Hindawi Publishing Corporation
Journal of Applied Mathematics
Volume 2013, Article ID 743908, 11 pages
http://dx.doi.org/10.1155/2013/743908

Research Article

On the 2-MRS Problem in a Tree with Unreliable Edges

Wei Ding,[1] Yu Zhou,[2] Guangting Chen,[3] Hongfa Wang,[1] and Guangming Wang[2]

[1] *Zhejiang University of Water Resources and Electric Power, Hangzhou, Zhejiang 310018, China*
[2] *School of Science, Hangzhou Dianzi University, Hangzhou, Zhejiang 310018, China*
[3] *Institute of Operational Research and Cybernetics, Hangzhou Dianzi University, Hangzhou, Zhejiang 310018, China*

Correspondence should be addressed to Wei Ding; dingweicumt@163.com

Received 27 June 2013; Revised 5 September 2013; Accepted 6 September 2013

Academic Editor: Han H. Choi

This paper extends the well-known most reliable source (1-MRS) problem in unreliable graphs to the 2-most reliable source (2-MRS) problem. Two kinds of reachable probability models of node pair in unreliable graphs are considered, that is, the superior probability and united probability. The 2-MRS problem aims to find a node pair in the graph from which the expected number of reachable nodes or the minimum reachability is maximized. It has many important applications in large-scale unreliable computer or communication networks. The #P-hardness of the 2-MRS problem in general graphs follows directly from that of the 1-MRS problem. This paper deals with four models of the 2-MRS problem in unreliable trees where every edge has an independent working probability and devises a cubic-time and quadratic-space dynamic programming algorithm, respectively, for each model.

1. Introduction

A computer network or communication network is commonly denoted by an undirected graph $G = (V, E)$, where V is the node set, each of which represents a processing or switching element, and E is the edge set, each of which represents a communication link [1, 2]. Given any two different nodes u and v, the communication between u and v is achieved by a u-to-v path. Network failures may frequently happen to links or nodes [3, 4]. Such a network is called *unreliable* or *probabilistic*. In the past decade, a large number of network reliability problems have been extensively studied [5–20]. Many of them can be reduced to the problem of finding an optimal location for placing a server in an unreliable network [7–12, 14, 15, 17–20]. In addition, the most reliable route problem has been studied in [9, 14, 19] and its delay-constrained version has been studied in [21]. Recently, the problem of placing servers [22] and the problem of assigning links [23] in probabilistic wireless networks have been considered. Also, the continuous data collection schemes have been proposed in probabilistic wireless sensor networks (WSNs) [24, 25].

Given an unreliable graph G with n nodes and m edges, we call v a *reachable* node of u (or call v reachable from u) if u can reach v correctly. The probability of successful communication from u to v is called the *reachable probability* of u to v. The *sum reachability* of a node refers to the expected number of reachable nodes from it. The node with a maximum sum reachability is called a *sum-max most reliable source* (Sum-Max 1-MRS) of the graph. The Sum-Max 1-MRS problem has caused a lot of researchers' interests; see [7–12, 14, 15, 17, 20]. The *minimum reachability* of a node refers to its minimum reachable probability to another node. The node maximizing its minimum reachability is called a *min-max most reliable source* (Min-Max 1-MRS) of the graph. The Min-Max 1-MRS problem has been studied in [9, 20]. The Sum-Max 1-MRS and Min-Max 1-MRS are collectively called a 1-MRS for short. In addition, we refer readers to [18, 19] for the *1-center problem* in unreliable graphs and related algorithms. Obviously, both 1-MRS and 1-center problems are a good location for placing a server in unreliable graphs.

It is well known that the 1-MRS problem and 1-center problem are both #P-hard in general graphs [3, 4, 18]. However, they are tractable under the most reliable route policy. Helander and Melachrinoudis presented a polynomial time algorithm [14], and Ding gave an $O(mn + n^2 \log n)$-time algorithm [9] for the 1-MRS problem. Santiváñez et al. designed a polynomial time algorithm for the 1-center problem [19].

Moreover, both the 1-MRS problem and 1-center problem are also tractable in several types of sparse networks. For tree graphs with unreliable edges, Melachrinoudis and Helander designed a quadratic time algorithm [15], and Xue designed a linear time algorithm for the 1-MRS problem [20]. Santivanez and Melachrinoudis gave a linear time algorithm for the 1-center problem [18]. Ding and Xue considered the 1-MRS problem in the tree graphs with unreliable nodes and devised a linear time algorithm using the complementary dynamic programming method [10]. For ring graphs, Ding gave a quadratic time algorithm [8]. For ring-tree graphs, Ding and Xue considered an underlying topology of a strip, presented a polynomial time divide-and-conquer algorithm [11], considered an underlying topology of a tree, and presented a fast parallel algorithm based on the complementary dynamic programming [12]. For series-parallel graphs, Colbourn and Xue devised a linear time dynamic programming algorithm [7].

As networks grow rapidly in size, they become increasingly vulnerable to failures. Therefore, a single server can no longer satisfy the requirement of service from the whole network. In this scenario, we suggest to place at least two servers on unreliable networks with a large size. The rest of the paper focuses on the case of placing two servers and extends the 1-MRS problem to the 2-MRS problem, including *sum-max 2-most reliable source* (Sum-Max 2-MRS) and *min-max 2-most reliable source* (Min-Max 2-MRS). Given any a node pair $\langle u, v\rangle$, the probability of u and v reaching another node w successfully is called the *reachable probability* of $\langle u, v\rangle$ to w. The paper considers two types of reachable probability models of node pair, that is, the *superior probability* and *united probability*, formally defined in Section 2.2. Under both probability models, a cubic-time and quadratic-space algorithm is presented, respectively, for finding a Sum-Max 2-MRS and a Min-Max 2-MRS on trees with unreliable edges. Note that this paper is the first one to propose and study the 2-MRS problem.

The remainder of this paper is organized as follows. In Section 2, we define notations and the 2-MRS problem formally and show several fundamental lemmas. We present a cubic-time algorithm, respectively, for the Sum-Max 2-MRS problem in Section 3 and the Min-Max 2-MRS problem in Section 4 on tree graphs with unreliable edges. In Section 5, we give an example for illustrating our algorithms. In Section 6, we conclude the paper with some future research topics.

2. Preliminaries

2.1. Notations. Let $\mathscr{G} = (\mathscr{V}, \mathscr{E}, p)$ be an undirected connected graph, where $\mathscr{V}$ is the node set, $\mathscr{E}$ is the edge set, and each edge $e \in \mathscr{E}$ has a weight $p(e)$ representing the working probability on e. Suppose that all edges have an independent working probability and all nodes are immune to failures. Let $\langle u, v\rangle$ be a node pair of $\mathscr{G}$. We use $\pi(u, v)$ to denote a simple path in $\mathscr{G}$ connecting u and v and also use $\pi(u, v)$ to denote the event that $\pi(u, v)$ works correctly for simplicity of presentation. Let $\Pr(\pi(u, v))$ denote the probability of $\pi(u, v)$ working correctly and $\Pr(u, v)$ the probability of u reaching v among $\mathscr{G}$. Specifically, $\Pr(\pi(u, v)) = p(e) = p(u, v)$ when $\pi(u, v)$ is just an edge $e = \{u, v\}$ in $\mathscr{G}$. Note that $\pi(u, v)$ works correctly if and only if all edges of $\pi(u, v)$ work correctly simultaneously. Let $\mathscr{V}(\pi(u, v))$ and $\mathscr{E}(\pi(u, v))$ denote the set of nodes and edges on $\pi(u, v)$, respectively. So,

$$\Pr(\pi(u, v)) = \prod_{e\in\mathscr{E}(\pi(u,v))} p(e). \tag{1}$$

Let $T = (V, E, p)$ be an undirected tree graph, where V is the node set, E is the edge set, and every edge $e \in E$ has a probability weight $p(e)$ as defined above. For any $\langle u, v\rangle$ of T, there exists a unique path $\pi(u, v)$ in T connecting u and v. Thus, it always holds that $\Pr(u, v) = \Pr(\pi(u, v))$ in T. Let $\Pr(u, u) = 1$ when $u = v$. An unrooted tree can be transformed into a rooted tree by designating any node as the root. Without any loss of generality, we pick out any node $u \in V$ and transform T into a tree rooted at u, denoted by $T_u = (V_u, E_u, p)$. Clearly, $V_u = V$ and $E_u = E$. For any $v \in V_u$, we use $C_u(v)$ to denote the set of the children of v in T_u and $T_u(v)$ to denote the subtree of T_u rooted at v. Let $V_u^{\alpha}(v)$ denote the set of nodes in $T_u(v)$ and $V_u^{\beta}(v)$ the set of nodes outside $T_u(v)$. Specifically, $C_u(v) = \emptyset$ and $V_u^{\alpha}(v) = \{v\}$ when v is a leaf of T_u. For any $v \in V_u$, we use $f_u(v)$ to denote the parent of v in T_u and $\mathscr{Q}_u(v)$ to denote the set of ancestors of v in T_u. Specifically, $\mathscr{Q}_u(v) = \{u\}$ when $v \in C_u(u)$ and $\mathscr{Q}_u(u) = \emptyset$. For any $w \in \mathscr{Q}_u(v)$, we use $s_u^v(w)$ to denote the child of w on $\pi(u, v)$ in T_u. Let $C_u^v(w)$ denote the set of children of w in T_u other than $s_u^v(w)$; that is, $C_u^v(w) = C_u(w) \setminus \{s_u^v(w)\}$. Let $H_u = \max_{v\in V_u} |\mathscr{Q}_u(v)|$. Suppose that u is located at the 0th level in T_u. So, T_u has $H_u + 1$ levels in total. Let $V_u(h)$, $h = 0, 1, 2, \ldots, H_u$, denote the set of nodes on the h-level of T_u. Also, we use D (resp., D_u) to denote the set of leaves of T (resp., T_u). Clearly, $|D_u|$ is equal to $|D| - 1$ if u is a leaf of T and $|D|$ if u is not a leaf.

2.2. Problem Statements. Given any $\langle u_i, u_j\rangle$ of $\mathscr{G}$ and any $v \in \mathscr{V}$, the maximum in the reachable probability of u_i to v and that of u_j to v is called the *superior probability* of $\langle u_i, u_j\rangle$ to v, denoted by $\mathscr{F}_1(u_i, u_j; v)$. The probability of u_i to v, u_j to v, or both is called the *united probability* of $\langle u_i, u_j\rangle$ to v, denoted by $\mathscr{F}_2(u_i, u_j; v)$. The superior probability and united probability are collectively called the *reachable probability* of node pair.

Problem 1. Given an undirected connected network $\mathscr{G} = (\mathscr{V}, \mathscr{E}, p)$, where every edge $e \in \mathscr{E}$ has a weight representing the working probability $p(e)$ on e, the objective is to find a node pair in $\mathscr{G}$ such that the *sum reachability* (resp., *minimum reachability*) of the node pair is maximized.

The *sum reachability* of $\langle u_i, u_j\rangle$ is referred to as the expected number of reachable nodes in $\mathscr{G}$ from $\langle u_i, u_j\rangle$, denoted by $\mathbb{E}_\lambda[u_i, u_j]$. The optimal solution of Problem 1 with the objective of maximizing the sum reachability of node pair is called *Sum-Max 2-MRS* of $\mathscr{G}$, denoted by $\langle u_i^*, u_j^*\rangle$. We have

$$\mathbb{E}_\lambda\left[u_i, u_j\right] = \sum_{v\in\mathscr{V}} \mathscr{F}_\lambda\left(u_i, u_j; v\right), \quad \lambda = 1, 2, \tag{2}$$

$$\mathbb{E}_\lambda\left[u_i^*, u_j^*\right] = \max_{u_i,u_j\in\mathscr{V}, u_i\neq u_j} \mathbb{E}_\lambda\left[u_i, u_j\right]. \tag{3}$$

The *minimum reachability* of $\langle u_i, u_j\rangle$ is referred to as the minimum reachable probability of $\langle u_i, u_j\rangle$, denoted by $\mathbb{M}_\lambda[u_i, u_j]$. The optimal solution of Problem 1 with the aim of maximizing the minimum reachability of node pair is called *Min-Max 2-MRS* of $\mathscr{G}$, denoted by $\langle u_i^\star, u_j^\star\rangle$. We have

$$\mathbb{M}_\lambda\left[u_i, u_j\right] = \min_{v\in\mathscr{V}} \mathscr{F}_\lambda\left(u_i, u_j; v\right), \quad \lambda = 1, 2, \tag{4}$$

$$\mathbb{M}_\lambda\left[u_i^\star, u_j^\star\right] = \max_{u_i,u_j\in\mathscr{V}, u_i\neq u_j} \mathbb{M}_\lambda\left[u_i, u_j\right]. \tag{5}$$

The Sum-Max 2-MRS problem and Min-Max 2-MRS problem are collectively called the *2-MRS problem*. Based on the #P-hardness of the 1-MRS problem in general graphs [3, 4], we conclude that the 2-MRS problem in general graphs is also #P-hard. However the 1-MRS problem in tree graphs is tractable [10, 15, 20]. In the remainder of this paper, we will deal with the 2-MRS problem in tree graphs. All the notations and their explanations used in the paper are listed in Table 2.

2.3. Fundamental Lemmas. Let $A \oplus B$ denote the union of two disjoint sets A and B. Lemma 2 shows the decomposition scheme at u_j of V_{u_i} for any $\langle u_i, u_j\rangle$ of T, see Figure 1. The proof is straightforward and omitted here.

Lemma 2. *Given any $\langle u_i, u_j\rangle$ of T, one has*

$$V_{u_i} = V_{u_i}^\alpha\left(u_j\right) \oplus V_{u_i}^\beta\left(u_j\right), \tag{6}$$

in which

$$V_{u_i}^\alpha\left(u_j\right) = \left(\bigoplus_{s\in C_{u_i}(u_j)} V_{u_i}^\alpha(s)\right) \oplus \left\{u_j\right\}, \tag{7}$$

$$V_{u_i}^\beta\left(u_j\right) = \left(\bigoplus_{w\in\mathscr{Q}_{u_i}(u_j)} \bigoplus_{s\in C_{u_i}^{u_j}(w)} V_{u_i}^\alpha(s)\right) \oplus \mathscr{Q}_{u_i}\left(u_j\right). \tag{8}$$

Lemma 3. *Given any tree $T_u = (V_u, E_u)$ rooted at u, one has,*

$$\sum_{v\in V_u} \left|\mathscr{Q}_u(v)\right| \leq \frac{1}{2}\left|V_u\right|\left(\left|V_u\right| - 1\right). \tag{9}$$

Proof. Let $T_u = (V_u, E_u)$ be a tree rooted at u with an arbitrary topology and $T_u^\Delta = (V_u^\Delta, E_u^\Delta)$ a line with the same number of nodes as T_u. First, we prove that $\sum_{v\in V_u} |\mathscr{Q}_u(v)| \leq \sum_{v\in V_u^\Delta} |\mathscr{Q}_u^\Delta(v)|$. In fact, T_u can always be derived from T_u^Δ in the following way. Let L denote the current line and set $L = T_u^\Delta$ initially. We take away the lowest node of L and attach it to another node one by one. Every time we take away the lowest node of L, we set the rest of L to the current line. So, we are sure to obtain a series of trees rooted at u. Suppose that we get m trees, $T_u^1, T_u^2, \ldots, T_u^m$ in order. Let $T_u^0 = T_u^\Delta$ and $a_k = \sum_{v\in V_u} |\mathscr{Q}_u^k(v)|$, $k = 0, 1, \ldots, m$. Given any $0 \leq k \leq m-1$, T_u^{k+1} is derived from T_u^k by moving the lowest node v' of L in T_u^k. The new level in T_u^{k+1} at which v' is located is lower than the previous level in T_u^k. So, $a_{k+1} \leq a_k$. Therefore, $a_0 \geq a_1 \geq \cdots \geq a_m$. Note that $H_u^\Delta = |V_u^\Delta| - 1$ and $|V_u^\Delta(h)| = 1$, $h = 0, 1, \ldots, H_u^\Delta$. We have

$$a_0 = \sum_{v\in V_u^\Delta} \left|\mathscr{Q}_u^\Delta(v)\right| = \sum_{h=1}^{|V_u^\Delta|-1} h = \frac{1}{2}\left|V_u^\Delta\right|\left(\left|V_u^\Delta\right| - 1\right). \tag{10}$$

Therefore, $\sum_{v\in V_u} |\mathscr{Q}_u(v)| \leq (1/2)|V_u|(|V_u| - 1)$ for any T_u. □

Lemma 4. *Given any tree $T_u = (V_u, E_u)$ rooted at u, one has,*

$$\sum_{v\in V_u} \left|C_u(v)\right| = \left|V_u\right| - 1, \tag{11}$$

$$\frac{\left(\left|V_u\right| - 1\right)^2}{\left|V_u \setminus D_u\right|} \leq \sum_{v\in V_u} \left|C_u(v)\right|^2 \leq \left(\left|V_u\right| - 1\right)^2. \tag{12}$$

Proof. It follows directly from $C_u(v) = \emptyset$, for all $v \in D_u$ and $|E_u| = |V_u| - 1$, that $\sum_{v\in V_u} |C_u(v)| = \sum_{v\in V_u\setminus D_u} |C_u(v)| = |E_u| = |V_u| - 1$.

On one hand,

$$\begin{aligned}\sum_{v\in V_u} \left|C_u(v)\right|^2 &= \sum_{v\in V_u\setminus D_u} \left|C_u(v)\right|^2 \geq \frac{\left(\sum_{v\in V_u\setminus D_u} \left|C_u(v)\right|\right)^2}{\left|V_u \setminus D_u\right|} \\ &= \frac{\left(\left|V_u\right| - 1\right)^2}{\left|V_u \setminus D_u\right|},\end{aligned} \tag{13}$$

where the equality holds if and only if all $|C_u(v)|$, $v \in V_u \setminus D_u$, are equal. Let $T_u^\circ = (V_u^\circ, E_u^\circ)$ be a star rooted at u, say, a special case of T_u. On the other hand, we prove that $\sum_{v\in V_u} |C_u(v)|^2 \leq \sum_{v\in V_u^\circ} |C_u^\circ(v)|^2$ for any T_u. Let $m = |V_u \setminus D_u|$ and $n = |V_u| - 1$. We label all the nodes in $V_u \setminus \{u\}$ by numbers $1, 2, \ldots, n$. Let $x_k = |C_u(v_k)|$, $k = 1, \ldots, m$. Then we build the following restricted optimization problem:

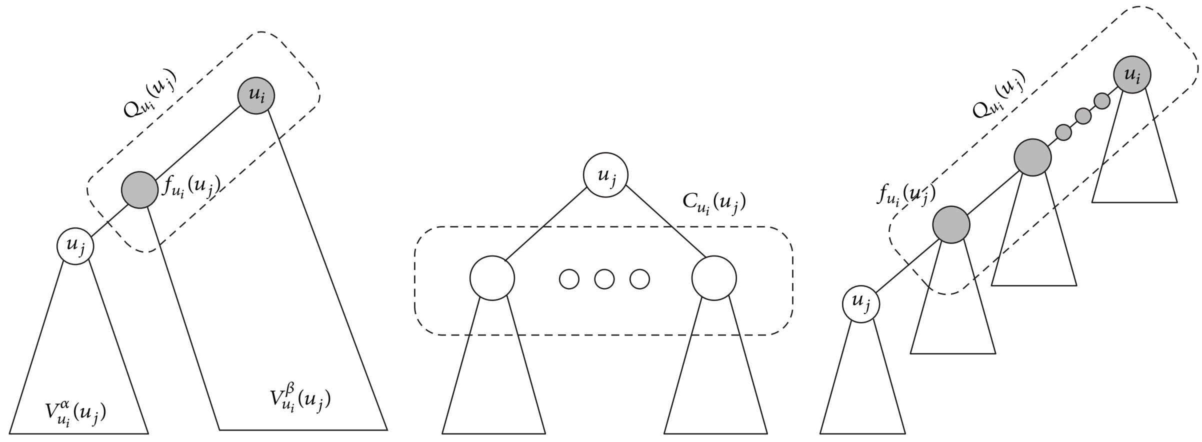
$$\begin{aligned}\max \quad & \sum_{k=1}^m x_k^2, \\ \text{s.t.} \quad & \sum_{k=1}^m x_k = n, \\ & x_k \geq 0, \quad x_k \in \mathbb{Z}, \quad k = 1, \ldots, m.\end{aligned} \tag{14}$$

Without loss of generality, we suppose that $\max\{x_1, x_2, \ldots, x_m\} = x_m$. By taking $x_m = n - \sum_{k=1}^{m-1} x_k$ into the above objective function, we obtain a new unrestricted optimization problem

$$\max_{x_k\geq 0, x_k\in\mathbb{Z}} f\left(x_1, \ldots, x_{m-1}\right) = \sum_{k=1}^{m-1} x_k^2 + \left(n - \sum_{k=1}^{m-1} x_k\right)^2. \tag{15}$$

We conclude that, for every $k = 1, \ldots, m-1$,

$$\frac{\partial f}{\partial x_k} = 2x_k - 2\left(n - \sum_{k=1}^{m-1} x_k\right) = 2\left(x_k - x_m\right) \leq 0. \tag{16}$$



(a) Decomposition of T_{ui} (b) Decomposition of $V^{\alpha}_{u_i}(u_j)$ (c) Decomposition of $V^{\beta}_{u_i}(u_j)$

FIGURE 1: Illustration of the decomposition of T_{u_i} for any $u_i \in V$.

Thus, $f_{\max}(x_1, \dots, x_{m-1}) = f(0, \dots, 0) = n^2$ and $x_m = n$. This implies that $\sum_{v \in V_u} |C_u(v)|^2 \le (|V_u| - 1)^2$ and the tree rooted at u satisfying the equality is just T_u°. □

Lemma 5. *Given any $\langle u_i, u_j \rangle$ of T, one has*

$$\Pr(u_i, u_j) = \Pr(u_i, v) \Pr(v, u_j), \quad \forall v \in V(\pi(u_i, u_j)). \tag{17}$$

Proof. First of all, for any two edge-disjoint paths π_1 and π_2 on T, we prove that π_1 and π_2 are independent. Suppose that π_1 contains edges $e_1, e_2, \dots, e_{k_1}$ and π_2 contains edges $e'_1, e'_2, \dots, e'_{k_2}$. It is obvious that $\{e_1, e_2, \dots, e_{k_1}\} \cap \{e'_1, e'_2, \dots, e'_{k_2}\} = \emptyset$. Since all of edges $e \in E$ are independent, we conclude that

$$\Pr(\pi_1 \pi_2) = \prod_{i_1=1}^{k_1} \Pr(e_{i_1}) \prod_{i_2=1}^{k_2} \Pr(e'_{i_2}) = \Pr(\pi_1) \Pr(\pi_2). \tag{18}$$

For any $v \in V(\pi(u_i, u_j))$, $\pi(u_i, u_j)$ comprises two edge-disjoint subpaths $\pi(u_i, v)$ and $\pi(v, u_j)$. Therefore, $\Pr(u_i, u_j) = \Pr(u_i, v)\Pr(v, u_j)$. □

Lemma 6. *Given any $\langle u_i, u_j \rangle$ of T and $v \in V_{u_i}$,*

(i) *if $v \in \mathcal{Q}_{u_i}(u_j)$, then*

$$\mathcal{F}_1(u_i, u_j; v) = \max\{\Pr(u_i, v), \Pr(u_j, v)\}, \tag{19}$$

$$\mathcal{F}_2(u_i, u_j; v) = \Pr(u_i, v) + \Pr(u_j, v) - \Pr(u_i, v)\Pr(u_j, v), \tag{20}$$

(ii) *if $v \in V^{\alpha}_{u_i}(u_j)$, then*

$$\mathcal{F}_\lambda(u_i, u_j; v) = \Pr(u_j, v), \quad \lambda = 1, 2, \tag{21}$$

(iii) *if $v \in V^{\alpha}_{u_i}(s)$, where $s \in C^{u_j}_{u_i}(w)$ and $w \in \mathcal{Q}_{u_i}(u_j)$, then*

$$\mathcal{F}_\lambda(u_i, u_j; v) = \mathcal{F}_\lambda(u_i, u_j; w)\, p(w, s) \Pr(s, v), \quad \lambda = 1, 2. \tag{22}$$

Proof. (i) When $v \in \mathcal{Q}_{u_i}(u_j)$, it follows directly from the definition of $\mathcal{F}_1(u_i, u_j; v)$ that $\mathcal{F}_1(u_i, u_j; v) = \max\{\Pr(u_i, v), \Pr(u_j, v)\}$. We see that $\pi(u_i, u_j)$ can be partitioned at v into two edge-disjoint subpaths $\pi(u_i, v)$ and $\pi(u_j, v)$. Lemma 5 implies that $\pi(u_i, v)$ and $\pi(u_j, v)$ are independent. The definition of $\mathcal{F}_2(u_i, u_j; v)$ means that the value of $\mathcal{F}_2(u_i, u_j; v)$ is equal to the probability of $\pi(u_i, v) \cup \pi(u_j, v)$. So,

$$\begin{aligned}
\mathcal{F}_2&(u_i, u_j; v) \\
&= \Pr(\pi(u_i, v) \cup \pi(u_j, v)) \\
&= \Pr(\pi(u_i, v)) + \Pr(\pi(u_j, v)) \\
&\quad - \Pr(\pi(u_i, v) \cap \pi(u_j, v)) \\
&= \Pr(u_i, v) + \Pr(u_j, v) - \Pr(u_i, v)\Pr(u_j, v).
\end{aligned} \tag{23}$$

(ii) When $v \in V^{\alpha}_{u_i}(u_j)$, $\pi(u_i, v)$ is composed of two edge-disjoint subpaths $\pi(u_i, u_j)$ and $\pi(u_j, v)$. It follows from Lemma 5 that $\Pr(u_i, v) = \Pr(u_i, u_j)\Pr(u_j, v)$. So,

$$\mathcal{F}_1(u_i, u_j; v) = \max\{\Pr(u_i, u_j), 1\}\Pr(u_j, v) = \Pr(u_j, v). \tag{24}$$

We see that $\pi(u_i, v)$ works correctly if and only if $\pi(u_i, u_j)$ and $\pi(u_j, v)$ work correctly simultaneously. So, $\pi(u_i, v) = \pi(u_i, u_j) \cap \pi(u_j, v)$. Thus,

$$\begin{aligned}
\mathcal{F}_2(u_i, u_j; v) &= \Pr((\pi(u_i, u_j) \cap \pi(u_j, v)) \cup \pi(u_j, v)) \\
&= \Pr(u_j, v).
\end{aligned} \tag{25}$$

(iii) When $v \in V^{\alpha}_{u_i}(s)$, where $s \in C^{u_j}_{u_i}(w)$ and $w \in \mathcal{Q}_{u_i}(u_j)$, we observe that $\pi(u_k, v)$, $k = i, j$, consists of two edge-disjoint subpaths $\pi(u_k, w)$ and $\pi(w, v)$. Also, $\pi(w, v)$ comprises two edge-disjoint subpaths $\pi(w, s)$ and $\pi(s, v)$ as well. We derive that $\Pr(u_k, v) = \Pr(u_k, w)p(w, s)\Pr(s, v)$ from Lemma 5. So,

$$\begin{aligned}\mathscr{F}_1(u_i, u_j; v) &= \max\{\Pr(u_i, v), \Pr(u_j, v)\} \\ &= \max\{\Pr(u_i, w), \Pr(u_j, w)\}\, p(w, s)\Pr(s, v) \\ &= \mathscr{F}_1(u_i, u_j; w)\, p(w, s)\Pr(s, v).\end{aligned} \tag{26}$$

We see that $\pi(u_k, v)$, $k = i, j$, works correctly if and only if $\pi(u_k, w)$ and $\pi(w, v)$ work correctly simultaneously. So, $\pi(u_k, v) = \pi(u_k, w) \cap \pi(w, v)$. Thus,

$$\begin{aligned}&\mathscr{F}_2(u_i, u_j; v) \\ &\quad= \Pr((\pi(u_i, w) \cap \pi(w, v)) \cup (\pi(u_j, w) \cap \pi(w, v))) \\ &\quad= \Pr(\pi(u_i, w) \cap \pi(w, v)) + \Pr(\pi(u_j, w) \cap \pi(w, v)) \\ &\qquad- \Pr((\pi(u_i, w) \cap \pi(w, v)) \cap (\pi(u_j, w) \cap \pi(w, v))) \\ &\quad= \Pr(u_i, w)\Pr(w, v) + \Pr(u_j, w)\Pr(w, v) \\ &\qquad- \Pr(u_i, w)\Pr(u_j, w)\Pr(w, v) \\ &\quad= \mathscr{F}_2(u_i, u_j; w)\, p(w, u)\Pr(u, v).\end{aligned} \tag{27}$$

□

3. Algorithm for Finding a Sum-Max 2-MRS

Definition 7. Given $T = (V, E, p)$ and $\langle u_i, u_j\rangle$ of T, one lets $\mathbb{E}^{\lambda}_{u_i}[u_j]$, $\lambda = 1, 2$, denote the sum reachability in T_{u_i} of $\langle u_i, u_j\rangle$. In addition, one lets $\mathscr{X}^{\lambda}_{u_i}(u_j)$ (resp., $\mathscr{Y}^{\lambda}_{u_i}(u_j)$) denote the sum reachability in $V^{\alpha}_{u_i}(u_j)$ (resp., $V^{\beta}_{u_i}(u_j)$) of $\langle u_i, u_j\rangle$.

Theorem 8. *Given any $\langle u_i, u_j\rangle$ of T, if $u_j \in V_{u_i}$, then one gets*

$$\mathbb{E}^{\lambda}_{u_i}[u_j] = \mathscr{X}^{\lambda}_{u_i}(u_j) + \mathscr{Y}^{\lambda}_{u_i}(u_j), \quad \lambda = 1, 2. \tag{28}$$

Proof. It follows directly from the definition of $\mathbb{E}^{\lambda}_{u_i}[u_j]$ given in Definition 7 that $\mathbb{E}_{\lambda}[u_i, u_j] = \mathbb{E}^{\lambda}_{u_i}[u_j] = \mathbb{E}^{\lambda}_{u_j}[u_i]$. We further derive from (2) that $\mathbb{E}^{\lambda}_{u_i}[u_j] = \sum_{v\in V_{u_i}} \mathscr{F}_{\lambda}(u_i, u_j; v)$. By (6) in Lemma 2 together with the definitions of $\mathscr{X}^{\lambda}_{u_i}(u_j)$ and $\mathscr{Y}^{\lambda}_{u_i}(u_j)$, we conclude that

$$\begin{aligned}\mathbb{E}^{\lambda}_{u_i}[u_j] &= \sum_{v\in V^{\alpha}_{u_i}(u_j)} \mathscr{F}_{\lambda}(u_i, u_j; v) + \sum_{v\in V^{\beta}_{u_i}(u_j)} \mathscr{F}_{\lambda}(u_i, u_j; v) \\ &= \mathscr{X}^{\lambda}_{u_i}(u_j) + \mathscr{Y}^{\lambda}_{u_i}(u_j).\end{aligned} \tag{29}$$

□

Theorem 9. *Given any $\langle u_i, u_j\rangle$ of T, if $u_j \in V_{u_i}$, then one gets*

$$\mathscr{X}^{\lambda}_{u_i}(u_j) = 1 + \sum_{s\in C_{u_i}(u_j)} p(u_j, s)\, \mathscr{X}^{\lambda}_{u_i}(s), \tag{30}$$

$$\begin{aligned}\mathscr{Y}^{\lambda}_{u_i}(u_j) = \sum_{w\in \mathscr{Q}_{u_i}(u_j)} &\mathscr{F}_{\lambda}(u_i, u_j; w) \\ &\times \left(\mathscr{X}^{\lambda}_{u_i}(w) - p\left(w, s^{u_j}_{u_i}(w)\right) \mathscr{X}^{\lambda}_{u_i}\left(s^{u_j}_{u_i}(w)\right)\right).\end{aligned} \tag{31}$$

Proof. The combination of the definition of $\mathscr{X}^{\lambda}_{u_i}(u_j)$ and (21) in Lemma 6 yields that $\mathscr{X}^{\lambda}_{u_i}(u_j) = \sum_{v\in V^{\alpha}_{u_i}(u_j)} \Pr(u_j, v)$. According to (7) in Lemma 2, for any $v \in V^{\alpha}_{u_i}(u_j)$, it is obvious that $\Pr(u_j, u_j) = 1$ if $v = u_j$ and otherwise there must be a child s of u_j if $C_{u_i}(u_j) \neq \emptyset$ such that v belongs to $V^{\alpha}_{u_i}(s)$. By Lemma 5, we obtain $\Pr(u_j, v) = p(u_j, s)\Pr(s, v)$. Therefore, for all $u_j \in V_{u_i} \setminus \{u_i\}$, we have

$$\begin{aligned}\mathscr{X}^{\lambda}_{u_i}(u_j) &= 1 + \sum_{s\in C_{u_i}(u_j)} p(u_j, s) \sum_{v\in V^{\alpha}_{u_i}(s)} \Pr(s, v) \\ &= 1 + \sum_{s\in C_{u_i}(u_j)} p(u_j, s)\, \mathscr{X}^{\lambda}_{u_i}(s).\end{aligned} \tag{32}$$

The definition of $\mathscr{Y}^{\lambda}_{u_i}(u_j)$ means $\mathscr{Y}^{\lambda}_{u_i}(u_j) = \sum_{v\in V^{\beta}_{u_i}(u_j)} \mathscr{F}_{\lambda}(u_i, u_j; v)$. According to (8) in Lemma 2, for any $v \in V^{\beta}_{u_i}(u_j)$, we are sure that v is either in $\mathscr{Q}_{u_i}(u_j)$ or in $V^{\alpha}_{u_i}(s)$, where s is a child other than $s^{u_j}_{u_i}(w)$ of some node w in $\mathscr{Q}_{u_i}(u_j)$. We can use (19) when $\lambda = 1$ and (20) when $\lambda = 2$ to compute $\mathscr{F}_{\lambda}(u_i, u_j; v)$ if $v \in \mathscr{Q}_{u_i}(u_j)$ and use (22) otherwise. Thus, for all $u_j \in V_{u_i} \setminus \{u_i\}$, we have

$$\begin{aligned}&\mathscr{Y}^{\lambda}_{u_i}(u_j) \\ &\quad= \sum_{v\in \mathscr{Q}_{u_i}(u_j)} \mathscr{F}_{\lambda}(u_i, u_j; v) + \sum_{w\in \mathscr{Q}_{u_i}(u_j)} \mathscr{F}_{\lambda}(u_i, u_j; w) \\ &\qquad\times \sum_{s\in C^{u_j}_{u_i}(w)} p(w, s) \sum_{v\in V^{\alpha}_{u_i}(s)} \Pr(s, v) \\ &\quad= \sum_{w\in \mathscr{Q}_{u_i}(u_j)} \mathscr{F}_{\lambda}(u_i, u_j; w) \left(1 + \sum_{s\in C^{u_j}_{u_i}(w)} p(w, s)\, \mathscr{X}^{\lambda}_{u_i}(s)\right) \\ &\quad= \sum_{w\in \mathscr{Q}_{u_i}(u_j)} \mathscr{F}_{\lambda}(u_i, u_j; w) \left(\mathscr{X}^{\lambda}_{u_i}(w) - p\left(w, s^{u_j}_{u_i}(w)\right)\right. \\ &\qquad\qquad \left.\times \mathscr{X}^{\lambda}_{u_i}\left(s^{u_j}_{u_i}(w)\right)\right).\end{aligned} \tag{33}$$

□

From (3), we conclude that

$$\mathbb{E}_{\lambda}[u_i^*, u_j^*] = \max_{u_i\in V} \max_{u_j\in V_{u_i}\setminus\{u_i\}} \mathbb{E}^{\lambda}_{u_i}[u_j]. \tag{34}$$

```
Input: an undirected tree T = (V, E, p) with each edge e ∈ E having
a probability weight p(e) ∈ (0, 1);
Output: a Sum-Max 2-MRS ⟨u*, v*⟩ of T.
Step 0 i ← 1;
Step 1 Use DFS to traverse T with u_i as the origin, store the
        resultant rooted tree T_{u_i}, record f_{u_i}(u_j) and compute
        Pr(u_i, u_j) for all u_j ∈ V, and store D_{u_i};
        if i ≤ |V| then i ← i + 1; goto Step 1;
        else i ← 1; goto Step 2; endif
Step 2 for h = H_{u_i}, H_{u_i} − 1, ..., 1, 0 do
          for all nodes u_j on the h-level of T_{u_i} do
            X^λ_{u_i}(u_j) ← 1;
            if u_j ∉ D_{u_i} then
              Compute X^λ_{u_i}(u_j) by (30);
            else break; endif
          endfor
        endfor
Step 3 for h = H_{u_i}, H_{u_i} − 1, ..., 1, 0 do
          for all nodes u_j on the h-level of T_{u_i} do
            Y^λ_{u_i}(u_j) ← 0;
            if u_j ≠ u_i then
              w ← u_j; k ← h;
              while k ≥ 1 do
                w' ← w; w ← f_{u_i}(w); s^{u_j}_{u_i}(w) ← w'; k ← k − 1;
                Compute F_λ(u_i, u_j; w) by (19) or (20);
                Y^λ_{u_i}(u_j) ← Y^λ_{u_i}(u_j) + F_λ(u_i, u_j; w) ×
                    (X^λ_{u_i}(w) − p(w, s^{u_j}_{u_i}(w)) X^λ_{u_i}(s^{u_j}_{u_i}(w)));
              endwhile
            else goto Step 4; endif
          endfor
        endfor
Step 4 for j = 1, 2, ..., |V| do
          if j ≠ i then
            Compute E^λ_{u_i}[u_j] by (28);
          else break; endif
        endfor
        if i < |V| then
          i ← i + 1; goto Step 2;
        else
          Find the maximum of all the values of E^λ_{u_i}[u_j] and
          then determine u_i^* and u_j^* by (34);
        endif
```

Algorithm 1: Algorithm SUM-MAX.

We can compute $\mathbb{E}_\lambda[u_i^*, u_j^*]$ in the following way: for any $u_i \in V$, we first compute $\mathbb{E}^\lambda_{u_i}[u_j]$ for all $u_j \in V_{u_i} \backslash \{u_i\}$ using (28) and then find the maximum among $|V| - 1$ values of $\mathbb{E}^\lambda_{u_i}[u_j]$. We finally get $\mathbb{E}_\lambda[u_i^*, u_j^*]$ by determining the maximum of the above $|V|$ maximums. This is essentially the main framework of our dynamic programming algorithm called SUM-MAX, shown in Algorithm 1. The key task is to compute all the values of $\mathbb{E}^\lambda_{u_i}[u_j]$. We see from Theorem 10 that the essence of computing $\mathbb{E}^\lambda_{u_i}[u_j]$ is to compute $\mathcal{X}^\lambda_{u_i}(u_j)$ and $\mathcal{Y}^\lambda_{u_i}(u_j)$, and further from Theorem 9 that we can compute $\mathcal{X}^\lambda_{u_i}(u_j)$ by (30) and $\mathcal{Y}^\lambda_{u_i}(u_j)$ by (31). Specifically, we derive $\mathcal{X}^\lambda_{u_i}(u_j) = 1$ from $C_{u_i}(u_j) = \emptyset$ when u_j is a leaf of T_{u_i} and $\mathcal{Y}^\lambda_{u_i}(u_i) = 0$ from $\mathcal{Q}_{u_i}(u_i) = \emptyset$ when $u_j = u_i$. Therefore, for any $u_i \in V$, we can first compute all the values of $\mathcal{X}^\lambda_{u_i}(u_j)$, $u_j \in V_{u_i}$, level by level from the bottom of T_{u_i} to the top and afterward compute all the values of $\mathcal{Y}^\lambda_{u_i}(u_j)$, $u_j \in V_{u_i}$, level by level likewise. Based on (31), we can accumulate the value of $\mathcal{Y}^\lambda_{u_i}(u_j)$ from u_j to u_i generation by generation for reducing the space.

In order to facilitate algorithm SUM-MAX working level by level, we need to transform T into a rooted tree at every $u_i \in V$ beforehand. For this purpose, we devise a preprocessing procedure called PREP. The major idea of procedure PREP is described roughly as follows: we use the *depth-first* *search* (DFS) method to traverse T. DFS starts from u_i. Let $\mathcal{Q}_{u_i}(u_i) = \emptyset$ initially. When DFS reaches a new node v via the edge $\{u, v\}$, we set $\mathcal{Q}_{u_i}(v) = \mathcal{Q}_{u_i}(u) \cup \{u\}$ and compute $\Pr(u_i, v) = \Pr(u_i, u)p(u, v)$. This process is repeated until DFS ends. DFS with u_i as the origin produces a tree rooted at the origin, say, T_{u_i}. All the $|V|$ times DFSs obtain all the values of $\Pr(u, v)$, for all $u, v \in V$, $u \neq v$, which makes preparations for computing $\mathcal{F}_\lambda(u_i, u_j; w)$, $w \in \mathcal{Q}_{u_i}(u_j)$, and further $\mathcal{Y}^\lambda_{u_i}(u_j)$ for any $u_i \in V$ and $u_j \in V_{u_i}$. In addition, DFS also finds the set of all the leaves of T_{u_i}, say, D_{u_i}.

Theorem 10. *Given an undirected tree $T = (V, E, p)$, where each edge $e \in E$ has an independent working probability $0 < p(e) < 1$, algorithm* SUM-MAX *can find a Sum-Max 2-MRS of T correctly with a time complexity of $O((1/2)|V|^3)$ and a space complexity of $O(|V|^2)$.*

Proof. First, we analyze the time complexity of SUM-MAX. Step 0 takes $O(1)$ time. Step 1 runs $|V|$ times DFS in total. In every running (i.e., for every $i = 1, 2, \ldots, |V|$), Step 1 spends $O(|V|)$ time to traverse T and store T_{u_i}, $O(1)$ time to record $f_{u_i}(u_j)$, $O(1)$ time to compute $\Pr(u_i, u_j)$ for each $u_j \in V$, and at most $O(|V|)$ time to determine D_{u_i}. So, Step 1 takes $O(|V|^2)$ time in all. Next, SUM-MAX runs Step 2, Step 3, and Step 4 in order for every $i = 1, 2, \ldots, |V|$. Step 2 and Step 3 are both based on the bottom-up dynamic programming. Step 2 computes all the values of $\mathcal{X}^\lambda_{u_i}(u_j)$, $u_j \in V$, which takes $O(|V|)$ time by (11). Step 3 computes all the values of $\mathcal{Y}^\lambda_{u_i}(u_j)$, $u_j \in V$, the time complexity of which is $O(\sum_{u_j \in V} |\mathcal{Q}_{u_i}(u_j)|) \le O((1/2)|V|^2)$ by (9). Step 4 spends $O(|V|)$ time to compute all the values of $\mathbb{E}_{u_i}(u_j)$, $u_j \in V \setminus \{u_i\}$. Therefore, the time complexity of SUM-MAX is at most $O((1/2)|V|^3)$.

Next, we discuss the space complexity of SUM-MAX. Step 0 occupies $O(1)$ space. For every $i = 1, 2, \ldots, |V|$, Step 1 requires $O(|V|)$ space to store T_{u_i}, $O(|V|)$ space to store all $f_{u_i}(u_j)$, $u_j \in V$, and $\Pr(u_i, u_j)$, respectively, and at most $O(|V|)$ space to store D_{u_i}. Thus, Step 1 occupies $O(|V|^2)$ space in total. For every $i = 1, 2, \ldots, |V|$, Step 2 requires $O(|V|)$ space to store all the values of $\mathcal{X}^\lambda_{u_i}(u_j)$, $u_j \in V$; Step 3 requires $O(|V|)$ space to store $\mathcal{Y}^\lambda_{u_i}(u_j)$, $u_j \in V$, which dominates the space complexity of Step 3, and Step 4 requires $O(|V|)$ space to store $\mathbb{E}^\lambda_{u_i}(u_j)$, $u_j \in V$. Therefore, the space complexity of SUM-MAX is $O(|V|^2)$. □

4. Algorithm for Finding a Min-Max 2-MRS

Definition 11. Given $T = (V, E, p)$ and $\langle u_i, u_j \rangle$ of T, one lets $\mathbb{M}^\lambda_{u_i}[u_j]$, $\lambda = 1, 2$, denote the minimum reachability in T_{u_i} of $\langle u_i, u_j \rangle$. Also, one uses, $\mathcal{I}^\lambda_{u_i}(u_j)$ (resp., $\mathcal{J}^\lambda_{u_i}(u_j)$) to denote the minimum reachability in $V^\alpha_{u_i}(u_j)$ (resp., $V^\beta_{u_i}(u_j)$) of $\langle u_i, u_j \rangle$.

Theorem 12. *Given any $\langle u_i, u_j \rangle$ of T, if $u_j \in V_{u_i}$, then one gets*

$$\mathbb{M}^\lambda_{u_i}[u_j] = \min\left\{\mathcal{I}^\lambda_{u_i}(u_j), \mathcal{J}^\lambda_{u_i}(u_j)\right\}, \quad \lambda = 1, 2. \tag{35}$$

Proof. We first derive from the definition of $\mathbb{M}^\lambda_{u_i}[u_j]$ in Definition 11 that $\mathbb{M}_\lambda[u_i, u_j] = \mathbb{M}^\lambda_{u_i}[u_j] = \mathbb{M}^\lambda_{u_j}[u_i]$ and further from (4) that $\mathbb{M}^\lambda_{u_i}[u_j] = \min_{v \in V_{u_i}} \mathcal{F}_\lambda(u_i, u_j; v)$. Combining (6) in Lemma 2 and the definitions of $\mathcal{I}^\lambda_{u_i}(u_j)$ and $\mathcal{J}^\lambda_{u_i}(u_j)$, we conclude that

$$\begin{aligned}
&\mathbb{M}^\lambda_{u_i}[u_j] \\
&= \min\left\{\min_{v \in V^\alpha_{u_i}(u_j)} \mathcal{F}_\lambda(u_i, u_j; v), \min_{v \in V^\beta_{u_i}(u_j)} \mathcal{F}_\lambda(u_i, u_j; v)\right\} \\
&= \min\left\{\mathcal{I}^\lambda_{u_i}(u_j), \mathcal{J}^\lambda_{u_i}(u_j)\right\}.
\end{aligned} \tag{36}$$

□

Theorem 13. *Given any $\langle u_i, u_j \rangle$ of T, if $u_j \in V_{u_i}$, then one gets*

$$\mathcal{I}^\lambda_{u_i}(u_j) = \min_{s \in C_{u_i}(u_j)} p(u_j, s)\, \mathcal{I}^\lambda_{u_i}(s), \tag{37}$$

$$\mathcal{J}^\lambda_{u_i}(u_j) = \min_{w \in \mathcal{Q}_{u_i}(u_j)} \mathcal{F}_\lambda(u_i, u_j; w) \min_{s \in C^{u_j}_{u_i}(w)} p(w, s)\, \mathcal{I}^\lambda_{u_i}(s). \tag{38}$$

Proof. From the definition of $\mathcal{I}^\lambda_{u_i}(u_j)$ and (21) in Lemma 6, we get that $\mathcal{I}^\lambda_{u_i}(u_j) = \min_{v \in V^\alpha_{u_i}(u_j)} \Pr(u_j, v)$. Combining (7) in Lemmas 2 and 5, we conclude that

$$\begin{aligned}
&\mathcal{I}^\lambda_{u_i}(u_j) \\
&= \min\left\{\Pr(u_j, u_j), \min_{s \in C_{u_i}(u_j)} \min_{v \in V^\alpha_{u_i}(s)} \Pr(u_j, v)\right\} \\
&= \min\left\{1, \min_{s \in C_{u_i}(u_j)} p(u_j, s) \min_{v \in V^\alpha_{u_i}(s)} \Pr(s, v)\right\} \\
&= \min_{s \in C_{u_i}(u_j)} p(u_j, s)\, \mathcal{I}^\lambda_{u_i}(s).
\end{aligned} \tag{39}$$

We derive $\mathcal{J}^\lambda_{u_i}(u_j) = \min_{v \in V^\beta_{u_i}(u_j)} \mathcal{F}_\lambda(u_i, u_j; v)$ from the definition of $\mathcal{J}^\lambda_{u_i}(u_j)$. Combining (8) in Lemma 2 and (22) in Lemma 6, we conclude that

$$\begin{aligned}
&\mathcal{J}^\lambda_{u_i}(u_j) \\
&= \min\left\{\min_{v \in \mathcal{Q}_{u_i}(u_j)} \mathcal{F}_\lambda(u_i, u_j; v), \min_{w \in \mathcal{Q}_{u_i}(u_j)} \mathcal{F}_\lambda(u_i, u_j; w)\right. \\
&\qquad \left. \times \min_{s \in C^{u_j}_{u_i}(w)} p(w, s) \min_{v \in V^\alpha_{u_i}(s)} \Pr(s, v)\right\} \\
&= \min_{w \in \mathcal{Q}_{u_i}(u_j)} \mathcal{F}_\lambda(u_i, u_j; w) \min\left\{1, \min_{s \in C^{u_j}_{u_i}(w)} p(w, s)\, \mathcal{I}^\lambda_{u_i}(s)\right\} \\
&= \min_{w \in \mathcal{Q}_{u_i}(u_j)} \mathcal{F}_\lambda(u_i, u_j; w) \min_{s \in C^{u_j}_{u_i}(w)} p(w, s)\, \mathcal{I}^\lambda_{u_i}(s).
\end{aligned} \tag{40}$$

□

From (5), we conclude that

$$\mathbb{M}_\lambda\left[u_i^\star, u_j^\star\right] = \max_{u_i\in V}\ \max_{u_j\in V_{u_i}\setminus\{u_i\}} \mathbb{M}^\lambda_{u_i}\left[u_j\right]. \qquad (41)$$

Observe that we can compute $\mathbb{M}_\lambda[u_i^\star, u_j^\star]$ in the same way as computing $\mathbb{E}_\lambda[u_i^*, u_j^*]$. Therefore, we can devise a dynamic programming algorithm called MIN-MAX to find a min-max 2-MRS of T based on the framework of SUM-MAX. The detailed presentation of MIN-MAX is omitted here and its major procedure is described as follows. Step 0 and Step 1 of MIN-MAX are same as those of SUM-MAX, which spends $O(|V|^2)$ time and requires $O(|V|^2)$ space in total. Let

$$\mathcal{K}^{u_j,\lambda}_{u_i}(s) = \min_{s'\in C_{u_i}(u_j)\setminus\{s\}} p\left(u_j, s'\right)\mathcal{I}^\lambda_{u_i}\left(s'\right), \quad \forall s\in C_{u_i}\left(u_j\right). \qquad (42)$$

For every $i = 1,\dots,|V|$, Step 2 of MIN-MAX computes all $\mathcal{I}^\lambda_{u_i}(u_j)$, $u_j\in V$, by (37) bottom-up on T_{u_i} which takes $O(|V|)$ time and requires $O(|V|)$ space. Also, Step 2 computes all $\mathcal{K}^{u_j,\lambda}_{u_i}(s)$, $s\in C_{u_i}(u_j)$, by (42) which takes $O(\sum_{u_j\in V}|C_{u_i}(u_j)|^2)$ time and requires $O(\sum_{u_j\in V}|C_{u_i}(u_j)|)$ space. By (12), we conclude that

$$|V|^2 \geq \sum_{u_j\in V}|C_{u_i}(u_j)|^2 \geq \frac{|V|^2}{\left|V\setminus D_{u_i}\right|} \geq \frac{|V|^2}{|V\setminus D|+1}. \qquad (43)$$

Hence, the time complexity of Step 2 is $\Omega(|V|^2)$. Also, we conclude by (11) that the space complexity of Step 2 is $O(|V|) + O(\sum_{u_j\in V}|C_{u_i}(u_j)|) = O(|V|)$. By (38), we conclude that

$$\mathcal{J}^\lambda_{u_i}\left(u_j\right) = \min_{w\in\mathcal{Q}_{u_i}(u_j)} \mathcal{F}_\lambda\left(u_i, u_j; w\right)\mathcal{K}^{w,\lambda}_{u_i}\left(s^{u_j}_{u_i}(w)\right). \qquad (44)$$

For every $u_j\in V$, Step 3 of MIN-MAX computes $\mathcal{J}^\lambda_{u_i}(u_j)$ using the method of comparing generation by generation amongst $\mathcal{Q}_{u_i}(u_j)$. In every comparison, Step 3 first computes $\mathcal{F}_\lambda(u_i, u_j; w)$ and then $\mathcal{J}^\lambda_{u_i}(u_j)$ by (44). So, Step 3 spends $O(|\mathcal{Q}_{u_i}(u_j)|)$ time and requires $O(1)$ space for every $u_j\in V$. Hence, Step 3 spends $O(\sum_{u_j\in V_{u_i}}|\mathcal{Q}_{u_i}(u_j)|) \leq O((1/2)|V|^2)$ time by Lemma 3 and requires $O(|V|)$ space. Step 4 of MIN-MAX is same as that of SUM-MAX. Therefore, we obtain Theorem 14.

Theorem 14. *Given an undirected tree $T = (V, E, p)$, where each edge $e\in E$ has an independent working probability $0 < p(e) < 1$, algorithm* MIN-MAX *can find a Min-Max 2-MRS of T correctly with a time complexity of $\Omega(|V|^3)$ and a space complexity of $O(|V|^2)$.*

5. Numerical Results

In this section, we give an example tree with 35 nodes shown in Figure 2 for illustrating algorithms SUM-MAX and MIN-MAX. The decimal associated with every edge of the tree represents its operational probability. All the nodes are labeled by numbers $00, 01, 02, \dots, 34$ in order. For ease of view and comparison, the data output by algorithms are corrected to four decimal places and listed in Table 1. We first introduce the notations shown in the first line of Table 1. Let

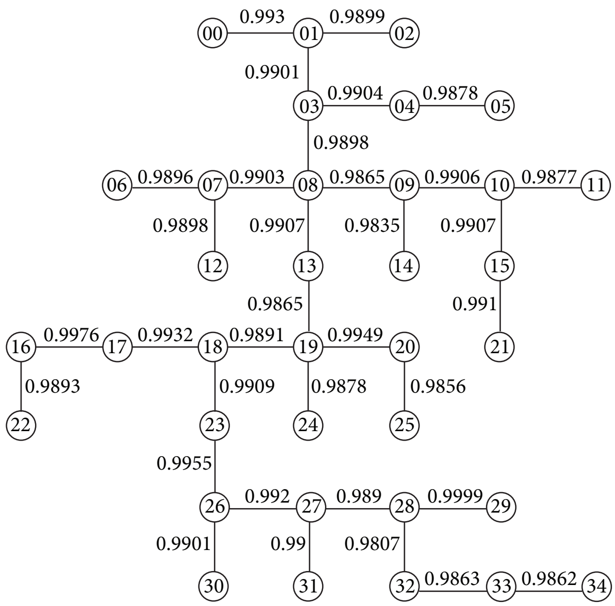


Figure 2: An example tree with a probability weight on every edge.

$$\mathbb{E}^\lambda_{u_i}\left[u_j^{\Delta_\lambda}\right] = \max_{u_j\in V_{u_i}\setminus\{u_i\}} \mathbb{E}^\lambda_{u_i}\left[u_j\right], \quad \lambda = 1, 2,\ \forall u_i\in V. \qquad (45)$$

And then by (34)

$$\mathbb{E}_\lambda\left[u_i^*, u_j^*\right] = \max_{u_i\in V}\mathbb{E}^\lambda_{u_i}\left[u_j^{\Delta_\lambda}\right]. \qquad (46)$$

Similarly, let

$$\mathbb{M}^\lambda_{u_i}\left[u_j^{\nabla_\lambda}\right] = \max_{u_j\in V_{u_i}\setminus\{u_i\}} \mathbb{M}^\lambda_{u_i}\left[u_j\right], \quad \lambda = 1, 2,\ \forall u_i\in V. \qquad (47)$$

And then by (41)

$$\mathbb{M}_\lambda\left[u_i^\star, u_j^\star\right] = \max_{u_i\in V}\mathbb{M}^\lambda_{u_i}\left[u_j^{\nabla_\lambda}\right]. \qquad (48)$$

From Table 1, it is easy to see that the maximum in the third column is $\mathbb{E}^1_{08}[26] = \mathbb{E}^1_{26}[08] = 34.1822$ and thus $\langle 8, 26\rangle$ is the unique Sum-Max 2-MRS of the tree under the superior probability. The maximum in the fifth column is $\mathbb{E}^2_{10}[33] = \mathbb{E}^2_{33}[10] = 34.5723$ and thus $\langle 10, 33\rangle$ is the unique Sum-Max 2-MRS of the tree under the united probability. Likewise, we can see easily that the maximum in the seventh column is

$$\mathbb{M}^1_k[13] = 0.9505, \qquad k = 28, 29, 32, \qquad (49)$$

and thus there are three pairs of Min-Max 2-MRS of the tree under the superior probability, that is,

$$\langle 13, 28\rangle, \langle 13, 29\rangle, \langle 13, 32\rangle. \qquad (50)$$

TABLE 1: All the major data produced by SUM-MAX and MIN-MAX.

u_i	$u_j^{\Delta_1}$	$\mathbb{E}^1_{u_i}[u_j^{\Delta_1}]$	$u_j^{\Delta_2}$	$\mathbb{E}^2_{u_i}[u_j^{\Delta_2}]$	$u_j^{\nabla_1}$	$\mathbb{M}^1_{u_i}[u_j^{\nabla_1}]$	$u_j^{\nabla_2}$	$\mathbb{M}^2_{u_i}[u_j^{\nabla_2}]$
00	23	33.9225	33	34.5400	26	0.9337	32	0.9573
01	23	34.0221	33	34.5495	27	0.9402	32	0.9578
02	23	33.8806	33	34.5358	23	0.9307	32	0.9571
03	23	34.1265	33	34.5441	28	0.9486	32	0.9586
04	23	34.0053	33	34.5387	27	0.9405	32	0.9579
05	23	33.8363	33	34.5223	23	0.9290	32	0.9569
06	23	33.9210	33	34.4942	27	0.9402	32	0.9578
07	23	34.0722	33	34.5079	28	0.9486	32	0.9587
08	26	34.1822	34	34.5031	28	0.9486	32	0.9594
09	23	34.1098	33	34.5543	28	0.9454	32	0.9673
10	23	34.0282	33	34.5723	27	0.9434	32	0.9665
11	19	33.8772	33	34.5555	26	0.9346	32	0.9656
12	23	33.9239	33	34.4945	27	0.9404	32	0.9579
13	26	34.0825	34	34.3685	28	0.9505	28	0.9505
14	23	33.8749	33	34.5319	27	0.9395	32	0.9660
15	23	33.9196	33	34.5699	27	0.9375	32	0.9658
16	08	34.0480	21	34.3445	18	0.9232	23	0.9274
17	08	34.0804	21	34.3436	18	0.9232	23	0.9274
18	08	34.1591	21	34.3338	23	0.9275	23	0.9275
19	08	34.0540	21	34.1940	27	0.9377	27	0.9377
20	08	33.9853	21	34.1965	26	0.9329	27	0.9375
21	19	33.8721	33	34.5577	23	0.9290	32	0.9651
22	08	33.9138	15	34.3392	18	0.9232	23	0.9273
23	08	34.1759	21	34.4128	00	0.9317	00	0.9317
24	08	33.8656	21	34.1872	23	0.9262	27	0.9373
25	08	33.7640	15	34.1887	18	0.9232	27	0.9371
26	08	34.1822	21	34.4471	01	0.9359	00	0.9359
27	08	34.1818	15	34.4928	03	0.9434	00	0.9434
28	08	34.1386	15	34.5332	13	0.9505	00	0.9539
29	08	34.1374	15	34.5331	13	0.9505	08	0.9539
30	08	34.0665	15	34.4416	00	0.9266	19	0.9357
31	08	34.0649	15	34.4872	01	0.9340	13	0.9430
32	08	34.0582	15	34.5631	13	0.9505	09	0.9673
33	08	34.0013	10	34.5723	13	0.9471	09	0.9671
34	08	33.9397	10	34.5677	13	0.9453	09	0.9669

The maximum in the ninth column is $\mathbb{M}^2_{09}[32] = \mathbb{M}^2_{32}[09] = 0.9673$ and thus $\langle 9, 32\rangle$ is the unique Min-Max 2-MRS of the tree under the united probability.

6. Discussions and Future Works

This paper suggested the models of superior probability and united probability of node pair and studied two kinds of 2-MRS problem (i.e., Sum-Max 2-MRS and Min-Max 2-MRS) in a tree with each edge having an independent working probability and all the nodes being immune to failures. The paper presents $O((1/2)|V|^3)$-time and $O(|V|^2)$-space algorithm for finding a Sum-Max 2-MRS of the tree and $\Omega(|V|^3)$-time and $O(|V|^2)$-space algorithm for finding a Min-Max 2-MRS. It is also interesting to study the 2-MRS problem in a series-parallel graph; see [7]. Two servers involved in the paper work synchronously. In a number of practical scenarios, however, one of two servers works and the other gets ready. In the case, we can first find the two most reachable nodes using the algorithms in [10, 20] and then placing two servers optimally by placing the working server at the most reachable node and the backup one at the second most reachable node.

When we are given a large-scale graph, we need to place more than two servers to supply synchronous service for the whole network. It is of interest to study the *k-MRS problem*

Table 2: Notations table.

Notation	Explanation
$\mathscr{G} = (\mathscr{V}, \mathscr{E}, p)$	An undirected connected graph
$T = (V, E, p)$	An undirected tree graph
$T_u = (V_u, E_u, p)$	A rooted version of T with u as the root
$\{u, v\}$	An edge of graph
$\langle u, v\rangle$	A node pair
$\pi(u, v)$	A simple path connecting nodes u and v (also, the event that $\pi(u, v)$ works correctly)
$\Pr(\pi(u, v))$	The probability of $\pi(u, v)$ working correctly
$\Pr(u, v)$	The probability of u reaching v
$\mathscr{V}(\pi(u, v))$	The set of nodes on $\pi(u, v)$
$\mathscr{E}(\pi(u, v))$	The set of edges on $\pi(u, v)$
$C_u(v)$	The set of the children of v in T_u
$T_u(v)$	The subtree of T_u rooted at v
$V_u^{\alpha}(v)$	The set of nodes in $T_u(v)$
$V_u^{\beta}(v)$	The set of nodes outside $T_u(v)$
$f_u(v)$	The parent of v in T_u
$\mathscr{Q}_u(v)$	The set of ancestors of v in T_u
$s_u^{v}(w)$	The child of w on $\pi(u, v)$ in T_u
$C_u^{v}(w)$	The set of children of w in T_u other than $s_u^{v}(w)$
H_u	The number of the most ancestors of node in T_u
h	The current level of T_u ($h = 1, 2, \ldots, H_u + 1$)
$V_u(h)$	The set of nodes on the h-level of T_u
D	The set of leaves of T
D_u	The set of leaves of T_u
$A \oplus B$	The union of two disjoint sets A and B
$\lambda = 1$	The superior probability
$\lambda = 2$	The united probability
$\mathscr{F}_1(u_i, u_j; v)$	The superior probability of $\langle u_i, u_j\rangle$ to v
$\mathscr{F}_2(u_i, u_j; v)$	The united probability of $\langle u_i, u_j\rangle$ to v
$\mathbb{E}_{\lambda}[u_i, u_j]$	The sum reachability of $\langle u_i, u_j\rangle$
$\mathbb{E}_{u_i}^{\lambda}[u_j]$	The sum reachability in T_{u_i} of $\langle u_i, u_j\rangle$
$\mathscr{X}_{u_i}^{\lambda}(u_j)$	The sum reachability in $V_{u_i}^{\alpha}(u_j)$ of $\langle u_i, u_j\rangle$
$\mathscr{Y}_{u_i}^{\lambda}(u_j)$	The sum reachability in $V_{u_i}^{\beta}(u_j)$ of $\langle u_i, u_j\rangle$
$\mathbb{M}_{\lambda}[u_i, u_j]$	The minimum reachability of $\langle u_i, u_j\rangle$
$\mathbb{M}_{u_i}^{\lambda}[u_j]$	The minimum reachability in T_{u_i} of $\langle u_i, u_j\rangle$
$\mathscr{I}_{u_i}^{\lambda}(u_j)$	The minimum reachability in $V_{u_i}^{\alpha}(u_j)$ of $\langle u_i, u_j\rangle$
$\mathscr{J}_{u_i}^{\lambda}(u_j)$	The minimum reachability in $V_{u_i}^{\beta}(u_j)$ of $\langle u_i, u_j\rangle$

with $k \geq 3$. It seems that our method proposed in the paper cannot be directly generalized to the k-MRS problem. Thus new ideas are required.

References

[1] J. A. Bondy and U. S. R. Murty, *Graph Theory with Applications*, Macmillan, London, UK, 1976.

[2] D. B. West, *Introduction to Graph Theory*, Prentice Hall, Upper Saddle River, NJ, USA, 1996.

[3] C. J. Colbourn, *The Combinatorics of Network Reliability*, International Series of Monographs on Computer Science, Oxford University Press, New York, NY, USA, 1987.

[4] D. R. Shier, *Network Reliability and Algebraic Structures*, Oxford Science Publications, Oxford University Press, New York, NY, USA, 1991.

[5] M. O. Ball and F. L. Lin, "Reliability model applied to emergency service vehicle location," *Operations Research*, vol. 41, no. 1, pp. 18–36, 1993.

[6] M. O. Ball, J. S. Provan, and D. R. Shier, "Reliability covering problems," *Networks*, vol. 21, no. 3, pp. 345–357, 1991.

[7] C. J. Colbourn and G. Xue, "A linear time algorithm for computing the most reliable source on a series-parallel graph with unreliable edges," *Theoretical Computer Science*, vol. 209, no. 1-2, pp. 331–345, 1998.

[8] W. Ding, "Computing the most reliable source on stochastic ring networks," in *Proceedings of the WRI World Congress on Software Engineering (WCSE '09)*, vol. 1, pp. 345–347, Xiamen, China, May 2009.

[9] W. Ding, "Extended most reliable source on an unreliable general network," in *Proceedings of the International Conference on Internet Computing and Information Services (ICICIS '11)*, pp. 529–533, Hong Kong, China, September 2011.

[10] W. Ding and G. Xue, "A linear time algorithm for computing a most reliable source on a tree network with faulty nodes," *Theoretical Computer Science*, vol. 412, no. 3, pp. 225–232, 2011.

[11] W. Ding and G. Xue, "A divide-and-conquer algorithm for finding a most reliable source on a ring-embedded tree network with unreliable edges," *Discrete Mathematics, Algorithms and Applications*, vol. 3, no. 4, pp. 503–516, 2011.

[12] W. Ding and G. Xue, "A fast parallel algorithm for finding a most reliable source on a general ring-tree graph with unreliable edges," in *Combinatorial Optimization and Applications*, vol. 6831 of *Lecture Notes in Computer Science*, pp. 98–112, Springer, Heidelberg, Germany, 2011.

[13] H. A. Eiselt, M. Gendreau, and G. Laporte, "Location of facilities on a network subject to a single-edge failure," *Networks*, vol. 22, no. 3, pp. 231–246, 1992.

[14] M. E. Helander and E. Melachrinoudis, "Facility location and reliable route planning in hazardous material transportation," *Transportation Science*, vol. 31, no. 3, pp. 216–226, 1997.

[15] E. Melachrinoudis and M. E. Helander, "A single facility location problem on a tree with unreliable edges," *Networks*, vol. 27, no. 3, pp. 219–237, 1996.

[16] P. B. Mirchandani and A. R. Odoni, "Locations of medians on stochastic networks," *Transportation Science*, vol. 13, no. 2, pp. 85–97, 1979.

[17] L. D. Nel and C. J. Colbourn, *Locating a Broadcast Facility in an Unreliable Network*, vol. 28, INFOR, 1990.

[18] J. Santivanez and E. Melachrinoudis, "Location of a reliable center on a tree network," *Operational Research*, vol. 7, no. 3, pp. 419–445, 2007.

[19] J. Santiváñez, E. Melachrinoudis, and M. E. Helander, "Network location of a reliable center using the most reliable route policy," *Computers and Operations Research*, vol. 36, no. 5, pp. 1437–1460, 2009.

[20] G. Xue, "Linear time algorithms for computing the most reliable source on an unreliable tree network," *Networks*, vol. 30, no. 1, pp. 37–45, 1997.

[21] Y. Xiao, K. Thulasiraman, X. Fang, D. Yang, and G. Xue, "Computing a most probable delay constrained path: NP-hardness

and approximation schemes," *IEEE Transactions on Computers*, vol. 61, no. 5, pp. 738–744, 2012.

[22] D. Yang, X. Fang, and G. Xue, "ESPN: efficient server placement in probabilistic networks with budget constraint," in *Proceedings of the IEEE INFOCOM 2011*, pp. 1269–1277, Shanghai, China, April 2011.

[23] F. J. Berchmans, W. K. Hon, A. C. Y. Huang, C. S. Liu, E. Lo, and D. K. Y. Yau, "Optimizing link assignment to enhance service in probabilistic network," in *Proceedings of the 7th Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks (SECON '10)*, Boston, Mass, USA, June 2010.

[24] S. Ji, R. Beyah, and Z. Cai, "Snapshot and continuous data collection in probabilistic wireless sensor networks," *IEEE Transactions on Mobile Computing*, 2013.

[25] S. Ji, J. He, Y. Pan, and Y. Li, "Continuous data aggregation and capacity in probabilistic wireless sensor netwroks," *Journal of Parallel and Distributed Computing*, vol. 73, no. 6, pp. 729–745, 2013.