillars that support the remainder of the paper:

1. *Well-definedness.* For every branching intensity $0 < \lambda < e$ the recursive distributional equation (RDE)

$$Z \overset{d}{=} \omega - \max_{1 \leqslant i \leqslant \xi}^{(2)} Z_i$$

admits a *unique* solution, so the distribution of Z is unambiguously specified. Here $\xi \sim \text{Poisson}(\lambda)$ and the Z_i are i.i.d. copies of Z.

2. *Transition-kernel interpretation.* On the *first-max path* produced by Algorithm 2, the sequence $(Z_{v_1}, Z_{v_2}, \dots)$ forms a time-homogeneous Markov chain with kernel

$$\kappa_\lambda(x, \mathrm{d}y) = \mathbb{P}\{\, Z_{\text{child}} \in \mathrm{d}y \;\big|\; Z_{\text{parent}} = x \,\}.$$

3. *Typical path length.* Combining branching-process estimates with the kernel description yields sharp exponential tails for the length L_{fm} of the first-max path:

$$\mathbb{P}\{L_{\text{fm}} \geqslant k\} \asymp q_\lambda^k \quad \text{with } 0 < q_\lambda < 1.$$

4.1 Well-Definedness of the Z Distribution

Consider the branching process with offsrping distribution $\xi \sim \text{Poisson}(\lambda)$. Let v be a node and ξ_v is the number of children of v. Define

$$\begin{aligned}\delta_v &:= \text{LPT(subtree of } v\text{; two down edges allowed)} - \text{LPT(subtree of } v\text{; one down edge allowed)} \\ &= \max_{i\in[\xi_v]}{}^{(2)}_{+}(\omega_{v_i} - \delta_{v_i}), \quad \text{(second maximum of Z-values of children of } v\text{)}\end{aligned}$$

where ω_{v_i} is the edge weight of (v, v_i).

Thus, we have

$$Z_v = \omega_v - \delta_v = \omega_v - \max_{i\in[\xi_v]}{}^{(2)}_{+} Z_{v_i}$$

Let

$$m(t) = \mathbb{E}t^\xi = \exp(\lambda(t-1))$$

denote the probability generating function of ξ. Then for $x \geqslant 0$, we denote the cumulative distribution function $F_\delta : [0,\infty) \to [0,1]$ of δ by

$$F_{\delta,\lambda}(x) := \mathbb{P}(\delta \leqslant x), \tag{3}$$

and the tail distribution $\bar{F}_{Z,\lambda} : [0,\infty) \to [0,1]$ of $Z \overset{d}{=} \omega - \delta$ by

$$\bar{F}(x) = \bar{F}_{Z,\lambda}(x) := \mathbb{P}(Z > x) = \mathbb{P}(\omega - \delta > x). \tag{4}$$

Here, we usually omit the subscript Z, λ as it is obvious in context. By the definition of δ, we have

$$\begin{aligned}\mathbb{P}(\delta \leqslant x) &= \sum_{k=0}^{\infty} \mathbb{P}(\xi = k)\mathbb{P}\left(\sum_{i=1}^{k} \mathbb{1}_{\omega_{v_i} - \delta_{v_i} > x} \leqslant 1\right), \\ &= m\left(\mathbb{P}(\omega - \delta \leqslant x)\right) + \mathbb{P}(\omega - \delta > x) m'\left(\mathbb{P}(\omega - \delta \leqslant x)\right)\end{aligned} \tag{5}$$

Let

$$p(x) := \mathbb{P}(Z > x) = \mathbb{P}(\omega - \delta > x) \tag{6}$$

Substituting it to the $m(t)$ we obtain the relation between δ and Z as

$$\mathbb{P}(\delta \leqslant x) = (1 + \lambda \bar{F}(x)) \cdot e^{-\lambda \bar{F}(x)} \tag{7}$$

In particular,

$$\mathbb{P}(\delta = 0) = (1 + \lambda \bar{F}(0)) \cdot e^{-\lambda \bar{F}(0)}. \tag{8}$$

The following lemma ensures that the function $\bar{F}$ is well-defined.

Lemma 4.1. *Suppose that $\lambda < e$. Let ω be a non-negative continuous random variable with probability density function $f(x)$. Then, there exists unique function $p(x) = \mathbb{P}(Z > x)$ such that*

$$\mathbb{P}(\delta \leqslant x) = m\left(1 - p(x)\right) + p(x) \cdot m'\left(1 - p(x)\right),$$

where $m(t) = \exp(\lambda(t-1))$ is the probability generating function of $\xi \sim \text{Poisson}(\lambda)$.

The proof of this lemma is provided in the appendix.

Thus, for each $\lambda < e$ $\bar{F}(x) = \bar{F}_\lambda(x) = \mathbb{P}(Z > x)$ is uniquely defined. For simplicity, we adopt the notation $\bar{F}(x)$.

We will explicitly compute the distribution of Z for the two examples: the uniform distribution and the exponential distribution.

Example 4.2. *For the $\omega \sim \text{Unif}(0,1)$ case, we have*

$$\bar{F}(x) = \mathbb{P}(Z > x) = \mathbb{P}(\omega - \delta > x) = \mathbb{E}(1 - x - \delta)_+$$

Moreover, from (7),

$$\bar{F}(x) = \mathbb{P}(\delta < \omega - x) = \int_x^1 \mathbb{P}(\delta < y - x)\, dy = \int_0^{1-x} F_\delta(y)\, dy = \int_0^{1-x} (1 + \lambda \bar{F}(y)) \cdot e^{-\lambda \bar{F}(y)}\, dy.$$

Thus, we need to solve the differential equation

$$\begin{aligned} &\bar{F}'(x) = -(1 + \lambda \bar{F}(1-x)) \cdot e^{-\lambda \bar{F}(1-x)} \text{ for } 0 < x < 1, \\ &\bar{F}(1) = 0, \quad \bar{F}'(0) = -1. \end{aligned}$$

Example 4.3. *Suppose that $\omega \sim \text{Exp}(1)$. Then,*

$$\bar{F}(x) = \mathbb{P}(\omega - \delta > x) = \mathbb{P}(\omega > \delta + x) = e^{-x} \mathbb{E} e^{-\delta}.$$

Define,

$$A := \lambda \mathbb{E} e^{-\delta} \in (0, \lambda).$$

Thus,

$$\bar{F}(x) = \frac{A}{\lambda} e^{-x} \text{ and } F_\delta(x) = (1 + A e^{-x}) \cdot \exp(-A e^{-x}).$$

Thus,

$$F_\delta(0) = \mathbb{P}(\delta = 0) = (1 + A) \cdot e^{-A}$$

But,

$$A = \lambda \mathbb{E} e^{-\delta} = \lambda \mathbb{P}(\delta = 0) + \lambda \int_0^1 A^2 y^2 e^{-Ay}\, dy$$
$$= \lambda \cdot (1 + A) \cdot e^{-A} + \lambda \cdot \frac{2e^{-A}}{A} \cdot (e^A - 1 - A - A^2/2).$$

There is unique solution $A_\lambda \in (0, \lambda)$ *to the equation*

$$\frac{A^2 e^A}{\lambda} = A(1 + A) + 2(e^A - 1 - A - A^2/2) = 2e^A - 2 - A,$$

or equivalently,

$$\lambda = \frac{A^2 e^A}{2e^A - 2 - A},$$

which is increasing function on $[0, \infty)$*. Therefore,*

$$F_\delta(x) = \begin{cases} (1 + A_\lambda \cdot e^{-x}) \cdot e^{-A_\lambda \cdot e^{-x}}, & \text{for } x \geqslant 0, \\ 0 & \text{o.w.} \end{cases}$$

is the cumulative distribution of δ*, and hence we can obtain the distribution of* $Z = \omega - \delta$*.*

Now, considering a Galton–Watson tree (GWT) $\mathcal{T}_\lambda$ with offspring distribution $\xi \sim \text{Poisson}(\lambda)$, we compute the expected maximum total weight $\mathbb{E}W(\mathcal{T}_\lambda)$ and the expected size of the tree $\mathbb{E}|\mathcal{T}_\lambda|$ using the recursive relations (1) and (2).

Lemma 4.4. *Suppose that* $\mathbb{E}Z_+ < \infty$*. Define* $h(x) := (2 + x)e^{-x}$*. Then, we have*

$$\frac{\mathbb{E}W(\mathcal{T}_\lambda)}{\mathbb{E}|\mathcal{T}_\lambda|} = 1 - \frac{1}{2} h(\lambda \bar{F}_\lambda(0)) \quad \text{and} \quad \mathbb{E}\left(\frac{W(\mathcal{T}_\lambda)}{|\mathcal{T}_\lambda|}\right) = 1 - \frac{1}{2\lambda} \int_0^\lambda h(y \bar{F}_y(0))\, dy. \tag{9}$$

The proof of this lemma is also deferred to the appendix.

4.2 Empirical Z distribution for $\omega \sim \text{Unif}(0, 1)$

Recall from Example 4.2 that the in the case of $\text{Unif}(0, 1)$ edge weights, the random variable Z satisfies the recursive distributional equation

$$\bar{F}'(x) = -(1 + \lambda \bar{F}(1 - x)) \cdot e^{-\lambda \bar{F}(1-x)} \text{ for } 0 < x < 1,$$
$$\bar{F}(1) = 0, \quad \bar{F}'(0) = -1.$$

To empirically verify this result, we compare the graph of the density function of Z to a histogram of Z values observed in simulated Galton-Watson trees against solved in the example, with the appropriate λ value.

However, the recursive distributional equation for the tail distribution of Z in the $\omega \sim \text{Unif}(0, 1)$ case does not admit an explicit solution, so we must find the PDF of Z via numerical integration. This script is included in the file `z-solver.py`.

To generate our Z-values, we generate Galton-Watson trees, compute the MVDP using **Algorithm 2**, and collect each root's first maximum child Z-value, $\max_{v \in N_\varnothing} Z(v)$.

The red curve plots the numerically computed density, and the blue histogram represents the empirical distribution of Z. They appear to match closely, supporting the correctness of the theoretical values.

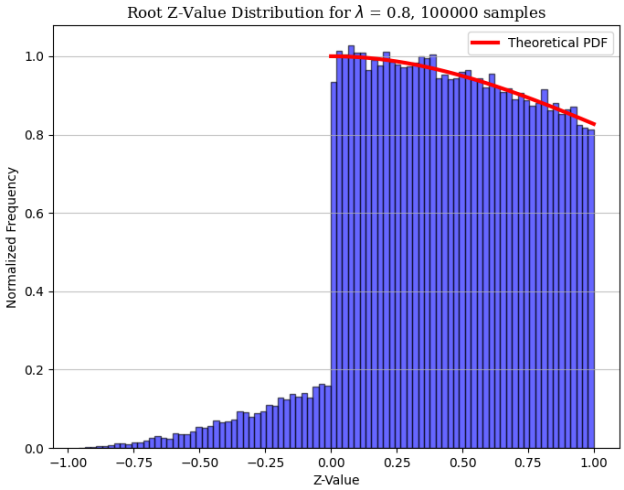


Figure 1: In blue is histogram of Z-values, red is density f_Z on $(0, 1)$

4.3 Transition Kernel Interpretation

Besides analyzing the behavior of Z-values for an arbitrary vertex, we seek to describe the behavior of Z-values along paths in the maximal-weight path cover. In particular, we study the behavior of the typical path produced by algorithm 2 for edge weight distributions $\omega \sim \text{Uniform}(0, 1)$ and $\omega \sim \text{Exp}(1)$.

Definition 4.5. *For a tree $\mathcal{T}$ rooted at $\varnothing$, the **first-max path** $\pi = \langle v_1, v_2, \ldots, v_k \rangle$ is the path beginning from the root's child with maximal Z-value, $v_1 = \arg\max_{u \in ch(\varnothing)} Z(u)$ in the maximal path cover of $\mathcal{T}$. It can be recovered by traversing maximal path edges starting from v_1, that is, $v_{i+1} = \arg\max_{u \in ch(v_i)} Z(u)$ for each $i \in \{1, 2, \ldots, k-1\}$.*

For a Galton-Watson tree $\mathcal{T}_\lambda = \text{GWT}(\lambda)$, we observe that the first-max path behaves as a finite, discrete-time Markov chain with measurable state space. Specifically, for a first-max path $\pi = \langle v_1, v_2, \ldots, v_k \rangle$ we have the sequence of Z-values $Z(v_1), Z(v_2), \ldots, Z(v_k)$ satisfying the Markov property. Such a Markov chain is supported on a measurable state space $S \subseteq \mathbb{R}$, which for $\omega \sim \text{Uniform}(0, 1)$ is $[-1, 1]$ and for $\omega \sim \text{Exp}(1)$ is all of $\mathbb{R}$. Given this setup, it is natural to study the stochastic kernel

$$\kappa_\lambda(x, \mathrm{d}y) = \mathbb{P}\left(Z(v_{i+1}) \in \mathrm{d}y \mid Z(v_i) = x\right).$$

However, direct computation of the kernel is prohibitively difficult in this case, so instead we discretize the state space S into finitely many bins, allowing for the use of matrix techniques for analyzing finite-state Markov chains. Fixing some number of bins $m \in \mathbb{N}$, we partition S into disjoint intervals $L_1, L_2, \ldots, L_m$ so that $\bigsqcup_i L_i = S$. When $\omega \sim \text{Uniform}(0,1)$, $S = [-1, 1]$ and we define the bins linearly, i.e., each L_i has equal length. When $\omega \sim \text{Exp}(1)$, $S = \mathbb{R}$, by using $\omega \sim -\log U$, where $U \sim \text{Unif}(0, 1)$, we define the bins with log scaling. That is, each interval is given by

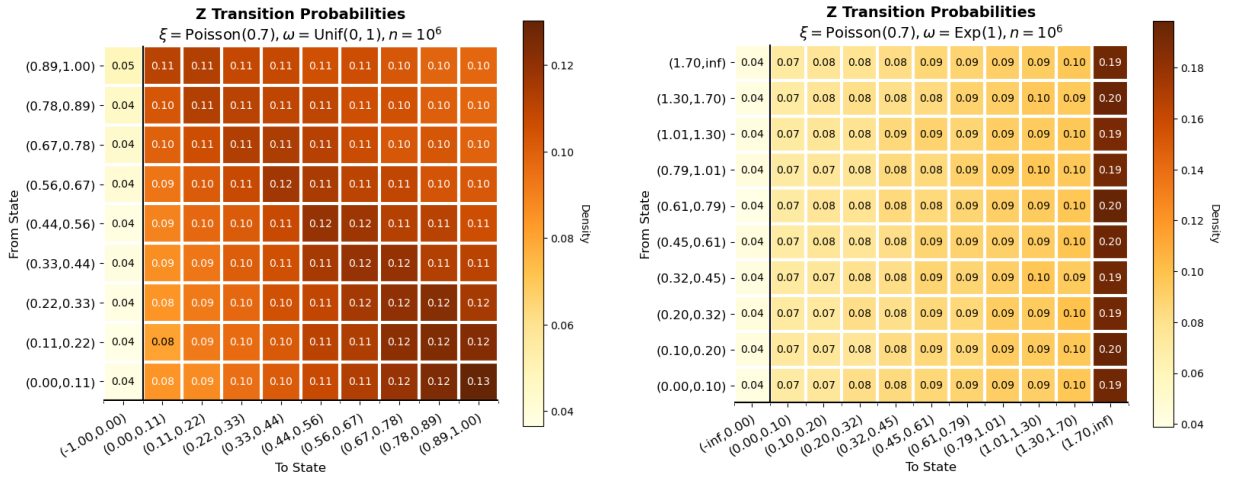
$$L_i = \left(\log\left(\frac{m}{i}\right), \log\left(\frac{m}{i-1}\right)\right),$$

for $i = m, m-1, \ldots, 2$. Then, given two adjacent vertices $v_{\text{from}}, v_{\text{to}}$ in the first-max path π, a transition from $Z(v_{\text{from}})$ to $Z(v_{\text{to}})$ can be viewed as a discrete transition from L_i to L_j where $Z(v_{\text{from}}) \in L_i$ and $Z(v_{\text{to}}) \in L_j$. This allows us to compute a transition matrix (P_{ij}) where each

entry is given by

$$P_{ij} = \mathbb{P}(Z(v_{\text{to}}) \in L_j \mid Z(v_{\text{from}}) \in L_i). \tag{10}$$

Using our implementation of algorithms 1 and 2, we simulated this discretization process in `better transition probabilities.ipynb`. For each of $n = 10^6$ samples, we generate a tree $\mathcal{T} = \text{GWT}(\lambda)$ and run algorithms 1 and 2 on $\mathcal{T}$. This allows us to extract the first-max path $\pi = \langle v_1, v_2, \ldots, v_k \rangle$ of $\mathcal{T}$. Then for every edge $(v_{\text{from}}, v_{\text{to}})$ in π, we log the occurrence of a transition between discretized bins L_i, L_j as above. Once we have collected n samples, we compute the sample probability for each P_{ij} given in equation 10. As such, note that the sum of the densities in each row (i.e., each "From State") sum to 1 since the L_i's form a partition on S. Finally, we render the following heatmaps of the resulting transition probability matrix.



(a) Uniform$(0, 1)$ weights, linear bin scaling

(b) Exp(1) weights, log bin scaling

Figure 2: Empirical Discretized Transition Matrices for $n = 10^6$ samples

The transition matrix for uniform weight trees is shown in figure 2a. We note the concentration of density along the anti-diagonal. In words, this means that first-max paths in the MWPC with uniform weights tend to alternate between high (near 1) and low (positive, near 0) Z-values. Additionally, in both 2a and 2b, we see a small amount of density evenly distributed in the non-positive "To State" column on the left. This aligns with intuition, since a transition to a vertex v_j with non-positive Z-value occurs if and only if v_j is the last vertex in the first-max path (i.e., $j = k$) since a non-positive Z-value means that it is disadvantageous for v_j to take its parent edge, so the first-max path ends. Therefore we see a transition to non-positive Z exactly once per sample, explaining the relatively low density compared to the rest of the heatmap.

In the case of exponential edge weights shown in figure 2b, we observe different behavior. Since Z-values are unbounded in the exponential case, we see that a typical transition prefers vertices with larger Z-value, as demonstrated by the gradually increasing density with each "To State" column. Again, this aligns with our understanding since a path in the MWPC

5 Path Length Analysis for the First-Max Path in GWBP Trees

Following Definition 4.5, we restrict attention to the *first-max path*—the unique self-avoiding path that begins at the root and, at every vertex, follows the child with the largest Z-value. This choice is justified by two facts:

(i) **Typicality.** Algorithm 2 always retains the first-max edge in the minimum-weight path cover (MWPC).

(ii) **Markov structure.** The sequence $(Z(v_1), Z(v_2), \dots)$ observed along the path forms a time-homogeneous Markov chain (Section 5.2).

Throughout we fix an offspring distribution $\xi \sim \text{Poisson}(\lambda)$ with $\lambda < 1$ and independent edge weights $\omega_e \sim \text{Unif}(0, 1)$.

5.1 Methodology

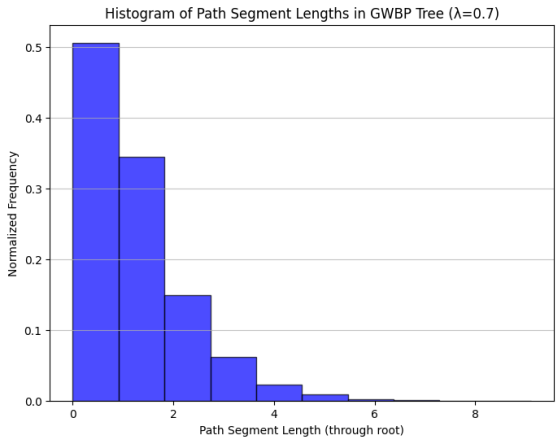
The analysis proceeds in three stages:

1. **Path-segment extraction**
   - For offspring rate $\lambda = 0.7$ we simulate $n = 100,000$ independent Galton–Watson trees, grown breadth-first to depth `MAXLEVEL`$= 10$ or until extinction.
   - Apply Algorithm 2 (MWPC) to every tree and read off the first-max path $\pi = \langle v_1, \dots, v_k \rangle$.
   - Record its segment length$= k$
2. **Histogram and empirical PMF construction**
   - Tally the frequencies f_k of observed lengths.
   - Form the empirical pmf $\widehat{P}_\lambda(k) = f_k / \sum_j f_j$.
   - Discard zero-count bins to avoid taking $\log 0$ in the next step.
3. **Log–linear fitting**
   - Regress $\log \widehat{P}_\lambda(k)$ on $(k-1)$ via ordinary least squares,

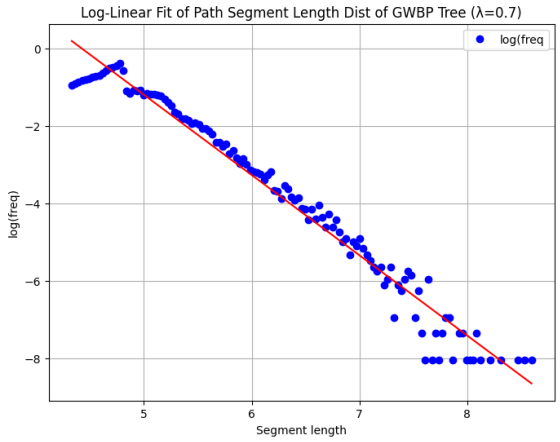
$$\log \widehat{P}_\lambda(k) = m_\lambda (k-1) + c,$$

   so the slope m_λ estimates $\log q_\lambda$ of the geometric tail.

The resulting histogram and the corresponding log–linear fit are shown in Figure 3a and Figure 3b, respectively.



(a) Histogram ($\lambda = 0.7$)



(b) Log–linear fit

Figure 3: Empirical first-max path statistics for $\lambda = 0.7$. Panel (a) shows the normalized histogram of segment lengths k for $n = 10^5$ simulated GWTs. Panel (b) plots $\log \widehat{P}_{0.7}(k)$ (log of the empirical frequency) against $(k-1)$; the red line is the least-squares fit whose slope is $m_{0.7} \approx -2.01$.

5.2 Results and Interpretation

The resulting PMF consistently shows a sharp exponential decay, with slopes near -2 across a wide range of simulations. This matches theoretical predictions for GWBP trees, indicating a clear scaling relationship:

$$P(k) \propto q^k$$

where $q = e^m$ captures the rate of decay. Additionally, the maximum segment length is bounded by the scaling relation:

$$\frac{\log n}{\text{slope of log-linear fit}}$$

6 Subcritical Erdős–Rényi Simulations

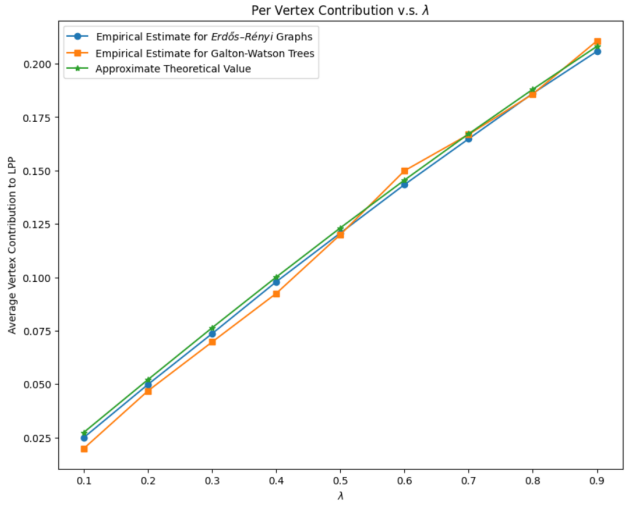
Consider the last passage time (LPT) on Erdős–Rényi graphs $G(n, \frac{\lambda}{n})$ with $\lambda < 1$. It is well-known that the size of non-tree components is asymptotically negligible. Consequently, it is justified to approximate the last passage time on Erdős–Rényi graphs by the corresponding problem on Galton–Watson trees (GWT). Formally, we expect that

$$\frac{W_{n,\lambda}}{n} \to \frac{W(\mathcal{T}_\lambda)}{|\mathcal{T}_\lambda|}.$$

where $W_{n,\lambda}$ is defined as the last passage time over the Erdős–Rényigraph with parameter λ.

The figure below empirically computes these values and compares them against the approximate theoretical per-vertex contribution derived from GWT(λ). Based on the figure, we observe that the empirical values for both Erdős–Rényi graphs and Galton–Watson trees closely align and consistently follow the theoretical prediction given by the integral in Lemma 4.4.

Therefore, we conclude that our integral is indeed correct, and furthermore, that the theoretical scaled last passage percolation on Erdős–Rényi graphs coincides precisely with that of the Galton–Watson trees.



For the experiment, 1000 samples were used for each Erdős–Rényi graph or Galton-Watson Tree. The Erdős–Rényi graphs consisted of 1000 nodes each. The theoretical value was computed via numerical integration of the integral in proposition 4.1. Although some values were computed with fewer interpolating points than others, this effect is negligible with respect to the the total amount of interpolating points each theoretical value used. The code is provided in the file `ER_Simulations.ipynb` linked in the GitHub.

Remark 1. *For the regime $1 < \lambda < e$, the random graph $G\big(n, \frac{\lambda}{n}\big)$ already contains a unique giant component of linear size. Nevertheless, its* 2-core*—the maximal subgraph with minimum degree at least two—is asymptotically negligible: it comprises only $o(n)$ vertices, while all but $o(n)$ edges lie in hanging tree-like branches attached to that core (see the "e-phenomenon" of the Karp–Sipser process,* [3, Theorem 2.2]*).*

This observation suggests the following programme for $1 < \lambda < e$.

1. *Apply the Karp–Sipser* leaf-removal *algorithm until no leaves remain, thereby peeling off the hanging trees from the giant component.*
2. *Analyse the last-passage time separately on*
   (a) *the pruned giant component (whose core size is $o(n)$), and*
   (b) *the collection of detached Galton–Watson trees, using the methods developed for* $\mathrm{GWT}(\lambda)$.

Because the core is sublinear, we expect the leading-order contribution to come from the tree part, so the scaled LPT on $G\big(n, \frac{\lambda}{n}\big)$ should coincide with that on the associated Galton–Watson forest. Establishing this rigorously is left for future work.

7 Supercritical GWT: Limit of $W(\mathcal{T}_\lambda)/|\mathcal{T}_\lambda|$

Here, we consider a supercritical Galton-Watson tree $\mathcal{T}_\lambda = \mathrm{GWT}(\lambda)$. When $\lambda > 1$, the offspring distribution $\xi \sim \mathrm{Poisson}(\lambda)$ of $\mathcal{T}_\lambda$ has expectation $\mathbb{E}\xi = \lambda > 1$. Hence, the Galton-Watson process is expected to run indefinitely with positive probability. Since infinite trees cannot be simulated directly, we truncate the tree at a finite level k and examine the behavior of the size-biased last passage time (SLPT) $W(\mathcal{T}_\lambda)/\,|\mathcal{T}_\lambda|$ on this truncated tree as we vary k, λ.

The code for the following data can be found in `supercritical avg weight.ipynb`.

k \ λ	1.1	1.3	1.5	1.7	1.9
6	0.239	0.261	0.286	0.294	0.308
8	0.235	0.268	0.286	0.300	0.307
10	0.240	0.272	0.287	0.301	0.310
12	0.238	0.274	0.292	0.300	0.307
14	0.245	0.272	0.288	0.300	0.306

Table 1: Values of the size-biased LPT $\frac{W(\mathcal{T}_\lambda)}{|\mathcal{T}_\lambda|}$ in $\mathcal{T}_\lambda = \mathrm{GWT}(\lambda)$ truncated at level k, with 5000 samples per entry.

Empirical Conclusions. Let

$$c_k(\lambda) \;=\; \frac{W\big(\mathcal{T}_\lambda^{(k)}\big)}{\big|\mathcal{T}_\lambda^{(k)}\big|}, \qquad k \in \mathbb{N},\ \lambda > 1,$$

denote the mean per–vertex weight on the Galton–Watson tree $\mathcal{T}_\lambda$ truncated at depth k. The data in Figure 1 support the following:

(1) Convergence in depth. For every fixed $\lambda > 1$,

$$c_k(\lambda) \;\xrightarrow[k\to\infty]{}\; c(\lambda),$$

with convergence already numerically stable by $k \approx 10$.

(2) Monotone growth in λ. The limiting constant $c(\lambda)$ is strictly increasing on $(1, \infty)$:

$$\lambda_1 < \lambda_2 \implies c(\lambda_1) < c(\lambda_2).$$

(3) Negligible contribution of the core. The oscillations of $c_k(\lambda)$ for successive k are of order 10^{-3}, consistent with the intuition that deeper generations add many vertices but only a vanishing fraction of additional path weight.

These findings indicate that the scaled last–passage time on a supercritical Galton–Watson tree admits a deterministic limit $c(\lambda)$ and that this limit grows with the branching parameter λ.

8 Conclusion and Open Questions

In this study, we explored the asymptotic behavior of mean-field last passage percolation (MF LPP) with continuous edge weights in the sparse regime, focusing on Erdős–Rényi random graphs and their local approximations via Galton–Watson branching trees. By introducing continuous edge weights, such as Uniform$(0, 1)$ and Exp(1) distributions, we extended previous analyses from the binary setting.

We developed robust algorithmic methods to efficiently compute maximal-weight vertex-disjoint path covers on finite trees, verified their correctness, and used extensive simulations to validate theoretical predictions. Specifically, we demonstrated the well-definedness of the recursive distributional

equation governing vertex advantage (Z-values), characterized the Markovian structure of optimal paths, and confirmed that typical path segments exhibit geometric tail behavior.

Furthermore, we numerically verified the convergence of scaled last passage times on Erdős–Rényi graphs to their Galton–Watson tree counterparts in the subcritical regime. This result highlights a deep connection between global combinational optimization on random graphs and local tree structures, supporting a rich interplay between probability, optimization, and combinatorics.

Several intriguing open questions remain for future research:

1. **Rigorous proofs of convergence:** While numerical evidence strongly suggests convergence, a rigorous analytical proof for the convergence of scaled last passage times in Erdős–Rényi graphs to Galton–Watson tree limits remains open.

2. **Central limit theorems:** Investigating the Gaussian fluctuations of the last passage time in both subcritical and supercritical regimes would deepen understanding of universality classes and scaling limits.

3. **Supercritical regime analysis:** While empirical results indicate convergence of the scaled last passage time in supercritical Galton–Watson trees, rigorous identification and analysis of the limiting constant $c(\lambda)$ for general continuous edge weights is needed.

9 Appendix

9.1 Proof of Proposition 3.1

By the connectedness of the tree, it suffices to show that the outputs of the algorithm are the same for any two neighboring vertices x and y in $\mathcal{T}$. Define

$$N_x := \{v \in \mathcal{T} : v \sim x\} \quad \text{and} \quad N_{x\setminus y} := N_x \setminus \{y\}$$

Similarly, we define N_y and $N_{y\setminus x}$ for the vertex y. Let $x_1, x_2 \in N_x$ be the children of x such that

$$Z(x_1) = \max_{v\in N_x} Z(v) \quad \text{and} \quad Z(x_2) = \max_{v\in N_x}{}^{(2)} Z(v),$$

where $\max^{(j)}$ is the j-th maximum. Define y_1 and y_2 in a similar manner for y.

We complete the proof by considering all possible cases for the values of

$$(Z(x_1), Z(x_2), Z(y_1), Z(y_2)), \quad \text{where} \quad Z(x_1) \geqslant Z(x_2), \text{ and } Z(y_1) \geqslant Z(y_2)$$

Furthermore, interchanging x and y allows us to further reduce the number of cases to consider.

1. $Z(x_1) \geqslant Z(x_2) > 0$ and $Z(y_1) \geqslant Z(y_2) > 0$.

   Setting x as the root, *i.e.,* $\varnothing = x$ yields

   $$W(y) = \sum_{v\in N_{y\setminus x}} W(v) + Z(y_1) + Z(y_2) \quad \text{and} \quad Z(y) = 0,$$

   and hence we obtain

   $$\begin{aligned} W(\varnothing) = W(x) &= W(y) + \sum_{v\in N_{x\setminus y}} W(v) + Z(x_1) + Z(x_2) \\ &= \sum_{v\in N_{y\setminus x}} W(v) + Z(y_1) + Z(y_2) + \sum_{v\in N_{x\setminus y}} W(v) + Z(x_1) + Z(x_2). \end{aligned}$$

Similarly, if $\varnothing = y$, then $W(x) = \sum_{v \in N_{x \setminus y}} W(v) + Z(x_1) + Z(x_2)$ and $Z(x) = 0$. Thus the value at the root is again

$$W(\varnothing) = W(y) = \sum_{v \in N_{y \setminus x}} W(v) + Z(y_1) + Z(y_2) + \sum_{v \in N_{x \setminus y}} W(v) + Z(x_1) + Z(x_2).$$

2. $Z(x_1) > 0, Z(x_2) = 0$, and $Z(y_1) \geqslant Z(y_2) > 0$.
   Setting $\varnothing = x$ yields

$$W(y) = \sum_{v \in N_{y \setminus x}} W(v) + Z(y_1) + Z(y_2) \quad \text{and} \quad Z(y) = 0.$$

   In this case, either $y = x_2$ or $y \neq x_2$, and we obtain

$$\begin{aligned} W(\varnothing) = W(x) &= W(y) + \sum_{v \in N_{x \setminus y}} W(v) + Z(x_1) \\ &= \sum_{v \in N_{y \setminus x}} W(v) + Z(y_1) + Z(y_2) + \sum_{v \in N_{x \setminus y}} W(v) + Z(x_1). \end{aligned}$$

   On the other hand, if we set $\varnothing = y$, then

$$W(x) = \sum_{v \in N_{x \setminus y}} W(v) + Z(x_1) \quad \text{and} \quad Z(x) > 0.$$

   In this case, either $x = y_1$ or $x = y_2$. By the definition of y_1, y_2, regardless of whether $x = y_1$, or $x = y_2$, we have

$$W(\varnothing) = W(y) = \sum_{v \in N_{y \setminus x}} W(v) + Z(y_1) + Z(y_2) + \sum_{v \in N_{x \setminus y}} W(v) + Z(x_1).$$

   The same argument, where the roles of x and y are swapped, applies to the case when $Z(x_1) \geqslant Z(x_2) > 0$, and $Z(y_1) > 0, Z(y_2) = 0$. Proofs for the other cases are similar.

3. $Z(x_1) = Z(x_2) = 0$ and $Z(y_1) \geqslant Z(y_2) > 0$.
4. $Z(x_1) = Z(x_2) = 0$, $Z(y_1) > 0$, and $Z(y_2) = 0$.
5. $Z(x_1) = Z(x_2) = Z(y_1) = Z(y_2) = 0$.

□

9.2 Proof of Proposition 3.2

If the size of a tree $n \leqslant 3$, then the statement is obviously true. Suppose, for contradiction, that there exsits a smallest tree $\mathcal{T}$ with size $|\mathcal{T}| = k$ such that the MP algorithm fails to yield the maximal total weight for $\mathcal{T}$. That is, there exists another vertex-disjoint path cover $\mathcal{C}$ such that

$$W(\mathcal{C}) > W(\mathcal{C}_{\mathrm{MP}}(\mathcal{T})), \tag{11}$$

where $W(\cdot)$ is the total weight of given path cover and $\mathcal{C}_{\mathrm{MP}}(\cdot)$ is the output of MP algorithm 2. *i.e.,* the MWPC of given tree. Note that by the minimality of k, the MP algorithm is optimal for all trees of size less than k.

Let $o \in \mathcal{T}$ be a leaf and denote its parent by x. Re-root $\mathcal{T}$ at o:

$$(\mathcal{T}, o) : \text{Tree rooted at } o;\ x \text{ is the unique child of } o.$$

We may assume that x has k children

$$x_1, \dots, x_k \quad \text{such that} \quad Z(x_1) \geqslant \cdots \geqslant Z(x_k), \quad k \geqslant 1$$

in the rooted tree $(\mathcal{T}, o)$.

Consider the subtree $\mathcal{T} \setminus \{o\}$

Then, there are four cases. either $\mathcal{C}$ covers the leaf o (*i.e.,* use the edge (o, x)), or it does not.

Case 1. Both $\mathcal{C}$ and $\mathcal{C}_{\mathbf{MP}}(\mathcal{T})$ cover o.

In this case, the problems are reduced to compare $W(\mathcal{C} \setminus \{(o, x)\})$ and $W(\mathcal{C}_{\mathrm{MP}}(\mathcal{T}) \setminus \{(o, x)\})$. By the induction hypothesis, we have

$$W(\mathcal{C} \setminus \{(o, x)\}) \leqslant W(\mathcal{C}_{\mathrm{MP}}(\mathcal{T}) \setminus \{(o, x)\}),$$

which contradicts (11).

Case 2. Both $\mathcal{C}$ and $\mathcal{C}_{\mathbf{MP}}(\mathcal{T})$ do not cover o.

The same argument applies as in Case 1, since we only need to compare the each value on the subtree $\mathcal{T} \setminus \{o\}$.

Case 3. $\mathcal{C}$ does not cover o, but $\mathcal{C}_{\mathbf{MP}}(\mathcal{T})$ covers o.

Since the MVDP of the algorithm covers o, we have

$$Z(x) > 0, \quad \text{or equivalently} \quad Z(x_2) < \omega(x),$$

where $\omega(x)$ is the weight of the parent edge of x. Then, we obtain

$$W(\mathcal{C}_{\mathrm{MP}}(\mathcal{T})) = \sum_{j=1}^{k} W(x_j) + Z(x_1) + \omega(x) \geqslant \sum_{j=1}^{k} W(x_j) + Z(x_1) + Z(x_2) \geqslant W(\mathcal{C}),$$

which contradicts (11).

Case 4. $\mathcal{C}$ covers o, but $\mathcal{C}_{\mathbf{MP}}(\mathcal{T})$ does not cover o.

Since the MVDP of the algorithm does not cover o, we have

$$Z(x) \leqslant 0, \quad \text{or equivalently} \quad Z(x_2) \geqslant \omega(x),$$

where $\omega(x)$ is the weight of the parent edge of x.

Let $(\mathcal{T}, x_i)$ denote the subtree rooted at x_i for $i = 1, \dots, k$. By the induction hypothesis, the MP algorithm is optimal for the subtrees $(\mathcal{T}, x_i)$:

$$W\left(\mathcal{C}|_{(\mathcal{T}, x_i)}\right) \leqslant W\left(\mathcal{C}_{\mathrm{MP}}|_{(\mathcal{T}, x_i)}\right) \quad \text{for } i \in \{1, \dots, k\},$$

where $\mathcal{C}|_{(\mathcal{T}, x_i)}$ denotes the restriction of the path cover $\mathcal{C}$ to the subtree $(\mathcal{T}, x_i)$ rooted at x_i. *i.e.,* $\mathcal{C}|_{(\mathcal{T}, x_i)} := \{(i, j) \in \mathcal{C} : (i, j) \in (\mathcal{T}, x_i)\}$. Observe that $\mathcal{C}$ covers x_1 and does not cover x_i for $i = 2, 3, \dots, k$, while $\mathcal{C}_{\mathrm{MP}}$ covers x_1, x_2 and does not cover all other x_i's. Thus,

$$W\left(\mathcal{C}_{\mathrm{MP}}|_{(\mathcal{T}, x_i)}\right) = W(x_i) \quad \text{for } i \in \{1, \dots, k\} \setminus \{2\}.$$

By the definition of $Z(x_2)$, we have

$$Z(x_2) = W(x_2) + \omega(x_2) - W\left(\mathcal{C}_{\mathrm{MP}}|_{(\mathcal{T}, x_i)}\right).$$

Therefore, we obtain

$$\begin{aligned} W(\mathcal{C}_{\mathrm{MP}}(\mathcal{T})) - W(\mathcal{C}) &= \left\{\sum_{i=1}^{k} W(x_i) + Z(x_1) + Z(x_2)\right\} - \left\{\sum_{i=1}^{k} W\left(\mathcal{C}_{\mathrm{MP}}|_{(\mathcal{T},x_i)}\right) + Z(x_1) + \omega(x)\right\}, \\ &\geqslant \{W(x_2) + \omega(x_2)\} - \left\{W\left(\mathcal{C}|_{(\mathcal{T},x_2)} + \omega(x)\right)\right\}, \\ &= Z(x_2) - \omega(x) \geqslant 0, \end{aligned}$$

which contradicts (11). □

9.3 Proof of Lemma 4.1

Note that $p(x)$ satisfies the following equation

$$p(x) = \int_0^\infty (1 + \lambda p(t)) e^{-\lambda p(t)} f(t + x)\, dt$$

Let $g(x) := \lambda p(x)$.

$$\mathbb{P}(\delta \leqslant x) = (1 + g(x))e^{-g(x)} \text{ for } x \in \mathrm{supp}(\omega).$$

Therefore,

$$\begin{aligned} p(x) = \frac{1}{\lambda} g(x) = \mathbb{P}(\omega - \delta > x) &= \int_x^\infty \mathbb{P}(\delta < y - x) f(y)\, dy \\ &= \int_0^\infty \mathbb{P}(\delta < t) f(t + x)\, dt \\ &= \int_0^\infty (1 + \lambda p(t)) e^{-\lambda p(t)} f(t + x)\, dt \end{aligned}$$

The last equality holds because δ (since ω is continuous) is a continuous random variable.

In other words,

$$p = T_\omega p,$$

where $T := T_\omega$ is the integral operator in the previous equation. Let $h(x) = h_\lambda(x) := (1 + \lambda x)e^{-\lambda x}$. Then,

$$h'(p) = -\lambda e^{-\lambda p}(1 + \lambda p) + \lambda e^{-\lambda p} = -\lambda^2 p e^{-\lambda p}.$$

Since $xe^{-x} \leqslant 1/e$, we have $|h'(p)| \leqslant \lambda \cdot \lambda p e^{-\lambda p} \leqslant \lambda/e$. Thus, we obtain

$$|(T(p_1(x)) - (T(p_2(x))| \leqslant C_0(\lambda) \int_0^\infty |p_1(t) - p_2(t)| f(t + x) dt \leqslant C_0 \int_x^\infty f(t)\, dt \|p_2 - p_1\|$$

where $C_0(\lambda) = \sup_{p \in (0,1)} |h'(p)| \leqslant \lambda/e < \infty$.

Since $\lambda < e$, $C_0 < 1$, and hence T is a contraction. By the fixed point theorem, we have the existence and uniqueness of function p. □

9.4 Proof of Lemma 4.4

Suppose there are k samples $Z_1, \ldots, Z_k$. Then,

$$\mathbb{P}(Z_{(k)} > x) = 1 - \mathbb{P}(Z \leqslant x)^k = 1 - (1 - \bar{F}_\lambda(x))^k,$$

and

$$\begin{aligned}\mathbb{P}(Z_{(k-1)} > x) &= 1 - \mathbb{P}(Z \leqslant x)^k - k\mathbb{P}(Z > x) \cdot \mathbb{P}(Z \leqslant x)^{k-1}, \\ &= 1 - (1 - \bar{F}_\lambda(x))^k - k\bar{F}_\lambda(x)(1 - \bar{F}_\lambda(x))^{k-1},\end{aligned}$$

where $Z_{(i)}$ denote the i-th maximum over k samples. Thus, we have

$$\begin{aligned}&\mathbb{E}\left(\max_{i\in[\xi_v]}{}^{(1)}_{+} Z(v_i)_+ + \max_{i\in[\xi_v]}{}^{(2)}_{+} Z(v_i)_+\right) \\ &= \int_0^\infty \sum_{k=0}^\infty e^{-\lambda}\frac{\lambda^k}{k!} \cdot \left(1 - (1 - \bar{F}_\lambda(x))^k + 1 - (1 - \bar{F}_\lambda(x))^k - k\bar{F}_\lambda(x)(1 - \bar{F}_\lambda(x))^{k-1}\right) dx \\ &= \int_0^\infty 2 - (2 + \lambda\bar{F}_\lambda(x)) \cdot e^{-\lambda\bar{F}_\lambda(x)} dx.\end{aligned}$$

The last integral is finite if $\mathbb{E}Z_+ < \infty$.

Proof of finiteness of the last integral. Note that

$$2 - (2 + \lambda y)e^{-\lambda y} = \int_0^{\lambda y} e^{-s}(1 + s)\, ds.$$

Since $e^{-s} \leqslant 1$ and since $\bar{F}(x)^2 \leqslant \bar{F}(x) \leqslant 1$, we have

$$\begin{aligned}2 - \left(2 + \lambda\,\bar{F}(x)\right) e^{-\lambda\,\bar{F}(x)} &= \int_0^{\lambda\bar{F}(x)} e^{-s}(1 + s)\, ds \\ &\leqslant \int_0^{\lambda\bar{F}(x)} (1 + s)\, ds = \lambda\bar{F}(x) + \frac{1}{2}\lambda^2\bar{F}(x)^2 \leqslant \left(\lambda + \tfrac{1}{2}\lambda^2\right)\bar{F}(x).\end{aligned}$$

Therefore,

$$\int_0^\infty 2 - (2 + \lambda\bar{F}_\lambda(x)) \cdot e^{-\lambda\bar{F}_\lambda(x)}\, dx \leqslant \int_0^\infty (\lambda + \tfrac{1}{2}\lambda^2)\bar{F}(x)\, dx = (\lambda + \tfrac{1}{2}\lambda^2)\mathbb{E}Z_+.$$

□

Note that

$$h'(x) = -(1 + x)e^{-x}.$$

Thus, $\bar{F}'_\lambda(x) = h'(\lambda\bar{F}_\lambda(1 - x))$ and from (1), we obtain

$$\begin{aligned}\frac{\mathbb{E}W(\mathcal{T}_\lambda)}{\mathbb{E}|\mathcal{T}_\lambda|} = \mathbb{E}\left(Z(v_1)_+ + Z(v_2)_+\right) &= 2 - \int_0^1 h(\lambda\bar{F}_\lambda(x))\, dx \\ &= 2 - xh(\lambda\bar{F}_\lambda(x))\big|_0^1 + \int_0^1 xh'(\lambda\bar{F}_\lambda(x))\lambda\bar{F}'_\lambda(x)\, dx \\ &= \lambda\int_0^1 xh'(\lambda\bar{F}_\lambda(x))h'(\lambda\bar{F}_\lambda(1 - x))\, dx \\ &= \lambda\int_0^1 (1 - x)h'(\lambda\bar{F}_\lambda(1 - x))h'(\lambda\bar{F}_\lambda(x))\, dx.\end{aligned}$$

Simplifying we get

$$\begin{aligned}\frac{\mathbb{E}W(\mathcal{T}_\lambda)}{\mathbb{E}|\mathcal{T}_\lambda|} &= \frac{\lambda}{2}\int_0^1 h'(\lambda\bar{F}_\lambda(x))h'(\lambda\bar{F}_\lambda(1-x))\,dx \\ &= \frac{1}{2}\int_0^1 h'(\lambda\bar{F}_\lambda(x))\cdot\lambda\bar{F}'_\lambda(x)\,dx = \frac{1}{2}h(\lambda\bar{F}_\lambda(x))\big|_0^1 = 1-\frac{1}{2}h(\lambda\bar{F}_\lambda(0)).\end{aligned}$$

By a theorem in [1], we have

$$\mathbb{E}\left(\frac{W(\mathcal{T}_\lambda)}{|\mathcal{T}_\lambda|}\right) = \frac{1}{\lambda}\int_0^\lambda \frac{\mathbb{E}W(\mathcal{T}_y)}{\mathbb{E}|\mathcal{T}_y|}\,dy = \frac{1}{\lambda}\int_0^\lambda \left(1-\frac{1}{2}h(y\bar{F}_y(0))\right)\,dy = 1-\frac{1}{2\lambda}\int_0^\lambda h(y\bar{F}_y(0))\,dy.$$

□

References

[1] D. Aldous and J. Pitman, *Tree-valued markov chains derived from galton-watson processes*, Annales de l'Institut Henri Poincaré, Probabilités et Statistiques **34** (1998), no. 5, 637–686.

[2] D. J. Aldous and A. Bandyopadhyay, *A survey of max-type recursive distributional equations*, Annals of Applied Probability **15** (2005), no. 2, 1047–1110.

[3] M. Glasgow, M. Kwan, A. Sah, and M. Sawhney, *A central limit theorem for the matching number of a sparse random graph*, Journal of the London Mathematical Society **111** (2025), no. 4, e70101.