RECOVERING THE FIRST VERTICES IN A PREFERENTIAL ATTACHMENT MODEL

Math 199 –

Thesis Presentation

Thu Nguyen

Advisor: Professor Ery Arias-Castro

CONTENTS

1. INTRODUCTION

Graph Theory ∩ Probability

- **Preferential Attachment Model:**
  - Seeks to describe the “the rich get richer” phenomena
    - Early advantages will compound, leading to bigger gaps
  - Dynamic graph model: grows over time
  - Offers great flexibility in twisting the growth rules, potential to adapting to real-life situations

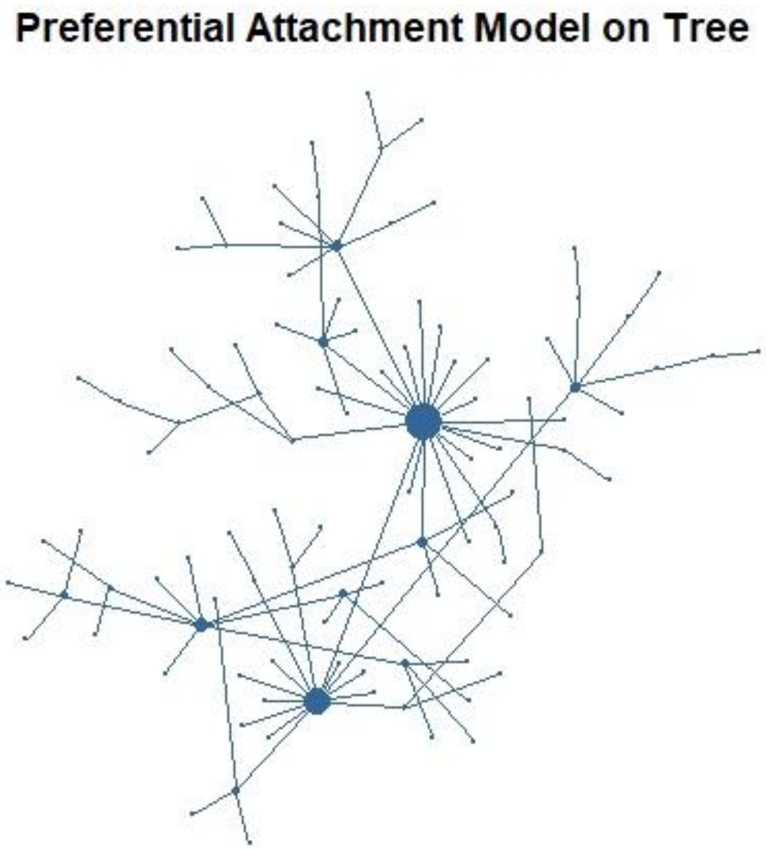
1. INTRODUCTION

- *PA model on Tree,* $T^{(\alpha)}(n)$, specification:
  - Parameters: $(T^{(0)}, \alpha)$,
    - where α controls the weights of favoritism, and $T^{(0)}$ is the starting seed graph
  - We define $T^{(\alpha)}(n)$ iteratively as follows:
    - Suppose at time n, we have the tree graph $T^{(\alpha)}(n)$, at the next iteration $n+1$:
      - Add new vertex, labeled $n+1$
      - Choose an existing vertex from $T^{(\alpha)}(n)$, say u, with probability

$$P(u\ is\ selected\ at\ step\ n+1) \propto \left(d_n(u)\right)^{\alpha}$$

      - Where $\left(d_n(u)\right)$ is the degree of vertex u in the tree graph $T^{(\alpha)}(n)$
  - When $\alpha = 0$, we have a Uniform Attachment Model, when $\alpha = 1$, we have the original Preferential Attachment Model suggested by Price in the 70s and later by Barabasi and Albert in the 90s
  - Here, we assume that $T^{(0)}$ is a singleton.

2. PROBLEM STATEMENTS



- Given a tree graph $T(n)$, we would like to recover the seed graphs:
  - The very first vertex, vertex $\mathbf{1}$
  - The first L vertices

- Objectives:
  - Algorithms
  - Accuracy & Precision
  - Complexity

3. RECOVERING VERTEX 1

- Main paper, from which the algorithm and methodology took inspiration:
  - Sebastien Bubeck, Luc Devroye, and Gabor Lugosi. Finding Adam in random growing trees. *Random Structures Algorithms*, 50(2):158{172, Nov 2016.

- Algorithm:
  - Look at the largest connected component after removing each vertex

---

Input: an unlabeled tree graph $T^0(n)$

For each $u \in V\big(T^0(n)\big)$ do:

  For each $v \in N(u)$ do:

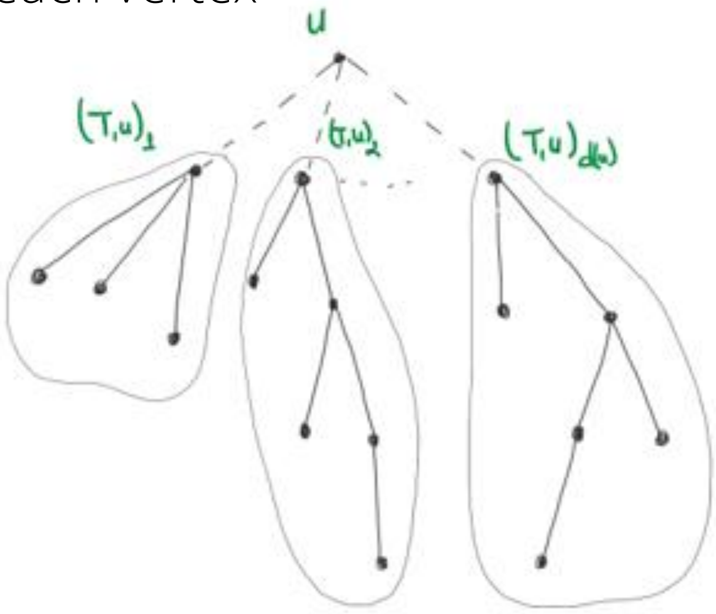
    Compute $|(T,u)_{v,\downarrow}|$

  End for

  Return $\varphi(u) = \max\limits_{v \in N(u)} |(T,u)_{v\downarrow}|$

End for

Output: K vertices with the smallest φ values



3. RECOVERING VERTEX 1

THEORETICAL RESULT

- (Bubeck et. al.) **Theorem 6:**

- Let $T^0(n)$ be a tree graph grown under the PA mdel. Let $\varepsilon > 0$ be given, let
  $K \geq c \, \frac{\log^2(1/\varepsilon)}{\varepsilon^4}$ for some positive constant c, then

$$\liminf_{n \to \infty} P(1 \in H_\psi(T^0(n))) \geq 1 - \varepsilon$$

(Asymptotically, we are guaranteed of the accuracy)

3. RECOVERING VERTEX 1

SIMULATION RESULT

- K vertices returned:

$$K \in [10] \cup \{5k, k \in [20]\}$$

- Graph sizes:

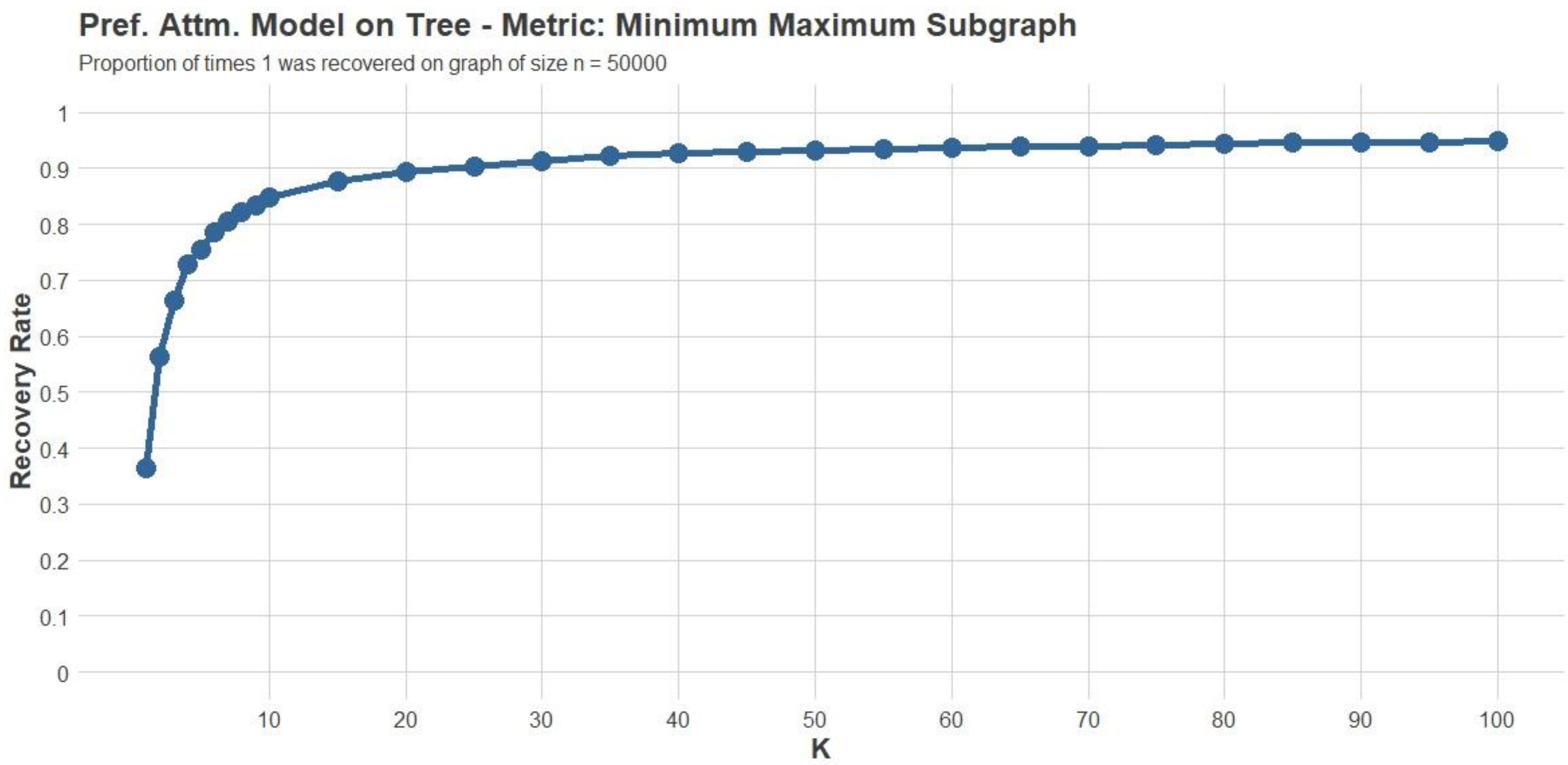
$$n \in \{5{,}000k, k \in [15]\}$$

- For each graph size, create **1000** Monte Carlo simulations

- Remark:
  - due to computational time, only compute φ values of the first 5,000 vertices:

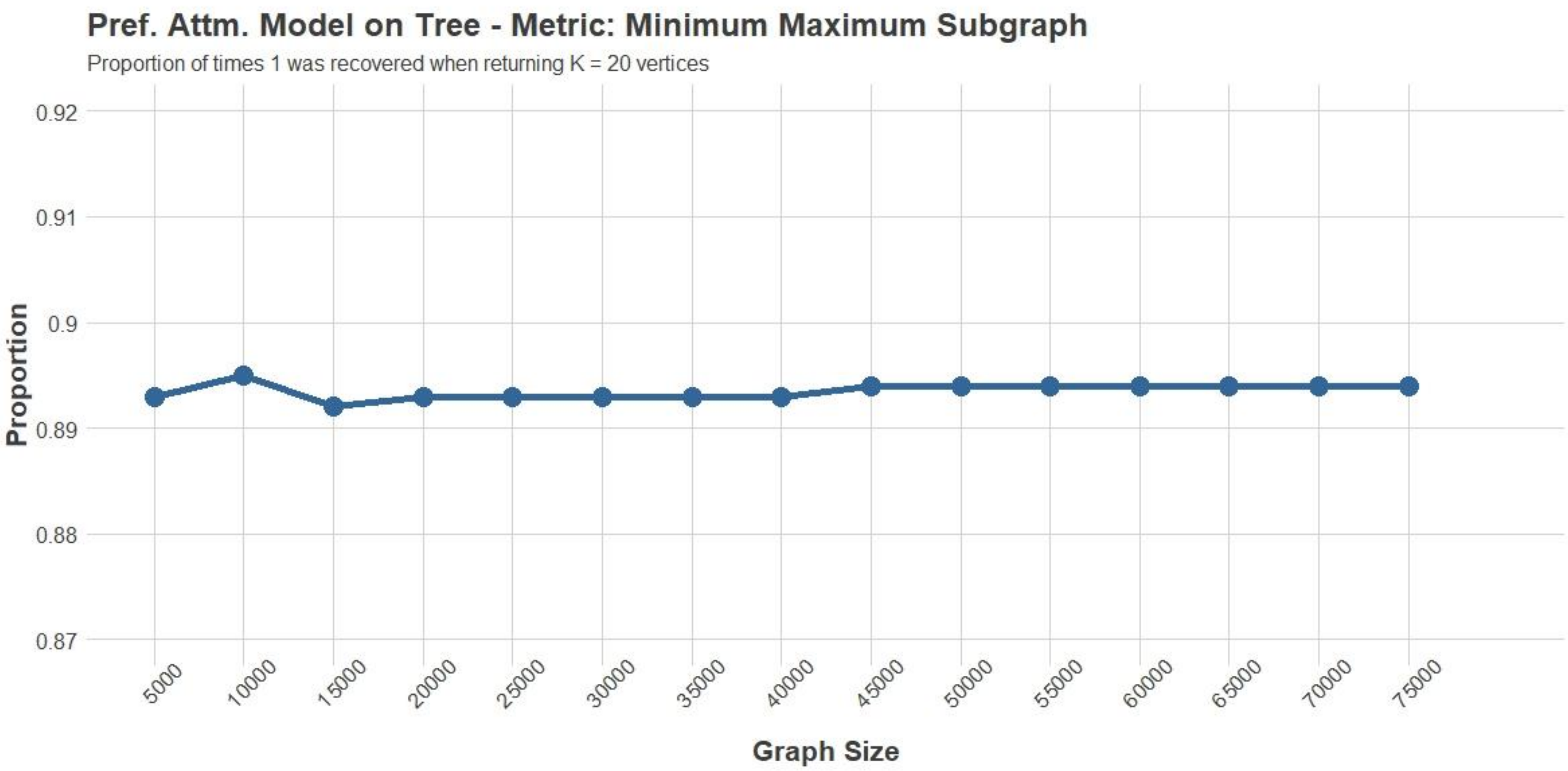
3. RECOVERING VERTEX 1

SIMULATION RESULT: FIXING GRAPH SIZE



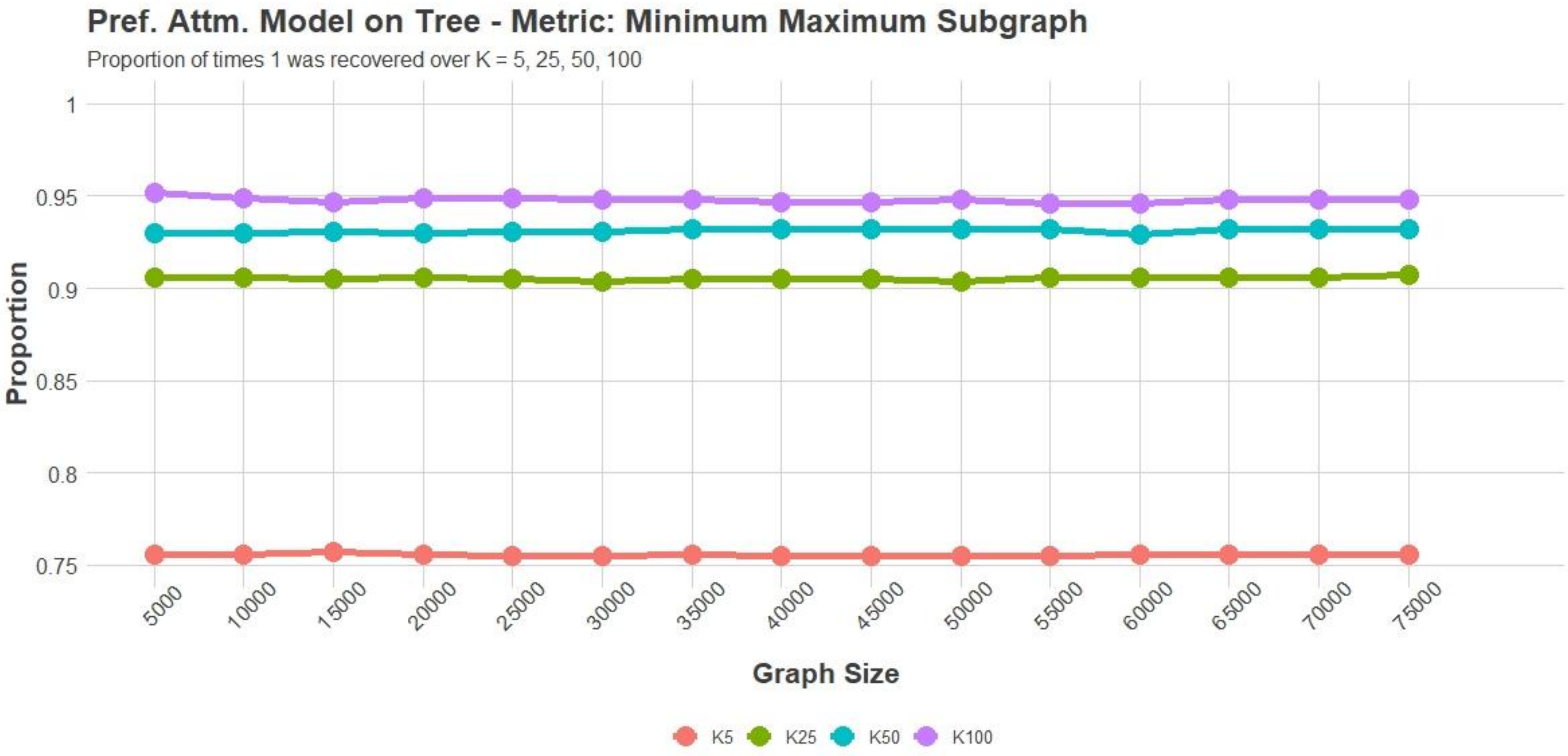
3. RECOVERING VERTEX 1

SIMULATION RESULT: FIXING K (NUMBER OF VERTICES RETURNED)



3. RECOVERING VERTEX 1

SIMULATION RESULT: DIFFERENT GRAPH SIZES & K

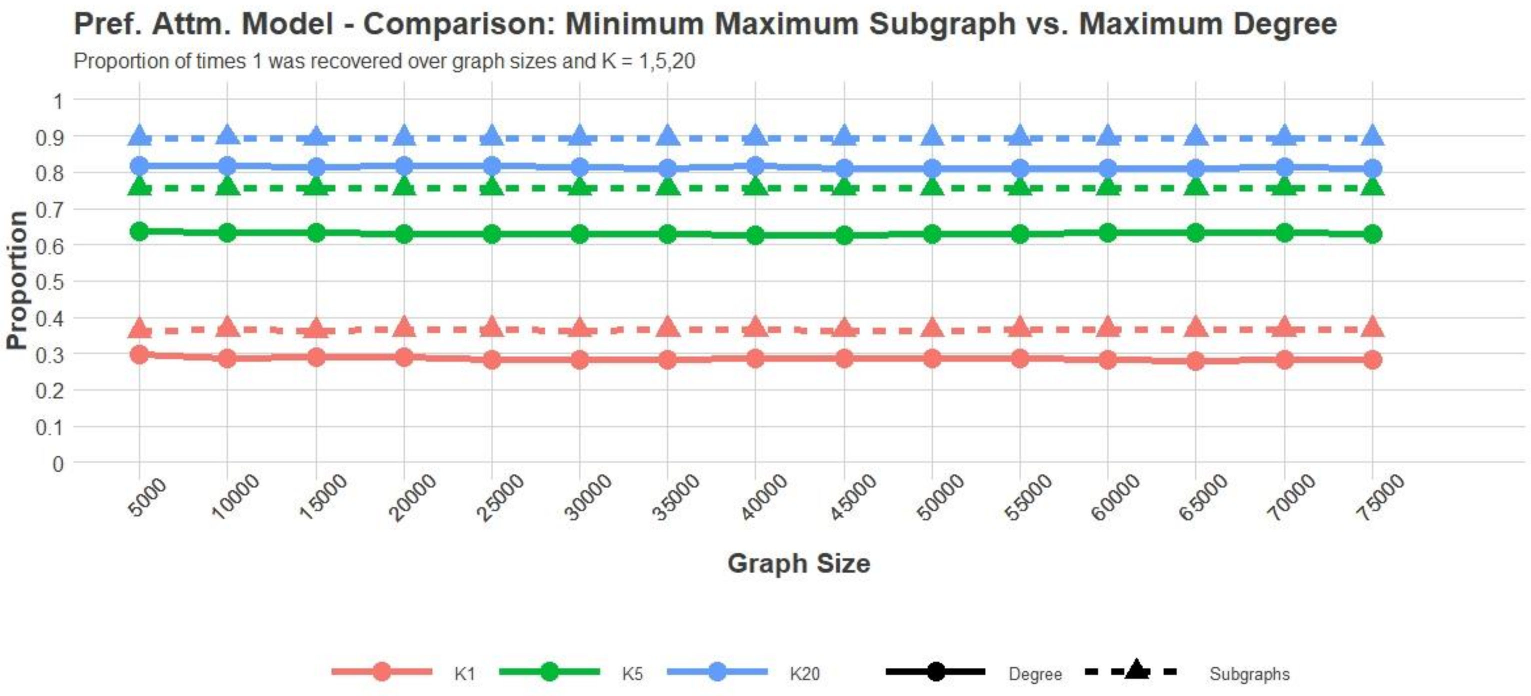


3. RECOVERING VERTEX 1

SIMULATION RESULT

- Although highly accurate, computing φ was also highly expensive.

- Naïve approach:
  - Returning vertices with the largest **vertex degree.**



3. RECOVERING VERTEX 1

PROPOSED ALGORITHM

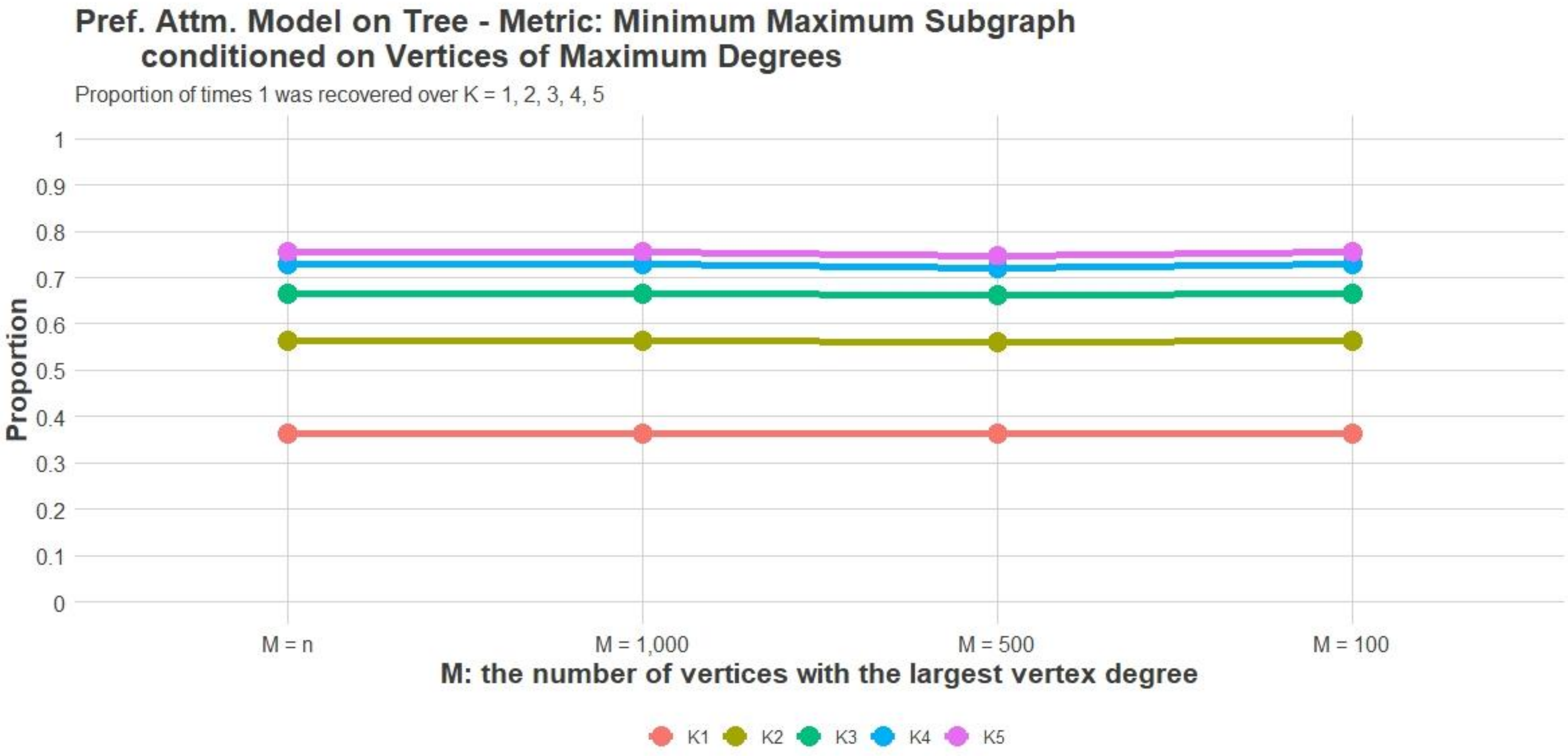
- Algorithm φ + vertex degree:
  - Narrow down the most probable vertices before computing $\boldsymbol{\varphi}$

Input: an unlabeled tree graph $T^0(n)$
Step 1: Return M vertices with the maximum degrees
Step 2: Compute φ values among those M vertices
Output: K vertices with the smallest φ values

- Goal:
  - Reduce complexity while giving up little (to no) loss in accuracy

3. RECOVERING VERTEX 1

SIMULATION RESULT: PROPOSED ALGORITHM



4. RECOVERING FIRST L VERTICES

- Sequential Local Search Algorithm:
  - Sequentially Search and Update over stages

Input: an unlabeled tree graph $T^0(n)$
Initialize a list $M = \emptyset$ by default
For each $l \in 1{:}L$ do
    If $M = \emptyset$ then
        Return K vertices with the smallest φ values
    Else
        Initialize X containing neighbors of all vertices in M
        Return from X K vertices with the smallest φ values
    End if
    Update M with those K vertices returned
Output: list M

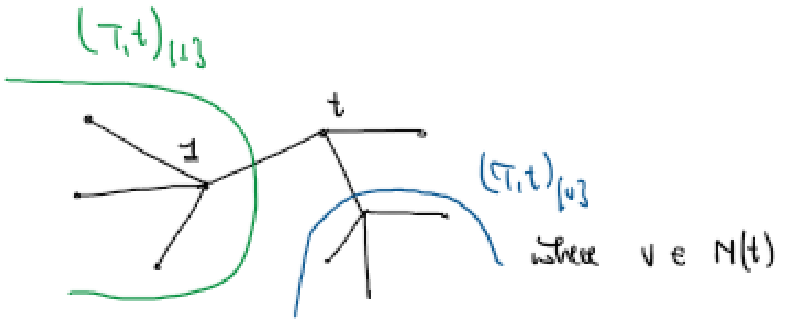
---

4. RECOVERING FIRST L VERTICES

CONJECTURES

- Conjecture 1:
  - In a tree graph $T(n)$, let $t > 1$ be any vertex, then

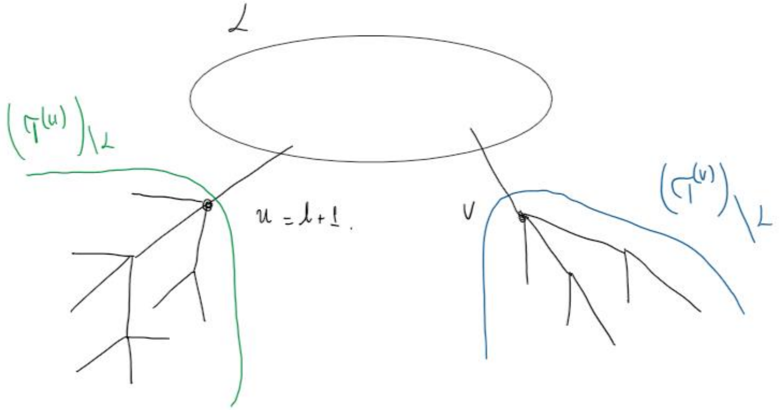
  $\forall v \in N(t)$:
  $$\lim_{n\to\infty} E\left[\left|(T,t)_{\{1\}}\right|\right] \geq \lim_{n\to\infty} E\left[\left|(T,t)_{\{v\}}\right|\right]$$



$(T,t)_{\{1\}}$ is the subtree starting at a child of t that contains vertex $\mathbf{1}$

- Conjecture 2:
  - In a tree graph $T(n)$, let $L = [l]$ where l is given, let $u = l + 1$ be the vertex appearing after the first l vertices, then

  $\forall v \in N(L)$:
  $$\lim_{n\to\infty} E\left[\left|\left(T^{(u)}\right)_{\backslash \mathrm{L}}\right|\right] \geq \lim_{n\to\infty} E\left[\left|\left(T^{(v)}\right)_{\backslash \mathrm{L}}\right|\right]$$



$\left(T^{(u)}\right)_{\backslash \mathrm{L}}$ is the subtree starting at u that does not contain L

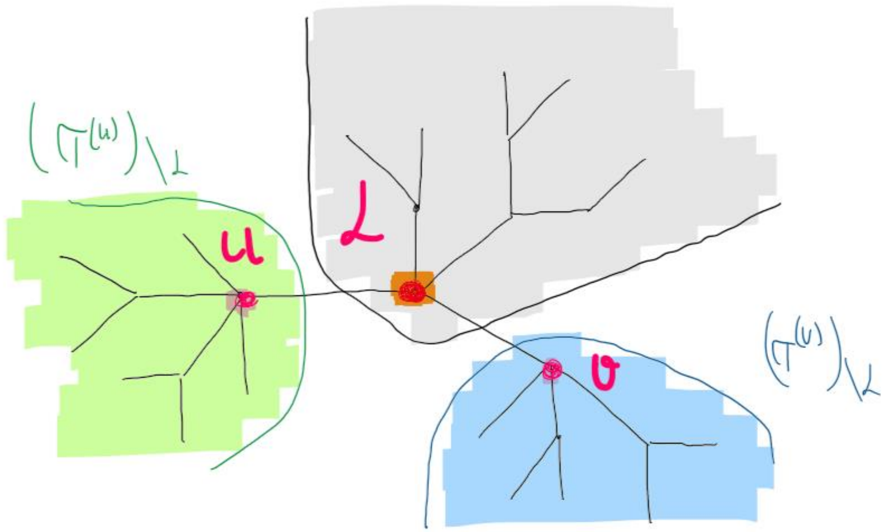
4. RECOVERING FIRST L VERTICES

CONJECTURES

- Implications in this context: suppose that we know the first l vertices, we want to recover vertex $\mathrm{u} = l + 1$ among $N(L)$, where L is the set of those l vertices.

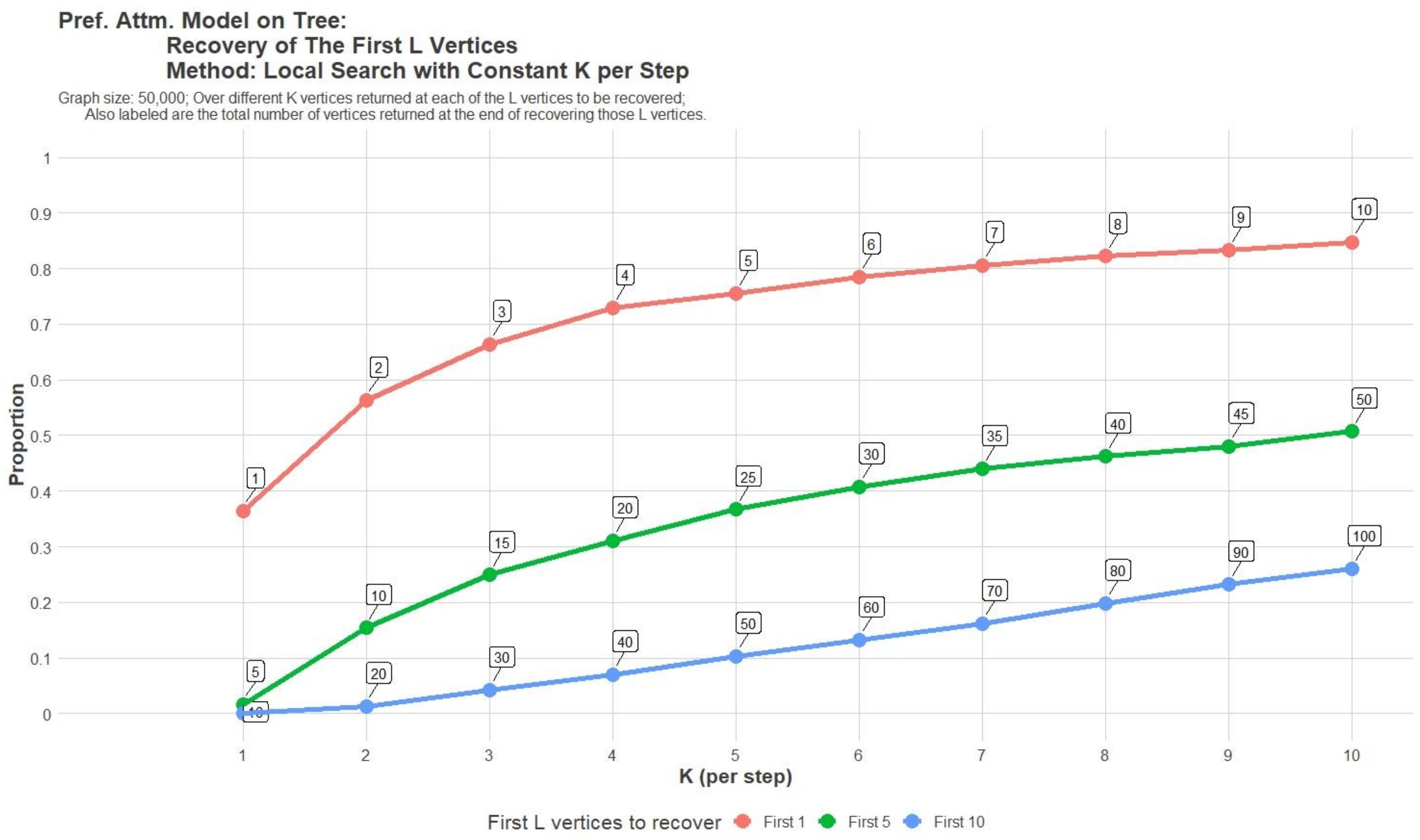
- Conjecture $\mathbf{1}$:
  - As $n \to \infty$, computing $\varphi(u)$ reduces to computing the size of the connected component containing L.

- Conjecture $\mathbf{2}$:
  - As $n \to \infty$, for all $v \in N(L)$, $$\left|\left(T^{(u)}\right)_{\setminus L}\right| > \left|\left(T^{(v)}\right)_{\setminus L}\right|$$ which implies that $\varphi(u) < \varphi(v)$.



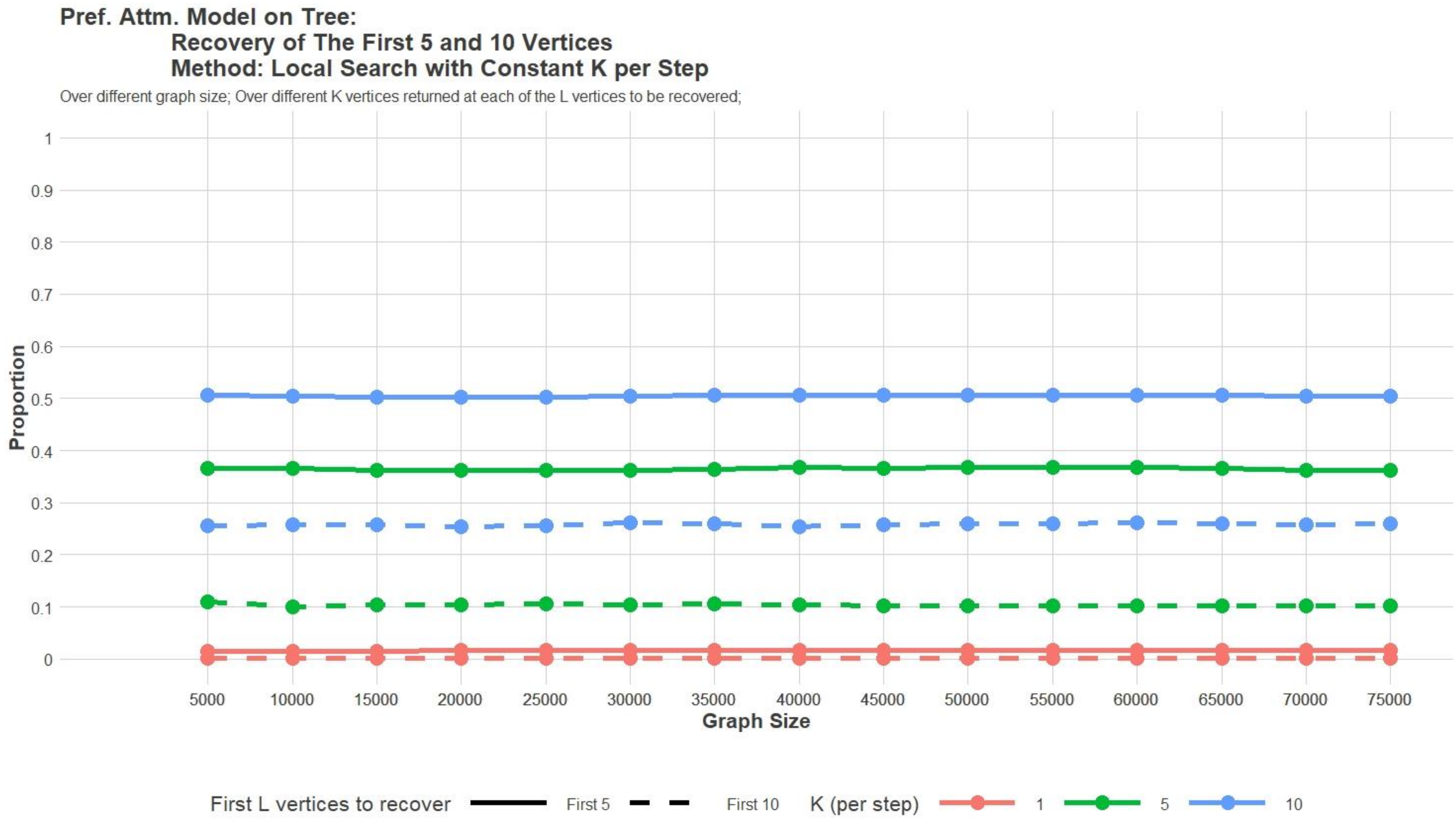
4. RECOVERING FIRST L VERTICES

SIMULATION RESULT: FIXING GRAPH SIZE



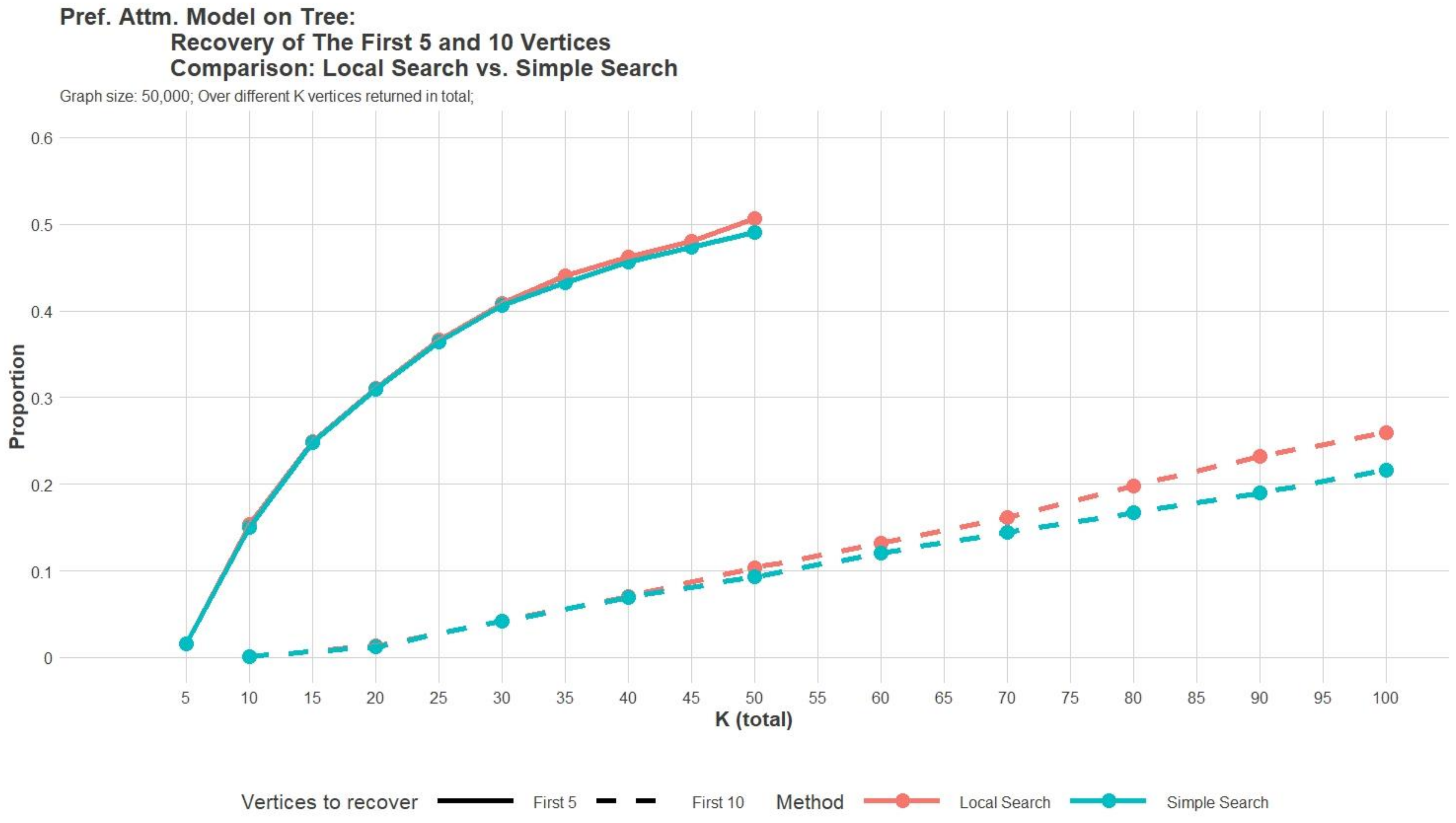
4. RECOVERING FIRST L VERTICES

SIMULATION RESULT: DIFFERENT GRAPH SIZES



4. RECOVERING FIRST L VERTICES

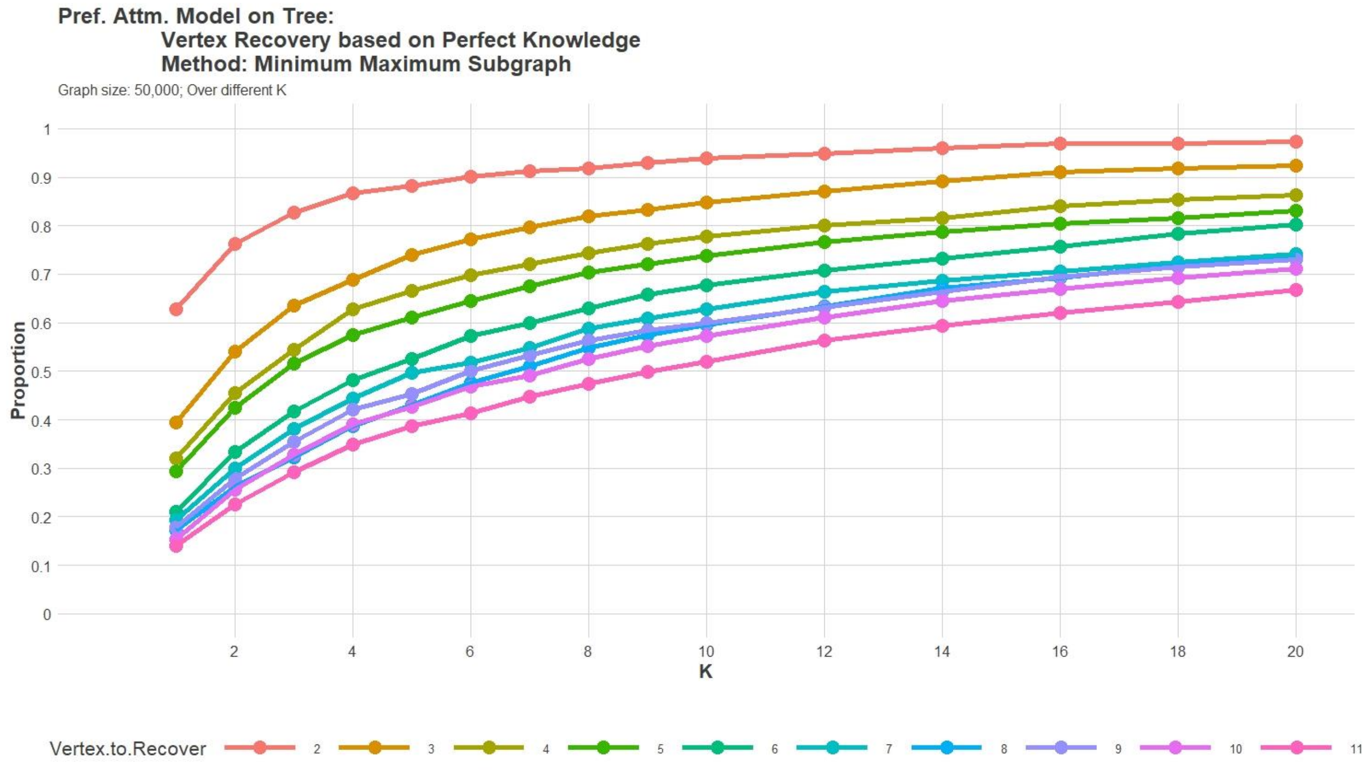
SIMULATION RESULT: COMPARISON: SEQUENTIAL VS. SIMPLE SEARCHES



4. RECOVERING FIRST L VERTICES

A SECOND LOOK

- **The problem becomes much harder quickly.**
- Even with perfect knowledge (knowing exactly all the first l vertices), recovering vertex $l+1$ is much harder for even very small l.



THANK YOU FOR YOUR ATTENTION!

Math 199 – Thesis Presentation

Thu Nguyen

Advisor: Professor Ery Arias-Castro

Recovering the First Vertices in a preferential attachment model

APPENDIX

1. Accuracy & Precision
2. Algorithm complexity
   - Vertex degree & φ value
3. Problem difficulty
4. Works in progress

1. ACCURACY & PRECISION

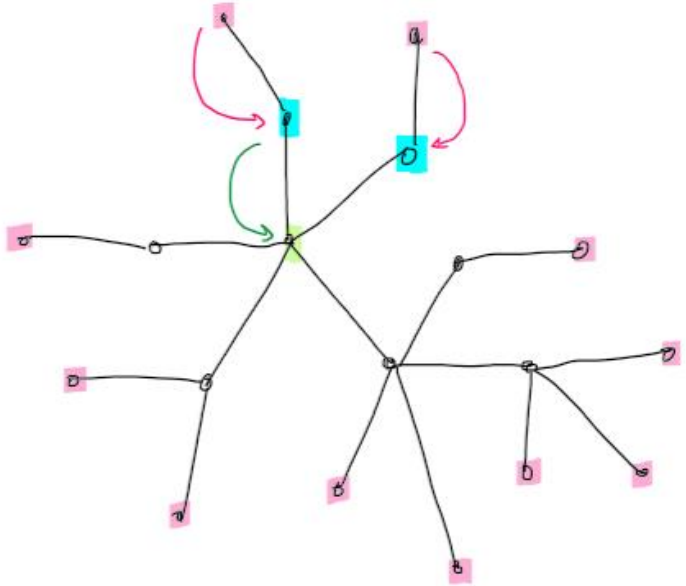
$$Accuracy = \frac{\#(sim\colon all\ l\ recovered)}{\#(simulations)}$$

$$Precision = \frac{l}{K}$$

Recovering	Accuracy	Precision
Vertex 1	$.35$	1
First 5	$.25$	$1/3$
First 10	$.25$	$1/10$
Vertex 1	$.75$	$.5$
First 5	$.38$	$.5$
First 10	$.1$	$.5$

2. ALGORITHM COMPLEXITY

- Vertex degree
  - Incidence list: $O(n)$ (in linear time or less)
- Vertex φ value
  - Incidence list: $O(n^2)$
    - For each vertex, do Depth First Search $O(2n)$
  - Iteratively counting from leaves
    - Might be able to get to linear time if the tree graphs satisfy certain structures
    - Currently working on Necessary Conditions if complexity is $O(n)$ on entire graph



3. PROBLEM DIFFICULTY

- (Lugosi & Pereira) **Theorem: Uniform Attachment Model on Tree, $G^{(0)}$ is a Path**

Let $\varepsilon \in (0, e^{-e^2})$. Let T_n be a UA tree with seed $S_l = P_l$ (path of length l) for

$$l \leq \frac{\log(1/\varepsilon)}{\log\log(1/\varepsilon)}$$

Then for any $n \geq 2l$ (n is the tree graph size), any seed-finding algorithm that outputs a vertex set H_n of size l has

$$P\left(|H_n \cap P_l| \leq \frac{l}{2} \right) \geq \varepsilon$$

- That is, we are going to miss at least half of the first l vertices with probability at least ε

4. WORKS IN PROGRESS

1. The trade-off between Accuracy & Precision
2. In recovering the first l vertices:
   - Sequential Local Search Algorithm:
     - treat K (the number of vertices returned per stage) as a parameter
3. The complexity of computing φ values
   - Alternative approaches besides traditional approaches
4. Conclusively proving the 2 conjectures