

On the PROBABILISTIC MIN SPANNING TREE Problem

Nicolas Boria · Cécile Murat · Vangelis Paschos

Received: 23 May 2011 / Accepted: 11 October 2011
© Springer Science+Business Media B.V. 2011

Abstract We study a probabilistic optimization model for MIN SPANNING TREE, where any vertex v_i of the input-graph $G(V, E)$ has some presence probability p_i in the final instance $G' \subset G$ that will effectively be optimized. Suppose that when this “real” instance G' becomes known, a spanning tree T , called *anticipatory* or *a priori* spanning tree, has already been computed in G and one can run a quick algorithm (quicker than one that recomputes from scratch), called *modification strategy*, that modifies the anticipatory tree T in order to fit G' . The goal is to compute an anticipatory spanning tree of G such that, its modification for any $G' \subseteq G$ is optimal for G' . This is what we call PROBABILISTIC MIN SPANNING TREE problem. In this paper we study complexity and approximation of PROBABILISTIC MIN SPANNING TREE in complete graphs under two distinct modification strategies leading to different complexity results for the problem. For the first of the strategies developed, we also study two natural subproblems of PROBABILISTIC MIN SPANNING TREE, namely, the PROBABILISTIC METRIC MIN SPANNING TREE and the PROBABILISTIC MIN SPANNING TREE 1,2 that deal with metric complete graphs and complete graphs with edge-weights either 1, or 2, respectively.

Keywords Complexity · Approximation · Probabilistic optimization · Spanning tree

Research supported by the French Agency for Research under the DEFIS program TODO, ANR-09-EMER-010.

N. Boria · C. Murat (✉) · V. Paschos
LAMSADE, CNRS and Université Paris-Dauphine, Paris cedex 16, France
e-mail: murat@lamsade.dauphine.fr

N. Boria
e-mail: boria@lamsade.dauphine.fr

V. Paschos
e-mail: paschos@lamsade.dauphine.fr

V. Paschos
Institut Universitaire de France, Paris, France

1 Introduction

The basic problematic of probabilistic combinatorial optimization (in graphs) is the following. We are given a graph $G(V, E)$ on which we have to solve some optimization problem Π . But, for some reasons depending on the reality modeled by G , Π is only going to be solved for some subgraph G' of G (determined by the vertices that will finally be present) rather than for the whole of G . The measure of how likely it is that a vertex $v_i \in V$ will belong to G' (i.e., will be present for the final optimization) is expressed by a probability p_i associated with v_i . How we can proceed in order to solve Π under this kind of uncertainty?

A first very natural idea that comes to mind is that one waits until G' is specified (i.e., it is present and ready for optimization) and, at this time, one solves Π in G' . This is what is called *reoptimization*. But what if there remains very little time for such a computation? In this case, another way to proceed is the following. One solves Π in the whole of G in order to get a feasible solution (denoted by S), called a *a priori* or *anticipatory solution*, which will serve her/him as a kind of benchmark for the solution on the effectively present subgraph G' . One has also to be provided with an algorithm that modifies S in order to fit G' . This algorithm is called *modification strategy* (let us denote it by \mathbb{M}) and produces from S a solution S' that is feasible for Π in G' . The objective now becomes to compute an anticipatory solution that, when modified by \mathbb{M} , remains “good” for any subgraph of G (if this subgraph is the one where Π will be finally solved). This amounts to computing a solution that optimizes a kind of expectation of the value of the modification of S over all the possible subgraphs of G , i.e., the sum of the products of the probability that G' is the finally present graph multiplied by the value of S' . Obviously, the presence-probability of G' is the probability that all of its vertices are present and the other vertices outside G' are absent. This expectation, depending on both the instance of the deterministic problem Π , the vertex-probabilities, and the modification strategy adopted, is called the *functional* and is expressed by:

$$E(G, S, \mathbb{M}) = \sum_{V' \subseteq V} \Pr[V'] m(G', S', \mathbb{M}) \quad (1)$$

where $\Pr[V'] = \prod_{v_i \in V'} p_i \prod_{v_i \in V \setminus V'} (1 - p_i)$ is the distribution describing probability of occurrence of a specific subset $V' \subseteq V$, i.e., of the graph $G[V']$ and $m(G', S', \mathbb{M})$ is the value of the solution S' produced by application of \mathbb{M} on the anticipatory solution S .

Seen in this way, the probabilistic version **PROBABILISTIC** Π of a (deterministic) combinatorial optimization problem Π becomes another equally deterministic problem Π' , the solutions of which have the same feasibility constraints as those of Π but with a different objective function where vertex-probabilities intervene. In this sense, probabilistic combinatorial optimization is very close to what in the last couple of years has been called “one stage optimisation under independent decision models”, an area very popular in the stochastic optimization community. It is hopefully clear from Eq. 1 and from the whole discussion above, that $m(G', S', \mathbb{M})$ strongly depends on the modification strategy \mathbb{M} used to adapt the anticipatory solution S to the present graph G' . So, both $E(G, S, \mathbb{M})$ and the a priori solution S^* optimizing it, also strongly depend on \mathbb{M} . In other words, as we will see later, for a fixed instance I of a deterministic problem Π , *two distinct modification strategies induce two distinct probabilistic problems* having I and Π as common deterministic supports. These two

distinct problems may have very different functionals and, consequently, different optimal anticipatory solutions that induce very different complexities for computing them.

What are the main mathematical problems dealing with probabilistic consideration of a problem Π in the sense discussed above? One can identify at least five interesting mathematical and computational problems dealing with probabilistic combinatorial optimization:

1. Write the functional down in an analytical closed form;
2. If such an expression of the functional is possible, prove that its value is polynomially computable (this amounts to proving that the decision version of the modified problem Π' belongs to **NP**);
3. Determine the complexity of the computation of the optimal *a priori* solution, i.e., of the solution optimizing the functional (in other words, determine the computational complexity of Π');
4. If Π' is **NP**-hard, study polynomial approximation issues;
5. Always, under the hypothesis of the **NP**-hardness of Π' , determine its complexity in the special cases where Π is polynomial, and in the case of **NP**-hardness, study approximation issues.

Let us note that, although curious, point 2 in the above list is neither trivial nor senseless. Simply consider that the summation for the functional includes, in a graph of order n , 2^n terms (one for each subgraph of G). So, polynomiality of the computation of the functional is, in general, not immediate.

Several optimization frameworks have been introduced by the operations research community for handling data uncertainty, the most well developed being *Stochastic programming* (see [7, 13, 29] for basics) and *Robust discrete optimization* (see, for example, [21]). The framework of *Probabilistic Combinatorial Optimization* where our work is located was introduced by [2, 16]. In [1–5, 16–19], restricted versions of routing and network-design probabilistic minimization problems (in complete graphs) have been studied under the robustness model dealt here (called *a priori optimization*). In [6], the analysis of the probabilistic minimum travelling salesman problem, originally performed in [2, 16], has been revisited and refined. Several other combinatorial problems have been handled in the probabilistic combinatorial optimization framework, like minimum coloring [8, 26], maximum independent set and minimum vertex cover [24, 25], longest path [23], Steiner tree problems [27, 28]. Note also that probabilistic minimum spanning tree has also studied by [4] but under a very different probabilistic model.

2 Preliminaries

We apply in this paper the probabilistic combinatorial optimization setting just described in the minimum spanning tree problem. Given an edge-weighted graph $G(V, E)$, with positive edge weights $d : E \rightarrow \mathbb{Q}^+$, the minimum spanning tree problem (MIN SPANNING TREE) consists of determining a minimum total edge-weight tree spanning V . MIN SPANNING TREE is a celebrated problem, very frequently modeling several kinds of networks in transports, communications, energy, logistics, etc.

One of the first algorithms for MIN SPANNING TREE is the seminal Kruskal's Algorithm [22] which consists of sorting edges in non-decreasing order with respect to their weights, and of inserting them in the current solution provided that they do not create cycles with those already inserted. The complexity of this algorithm is $O(m \log n)$ where $m = |E|$ and $n = |V|$. Several improvements have been performed upon this result, for instance, the $O(m \log \log n)$ result given independently by [31] and [11]. More recently, [10] proposes an algorithm for MIN SPANNING TREE (the best known until now, to our knowledge) with running time $O(m\alpha(m, n))$, where α is the functional inverse of Ackerman's function defined by [30]. We also quote an implementation of Prim's algorithm (this is another famous MIN SPANNING TREE-algorithm) using Fibonacci heaps that works in $O(\max\{m, n \log n\})$ [12].

MIN SPANNING TREE has been actively studied under several optimization models like on-line computation, dynamic optimization, etc. Its study always motivates numerous researchers in theoretical computer science and in operational research.

The formal definition of PROBABILISTIC MIN SPANNING TREE under modification strategy M is as follows. Consider a complete weighted graph $G(V, E)$ on n vertices, with edge weights given by a function $d: E \rightarrow \mathbb{Q}^+$. Set $V = \{v_1, v_2, \dots, v_n\}$. Each vertex $v_i \in V$, is associated with a presence probability $p_i \in \mathbb{Q}^+$ measuring, as already mentioned, how likely is that v_i will be present in the instance where PROBABILISTIC MIN SPANNING TREE will really be solved. We assume that subgraph $G'(V', E')$ of G materializes as the outcome of n independent Bernoulli trials, one per vertex $v_i \in V$: $v_i \in V'$ with probability p_i . Then, $E' = \{(u, v) \in E : u \in V' \text{ and } v \in V'\}$. The assumption that the input-graph is complete is made in order to ensure connectivity of the tree T' for any subgraph $G' \subseteq G$; in any case if the real-world problem at hand implies non-complete graphs, one can complete them by "appropriately heavy" edges.

Consider an application of PROBABILISTIC MIN SPANNING TREE coming from message broadcasting in a computer network [9]. We use there a tree spanning the vertices of this network. Assuming that the weight of an edge of the network represents the transmission-cost on the link represented by this edge, we look for a minimum-cost tree spanning all the vertices of the underlying network. In distributed computation, the structure of a tree allows broadcasting from the root to the leaves, and vice-versa, as well as resolution of several problems dealing with synchronization, or mutual exclusion [9]. Assume that each vertex v_i of the network can fail or be unavailable with some probability p_i , except some vertex, denoted by v_1 , that represents the central server that "never" fails (or very rarely). In such case, we can consider that the presence probability of v_1 is 1 and of any other vertex v_i is $1 - p_i$. Given some modification strategy, motivated by the particular application to be handled, the objective is to determine some spanning tree with minimum expectation. Other applications dealing with PROBABILISTIC MIN SPANNING TREE can be found in [2].

In what follows, we design two modification strategies for PROBABILISTIC MIN SPANNING TREE and study the two distinct probabilistic optimization problems derived. Our assumption that the input-graph of PROBABILISTIC MIN SPANNING TREE is complete is made mainly in order that the modification strategies associated to these models, called CLOSEST_ANCESTOR and ROOT, respectively, compute feasible solutions. Let us note that the PROBABILISTIC MIN SPANNING TREE model proposed in [4], constructs feasible solutions sometimes using vertices non-present in V' when the vertices of V' don't allow achievement of such solutions (i.e., when the obtained arborescence

cannot be connected in $G[V']$). In Section 4, we handle **PROBABILISTIC MIN SPANNING TREE** under strategy **CLOSEST_ANCESTOR**, and derive an analytic expression of the functional of the solutions T^* . We next show that, under this strategy the problem of the *a priori optimization*, i.e., the problem of determining an anticipatory solution minimizing the functional, is **NP-hard** in general complete graphs (Section 4.1). Subsequently, we study complexity of the **PROBABILISTIC MIN SPANNING TREE** problem when dealing with particular cases of anticipatory solutions (Section 4.2). We next derive polynomial-time approximation results for metric graphs (Section 4.3.1) and for graphs where edge-weights are either 1 or 2 (Section 4.3.2). Next, in Section 5, we handle **PROBABILISTIC MIN SPANNING TREE** under another modification strategy, called **ROOT**, and show that, under this strategy, **PROBABILISTIC MIN SPANNING TREE** is polynomial.

Let us note that a complementary framework to the one of the *a priori optimization*, is the *reoptimization* consisting of solving ex nihilo and optimally the portion of the instance presented for optimization. Reoptimization is introduced in [16]. Let $\text{opt}(G')$ refer to the weight of the optimum spanning tree on G' for every subgraph $G'(V', E')$ of G . The expected minimum weight over the distribution of subgraphs of G , i.e., the functional of reoptimization is:

$$E^*(G) = \sum_{V' \subseteq V} \Pr[V'] \text{opt}(G') \quad (2)$$

Obviously, denoting by T_M^* the optimal anticipatory solution of **PROBABILISTIC MIN SPANNING TREE**, under any modification strategy M , it holds that:

$$E^*(G) \leq E(G, T_M^*, M) \quad (3)$$

In Section 3, we perform further study of reoptimization and exhibit some relations between it and (deterministic) **MIN SPANNING TREE**.

For simplicity, in Sections 4 and 5 we simplify notations and omit strategies' names from the expressions for the functionals. However, we underline once more the fact that the probabilistic versions of **PROBABILISTIC MIN SPANNING TREE** studied in these sections are indeed two distinct optimization problems.

3 Reoptimization and **MIN SPANNING TREE**

We study in this section relations between the value of the expectation $E^*(G)$ in the reoptimization setting and the optimal value of the (deterministic) **MIN SPANNING TREE**. Let us note that the technique used for achieving the result of the following Proposition 1 is the same as for the analogous result in [4], up to the fact that we set $p_1 = 1$.

Proposition 1 *Consider a complete edge-weighted graph G defined on a set V of n vertices $V = \{v_1, \dots, v_n\}$ associated with a system of vertex probabilities $p_1 = 1$, $p_i = p$, $i = 2, \dots, n$. Denote by $\text{opt}(G)$ the value of an optimal solution T^* for (deterministic) **MIN SPANNING TREE**. Then, $E^*(G) \geq p \text{opt}(G)$.*

Proof Since vertex v_1 (assumed to be the root of every tree solution of MIN SPANNING TREE in every subgraph of G) is always present, setting $G' = G[V']$, Eq. 2 can be written as:

$$E^*(G) = \sum_{k=2}^n p^{k-1} (1-p)^{n-k} \times \sum_{\substack{V' \subseteq V \\ |V'|=k}} \text{opt}(G') \quad (4)$$

Set $D_k = \sum_{V' \subseteq V, |V'|=k} \text{opt}(G')$. Then, Eq. 4 becomes:

$$E^*(G) = \sum_{k=2}^n p^{k-1} (1-p)^{n-k} D_k \quad (5)$$

We first show that:

$$D_k \geq \binom{n-2}{k-2} \text{opt}(G) \quad (6)$$

Denote by $E[T^*]$ the edge-set of the tree T^* . Then, for every $i \neq 1$, set $V_i = V \setminus \{v_i\}$ and denote by $f(i)$, the father of i in T^* . Setting $G_i = G[V_i]$, we have for every $(v_i, v_{f(i)}) \in E[T^*]$, $i \neq 1$:

$$\text{opt}(G_i) + d_{i, f(i)} \geq \text{opt}(G) \quad (7)$$

since a minimum spanning tree of G_i plus some edge connecting v_i to it is a tree of G . Summing Eq. 7 over $i = 2, \dots, n$, we get:

$$\sum_{i=2}^n \text{opt}(G_i) + \sum_{i=2}^n d_{i, f(i)} \geq (n-1) \text{opt}(G) \quad (8)$$

Moreover, obviously, it holds that

$$\sum_{i=2}^n d_{i, f(i)} = \text{opt}(G) \quad (9)$$

From Eqs. 8 and 9 the following is deduced:

$$D_{n-1} = \sum_{i=2}^n \text{opt}(G_i) \geq (n-2) \text{opt}(G) \quad (10)$$

Let us now consider the $t = C_{n-1}^{k-1}$ subsets of V of size k (recall that v_1 is omnipresent), A_1, A_2, \dots, A_t . For any i , set $A_{i,j} = A_i \setminus \{j\}$. With a reasoning completely similar to the above, setting $G_{i,j} = G[A_{i,j}]$, we can derive:

$$\sum_{j \in A_i} \text{opt}(G_{i,j}) \geq (k-2) \text{opt}(G_i) \quad (11)$$

$$\sum_{i=1}^t \sum_{j \in A_i} \text{opt}(G_{i,j}) \geq (k-2) \sum_{i=1}^t \text{opt}(G_i) = (k-2) D_k \quad (12)$$

where Eq. 12 is obtained summing Eq. 11 for every i . Since in the lefthand side term each of the C_{n-1}^{k-2} subsets of V (of cardinality $k-1$) is taken into account $n-k+1$ times, this term can be written:

$$\sum_{i=1}^t \sum_{j \in A_i} \text{opt}(G_{i,j}) = (n-k+1)D_{k-1} \quad (13)$$

Indeed, w.l.o.g. let us assume that $A_1 = \{v_1, v_2, \dots, v_{k-1}, v_k\}$, $A_2 = \{v_1, v_2, \dots, v_{k-1}, v_{k+1}\}$, $A_3 = \{v_1, v_2, \dots, v_{k-1}, v_{k+2}\}$, ..., $A_{n-k+1} = \{v_1, v_2, \dots, v_{k-1}, v_n\}$. Let us consider set $A_{1,k}$ obtained from A_1 by removing vertex v_k :

$$\underbrace{\overbrace{v_1, v_2, \dots, v_{k-1}}^{A_{1,k}}, v_k, \dots, v_n}_{A_1}$$

We build in the same way pairs $(A_2, A_{2,k+1})$, $(A_3, A_{3,k+2})$, ..., $(A_{n-k+1}, A_{n-k+1,n})$:

$$\begin{aligned} & \underbrace{\overbrace{v_1, v_2, \dots, v_{k-1}, v_{k+1}}^{A_{2,k+1}}, v_k, v_{k+2}, \dots, v_n}_{A_2} \\ & \vdots \\ & \underbrace{\overbrace{v_1, v_2, \dots, v_{k-1}, v_n}^{A_{n-k+1,n}}, v_k, \dots, v_{n-1}}_{A_{n-k+1}} \end{aligned}$$

We can easily see that $A_{1,k} = A_{2,k+1} = A_{3,k+2} = \dots = A_{n-k+1,n}$; hence, any of the C_{n-1}^{k-2} distinct subsets of V of cardinality $k-1$ can be obtained using $n-k+1$ different ways, that proves Eq. 13.

Combining Eqs. 12 and 13 we get:

$$D_{k-1} \geq \frac{k-2}{n-k+1} D_k \quad (14)$$

By successively applying Eq. 14 and using Eq. 10 we derive:

$$\begin{aligned} D_k & \geq \frac{k-1}{n-k} D_{k+1} \geq \frac{(k-1)k}{(n-k)(n-(k+1))} D_{k+2} \\ & \vdots \\ & \geq \frac{(k-1)k \dots (n-3)}{(n-k)(n-(k+1)) \dots (n-(n-2))} D_{n-1} \\ & \geq \frac{(k-1)k \dots (n-3)(n-2)}{(n-k)(n-(k+1)) \dots (n-(n-1))} \text{opt}(G) \\ & = \frac{(n-2)!}{(k-2)!(n-k)!} \text{opt}(G) = \binom{n-2}{k-2} \text{opt}(G) \end{aligned}$$

as claimed in Eq. 6. Combining it with Eq. 5, we get:

$$E^*(G) \geq \sum_{k=2}^n p^{k-1} (1-p)^{n-k} \binom{n-2}{k-2} \text{opt}(G) = p \text{opt}(G)$$

completing so the proof of the proposition. \square

Combining Eq. 3 and Proposition 1, the following holds, for any modification strategy M :

$$E(G, T_M^*, M) \geq p \text{opt}(G) \quad (15)$$

Note that equality in Eq. 15 is attained for $p = 1$.

4 PROBABILISTIC MIN SPANNING TREE under CLOSEST_ANCESTOR

Consider a tree T spanning V and number vertices in T in a left-to-right breadth-first-search (bfs) way. Consider a subgraph $G'(V', E') = G[V']$ of G induced by a set $V' \subseteq V$. The modification strategy (adjusting T to a spanning tree T' of G' and denoted by CLOSEST_ANCESTOR in what follows) that will be analyzed in this section works as follows:

- Remove the vertices of $V \setminus V'$ and the edges of E incident to these vertices; let $F(V') = \{T_1, T_2, \dots, T_k\}$ be the so-obtained forest and assume that $v_1 \in V(T_1)$ and that, for $i, j = 2, \dots, k$, $i < j$ if the index of the root of T_i is smaller than that of the root of T_j ;
- For $i = 2, \dots, k$ add as father of the root of T_i its largest-index ancestor that is still present in $V(T)$.

Note that, given two vertices v_j and v_l with $j < l$ (in the bfs numbering of T), if v_j is not an ancestor of v_l in T , then edge (v_j, v_l) will never belong to any T' modification of T for any $V' \subseteq V$. Clearly, the complexity of CLOSEST_ANCESTOR is $O(n)$. Note also that, with respect to the application sketched in Section 1, this strategy allows to maintain the same order of broadcasting to the several internal nodes of the anticipatory tree, independently on the sub-instance that will finally be realized.

Figure 1b gives an example of how strategy CLOSEST_ANCESTOR works starting from an initial tree T shown in Fig. 1a and assuming that vertices 3, 5, 6, 9, 13 and 16 are absent from V' .

Our goal in this section is to study the following problem: *find an algorithm for taking “good” a priori decisions, i.e., that determines a spanning tree T^* , that optimizes $E(G, T)$ under modification strategy CLOSEST_ANCESTOR*; this is PROBABILISTIC MIN SPANNING TREE (under strategy CLOSEST_ANCESTOR). In what follows, we show that this problem is **NP**-hard in general complete graphs with $p_1 = 1$. We then study approximation of this problem in metric graphs as well as in a particular

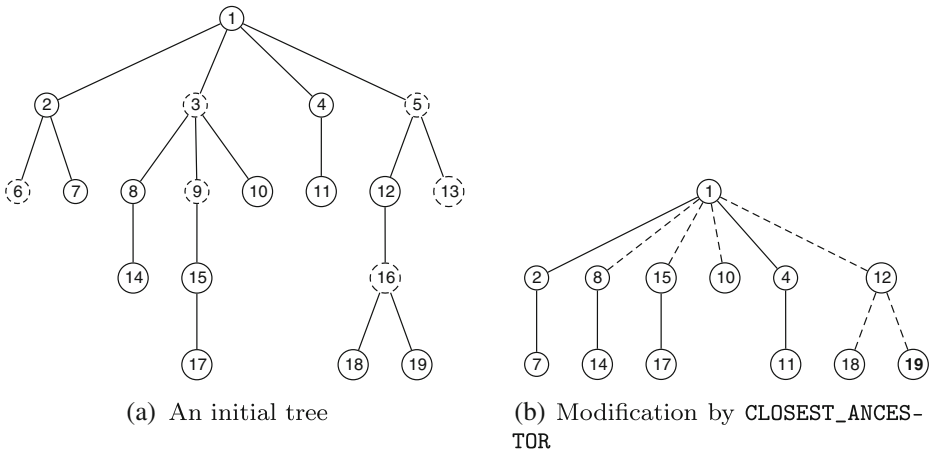


Fig. 1 An anticipatory spanning tree T and its adjustment

subclass of them where edge-weights are either 1 or 2. The approximation ratio is defined as $E(G, T)/E(G, T^*)$.

4.1 The Complexity of PROBABILISTIC MIN SPANNING TREE

The following result holds for the functional $E(G, T)$ associated with an anticipatory spanning tree T and the modification strategy CLOSEST_ANCESTOR.

Lemma 1 Consider a complete graph $G(V, E)$, provided with a vertex-probability system $(p_i)_{i=1,\dots,n}$ with $p_1 = 1$, any edge (v_i, v_j) of which has weight d_{ij} and a spanning tree T of G . Then, the expectation associated with CLOSEST_ANCESTOR can be expressed by:

$$E(G, T) = \sum_{(v_i, v_j) \in T} p_i p_j d_{ij} + \sum_{v_i \in V} \sum_{\substack{v_j \in D(v_i) \\ (v_i, v_j) \notin T}} p_i p_j \prod_{v_k \in \mu[v_i, v_j]} (1 - p_k) d_{ij} \quad (16)$$

where $D(v_i)$ is the set of successors of v_i in T and $\mu[v_i, v_j]$ is the set of vertices in the (unique) path of T from v_i to v_j not including neither of them.

Proof Following CLOSEST_ANCESTOR, if $(v_i, v_j) \in T$, then this edge will be in T' iff $v_i, v_j \in V'$; on the other hand, if $(v_i, v_j) \notin T$, then edge (v_i, v_j) will be added in T' iff $v_i, v_j \in V'$ and $v_j \in D(v_i)$ and every vertex in $\mu[v_i, v_j]$ is not in V' . From these observations we derive using Eq. 1:

$$\begin{aligned} E(G, T) &= \sum_{V' \subseteq V} \Pr[V'] m(G', T') = \sum_{V' \subseteq V} \Pr[V'] \sum_{(v_i, v_j) \in T'} d_{ij} \\ &= \sum_{V' \subseteq V} \Pr[V'] \sum_{(v_i, v_j) \in T' \cap T} d_{ij} + \sum_{V' \subseteq V} \Pr[V'] \sum_{(v_i, v_j) \in T' \cap (E \setminus T)} d_{ij} \end{aligned}$$

$$\begin{aligned}
&= \sum_{(v_i, v_j) \in T} \sum_{V' \subseteq V} \Pr[V'] \mathbf{1}_{\{(v_i, v_j) \in T'\}} d_{ij} \\
&\quad + \sum_{(v_i, v_j) \in (E \setminus T)} \sum_{V' \subseteq V} \Pr[V'] \mathbf{1}_{\{(v_i, v_j) \in T'\}} d_{ij} \\
&= \sum_{(v_i, v_j) \in T} p_i p_j d_{ij} + \sum_{v_i \in V} \sum_{\substack{v_j \in D(v_i) \\ (v_i, v_j) \notin T}} p_i p_j \prod_{v_k \in \mu[v_i, v_j]} (1 - p_k) d_{ij}
\end{aligned}$$

as claimed. \square

Clearly, Eq. 16 can be computed in polynomial time. So, the following corollary holds.

Corollary 1 PROBABILISTIC MIN SPANNING TREE \in *NPO*.

As already mentioned, PROBABILISTIC MIN SPANNING TREE consists of determining an anticipatory spanning tree T^* of G minimizing $E(G, T)$. Unfortunately, Lemma 1 does not derive a compact combinatorial characterization for the optimal anticipatory solution of PROBABILISTIC MIN SPANNING TREE. In particular, the form of the functional does not imply solution, for instance, of some well-defined weighted version of the (deterministic) MIN SPANNING TREE. This is due to the second term of the expression for $E(G, T)$ in Eq. 16. There, the “costs” assigned to the edges depend on the structure of the anticipatory solution chosen and of the present subgraph of G .

Revisit functional’s expression 16. For a vertex v_i , denote by $f(v_i)$ its father in T , by $p_{f(i)}$ the presence probability of $f(v_i)$ and by $A(v_i)$ the set of its ancestors in T . Then, Eq. 16 can be rewritten as:

$$\begin{aligned}
E(G, T) &= \sum_{v_i \in V \setminus \{v_1\}} p_i p_{f(i)} d_{if(v_i)} \\
&\quad + \sum_{v_i \in V \setminus \{v_1\}} \sum_{v_j \in A(v_i) \setminus \{f(v_i)\}} p_i p_j \prod_{v_k \in \mu[v_i, v_j]} (1 - p_k) d_{ij} \\
&= \sum_{v_i \in V \setminus \{v_1\}} C_i
\end{aligned} \tag{17}$$

where:

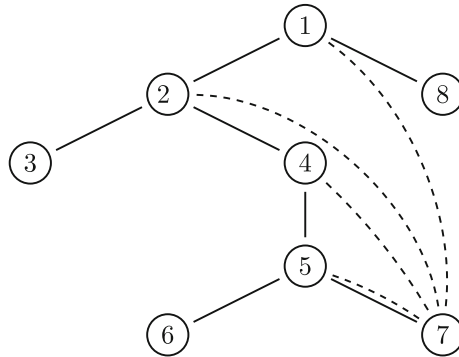
$$C_i = p_i p_{f(i)} d_{if(v_i)} + \sum_{v_j \in A(v_i) \setminus \{f(v_i)\}} p_i p_j \prod_{v_k \in \mu[v_i, v_j]} (1 - p_k) d_{ij} \tag{18}$$

and can be seen as the contribution of vertex v_i in $E(G, T)$.

As an example, assume that edge-weights in G are identical. We can show that, if $d_{ij} = d$, $(v_i, v_j) \in E$, then $C_i = d \times p_i$, $v_i \in V$. In order to give some intuition about it let us consider the anticipatory tree of Fig. 2, assume that edge weights in the input-graph are identical and equal to d and take, say, vertex 7. The contribution of it in Eq. 16 is:

$$\begin{aligned}
C_7 &= d \times p_7 [p_5 + p_4 (1 - p_5) + p_2 (1 - p_5) (1 - p_4) + (1 - p_5) (1 - p_4) (1 - p_2)] \\
&= d \times p_7
\end{aligned} \tag{19}$$

Fig. 2 About a reinterpretation of functional's expression 16



The same holds for the contribution of any other vertex in the tree.

Indeed, consider some vertex $v_i \in V$ and assume, for simplicity, that vertices in the path of T from v_1 to v_i are numbered from 1 to i . By writing down C_i and by some algebra as previously in Eq. 19 we derive $C_i = d \times p_i$. Hence, Eq. 17 becomes:

$$E(G, T) = d \times \sum_{i=2}^n p_i$$

Proposition 2 *If edge-weights are all identical, $E(G, T) = d \times \sum_{i=2}^n p_i$. In this case all the anticipatory solutions have the same value.*

Corollary 2 *If $p_1 = 1$, $p_i = p$, $i = 2, \dots, n$ and $d_{ij} = d$, $(v_i, v_j) \in E$, $i \neq j$, then, for any tree T spanning V , $E(G, T) = dp(n - 1)$.*

Discussion above can be generalized for deriving a general upper bound for $E(G, T)$ and for every anticipatory solution T . Set $D = \max\{d_{ij} : (v_i, v_j) \in E\}$. Then, Eq. 18 leads to $C_i \leq D \times p_i$ and Eq. 17 to $E(G, T) \leq D \times \sum_{i=2}^n p_i$ and the following corollary that will be used later holds.

Corollary 3 *If $D = \max\{d_{ij} : (v_i, v_j) \in E\}$ then, for any anticipatory solution T of PROBABILISTIC MIN SPANNING TREE, $E(G, T) \leq D \times \sum_{i=2}^n p_i$.*

We now study the complexity of PROBABILISTIC MIN SPANNING TREE under CLOSESTANCESTOR. We prove in what follows that its decision version, denoted by PROBABILISTIC MIN SPANNING TREE(K), and stated as: “given an edge-weighted complete graph $G(V, E)$, provided with a vertex-probability system $(p_i)_{i=1, \dots, n}$ with $p_1 = 1$ and a constant K , does there exist a tree T such that $E(G, T) \leq K$?”, where $E(G, T)$ is given by Eq. 16, is **NP**-complete. Before giving the proof (Proposition 3 just below) let us note that the reduction presented uses the same starting problem, the 3 EXACT COVER, as the analogous reduction of [4]. But, since we assume here that $p_1 = 1$ (that is not the case in [4]) and since the functional given in Eq. 16 includes in its expression addition of edges that do not make part of the a priori tree, the reduction of [4] does not work as it is. Indeed, as one can see in the proof of Proposition 3, the derived graph is different from that in the reduction of [4] as well

as the vertex probabilities and the arc-weights. All this makes that the proof of the reduction is based upon different technical arguments.

Proposition 3 **PROBABILISTIC MIN SPANNING TREE(K) is \mathbf{NP} -complete.**

Proof It is easy to see that **PROBABILISTIC MIN SPANNING TREE** $\in \mathbf{NP}$ (Corollary 1). In order to show completeness, we reduce **3 EXACT COVER** to **PROBABILISTIC MIN SPANNING TREE**. **3 EXACT COVER** is defined as follows: given a ground set X of size $3q$ and a collection \mathcal{E} of $3q$ subsets of X each of size 3, does there exist a sub-collection $\mathcal{E}' = \{S_1, \dots, S_q\}$ of \mathcal{E} such that $\bigcup_{i=1}^q S_i = X$? (obviously, \mathcal{E}' is a partition of X).

Consider an arbitrary instance $I(X, \mathcal{E})$ of **3 EXACT COVER**; we construct the following instance for **PROBABILISTIC MIN SPANNING TREE**:

- The vertex-set V is a set of $6q + 2$ vertices built by associating a vertex x_i with an element $x_i \in X$, a vertex y_j with a set $S_j \in \mathcal{E}$ and by adding a vertex r (playing the role of the omnipresent root) and a vertex s (representing the solution); for some positive fixed constant $p < 1/2$, vertices x_i are provided with probability p , vertices y_j with probability $1 - p$ and vertices r and s with probability 1;
- Edge-weights are defined as follows:
 - For every $S_j = \{x_{i_1}, x_{i_2}, x_{i_3}\}$, $j = 1, \dots, 3q$, $d_{ri_1} = d_{ri_2} = d_{ri_3} = 1$;
 - Edges linking s to vertices y_j have weight $M > 3p^2/2(1 - p)$ and those linking s to vertices of x_i have weight $M/p + 2$;
 - Edges linking r to vertices y_j as well as edge (r, s) have all weight 0, while edges linking r to vertices x_i have weight $M/p^2 + 1$;
 - All the other edges have arbitrarily large weight $B \gg M/p^2 + 1$;
- $K = q(M(1 + 2p) + 3p(p + 1))$.

It is easy to see that this reduction is polynomial. It is illustrated in Fig. 3 where, for readability, some edges, in particular those of weight B are omitted.

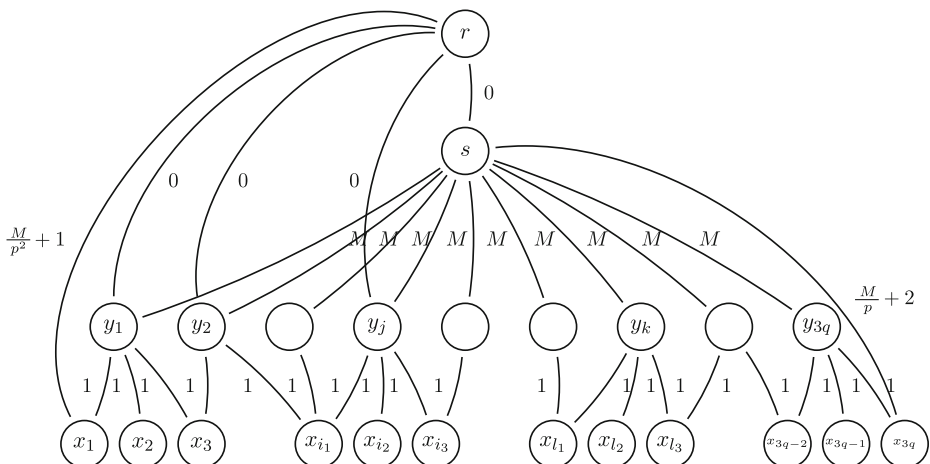


Fig. 3 An example for the reduction from **3 EXACT COVER** to **PROBABILISTIC MIN SPANNING TREE**

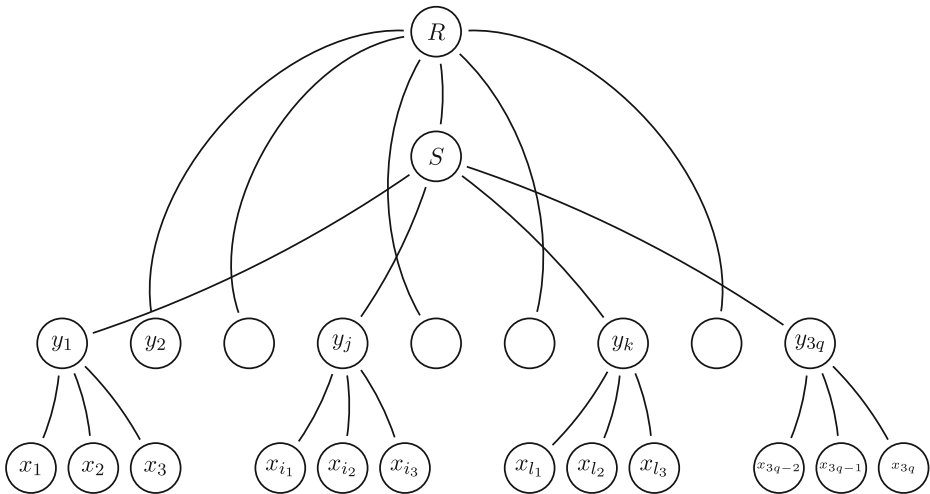


Fig. 4 The shape of T^*

We now prove that if 3 EXACT COVER admits a solution \mathcal{E}^* , then G has a minimum spanning tree T^* the shape of which is as in Fig. 4 and whose value is:

$$\begin{aligned} E(G, T^*) &= q \left[(1-p)M + 3p(1-p) + 3p(1-(1-p)) \left(\frac{M}{p} + 2 \right) \right] \\ &= q [M - pM + 3p - 3p^2 + 3pM + 6p^2] \\ &= q [M(1+2p) + 3p(p+1)] = K \end{aligned}$$

In other words, we are going to show that there exists a tree spanning V with expectation at most K , iff X admits a partition among the sets of \mathcal{E} .

The \Rightarrow part of the proof is easy. Given a solution \mathcal{E}^* for 3 EXACT COVER in $I(X, \mathcal{E})$, a solution T^* for PROBABILISTIC MIN SPANNING TREE is immediately built in G as shown in Fig. 4.

We prove the \Leftarrow part by inspection of all the possible trees spanning V . Let us do two easy preliminary remarks:

- Any tree using an edge of weight B has expectation greater than K ;
- Any tree where absent edges are replaced (following strategy CLOSEST_ANCESTOR) by edges of weight B also has expectation greater than K .

As a consequence, the trees that are to be inspected are those that any reconnection will not use edges of weight B , i.e., those that span the subgraph shown in Fig. 5. There exist 16 such trees illustrated in Figs. 6, 7, 8 and 9. For reasons of economy, in all these figures, only one out of the three children of each y_i is displayed. Note that tree T_1 in Fig. 6 corresponds to the optimal solution, whose full shape is shown in Fig. 4.

In all, trees in Figs. 6–9 have the following functionals (computed over their full shapes, i.e., considering all the three children of each y_i):

$$\begin{aligned} E(G, T_1) &= 1 \times 1 \times 0 + 1 \times (1-p) \times M + 3 \times (1-p) \times p \times 1 \\ &\quad + 3 \times p \times (1 - (1-p)) \times \left(\frac{M}{p} + 2\right) \\ &= M(1-p) + 3p(1-p) + 3pM + 6p^2 = M(1+2p) + 3p(p+1) \end{aligned}$$

$$E(G, T_2) = 3p \left(\frac{M}{p^2} + 1\right) = \frac{3}{p} \times M + 3p$$

$$E(G, T_3) = 3(1-p)p + 3p^2 \left(\frac{M}{p^2} + 1\right) = 3M + 3p$$

$$E(G, T_4) = 3p \left(\frac{M}{p} + 2\right) = 3M + 6p$$

$$E(G, T_5) = (1-p)M + 3p \left(\frac{M}{p^2} + 1\right) = M \left(1-p + \frac{3}{p}\right) + 3p$$

$$E(G, T_6) = (1-p)M + 3p \left(\frac{M}{p} + 2\right) = M(4-p) + 6p$$

$$\begin{aligned} E(G, T_7) &= (1-p)M + (1-p)p + 3p \left(\frac{M}{p^2} + 1\right) + p^2 \left(\frac{M}{p} + 2\right) \\ &= M \left(\frac{3}{p} + 1\right) + 4p + p^2 \end{aligned}$$

$$E(G, T_8) = (1-p)M + 3p(1-p) + 3p^2 \left(\frac{M}{p^2} + 1\right) = M(4-p) + 3p$$

$$E(G, T_9) = 3p \left(\frac{M}{p^2} + 1\right) + p \left(\frac{M}{p} + 2\right) + p(1-p) = M \left(1 + \frac{3}{p}\right) + 6p - p^2$$

$$\begin{aligned} E(G, T_{10}) &= p \left(\frac{M}{p} + 2\right) + 3p(1-p) + 3p^2 \left(\frac{M}{p^2} + 1\right) + (1-p)^2 M \\ &= M(5-2p+p^2) + 5p \end{aligned}$$

$$E(G, T_{11}) = (1-p)M + p \left(\frac{M}{p} + 2\right) + 3p \left(\frac{M}{p^2} + 1\right) = M \left(2-p + \frac{3}{p}\right) + 5p$$

$$E(G, T_{12}) = (1-p)M + 3p \left(\frac{M}{p} + 2\right) = M(4-p) + 6p$$

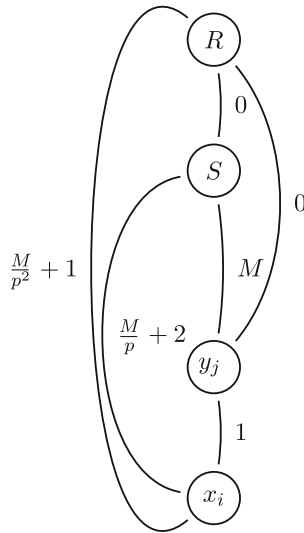
$$\begin{aligned} E(G, T_{13}) &= p(1-p) + 3p \left(\frac{M}{p} + 2\right) + (1-p)^2 M \\ &= M(3 + (1-p)^2) + 7p - p^2 \end{aligned}$$

$$E(G, T_{14}) = p(1-p) + 3p \left(\frac{M}{p^2} + 1\right) = M \frac{3}{p} + 4p - p^2$$

$$E(G, T_{15}) = (1-p)M + 3p \left(\frac{M}{p^2} + 1\right) = M \left(\frac{3}{p} + 1 - p\right) + 3p$$

$$E(G, T_{16}) = 3p \left(\frac{M}{p^2} + 1\right) + p \left(\frac{M}{p} + 2\right) = M \left(\frac{3}{p} + 1\right) + 5p$$

Fig. 5 The “interesting” subgraph of G



Trees $T_2, T_3, T_4, T_9, T_{10}, T_{13}$ and T_{14} correspond to the case where y_j , that is, the set S_j of \mathcal{E} is not part of a 3 EXACT COVER-solution (in other words, the element x_i of the ground set X is not covered by S_j). In this case, the best one can do for the corresponding spanning tree is to take edges (r, s) and (r, y_j) . The rest of the trees sketched in Figs. 6–9 correspond to the case where S_j covers x_i in some 3 EXACT COVER-solution.

In all, $E(G, T_1)$ is minimum over every other $E(G, T_i)$, $i = 2, \dots, 16$, and it is the only that is smaller than K . Tree T_1 corresponds to q subtrees as the first one in Fig. 6 but having three leaves plus $2q$ subtrees that are simple edges of the form (r, y_j) . Obviously, given T_1 a solution for 3 EXACT COVER is immediately

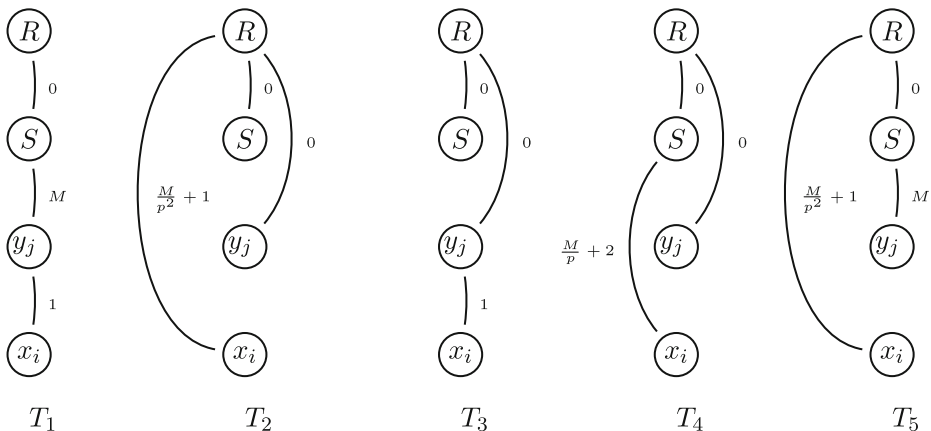


Fig. 6 Five out of 16 trees spanning component of Fig. 5

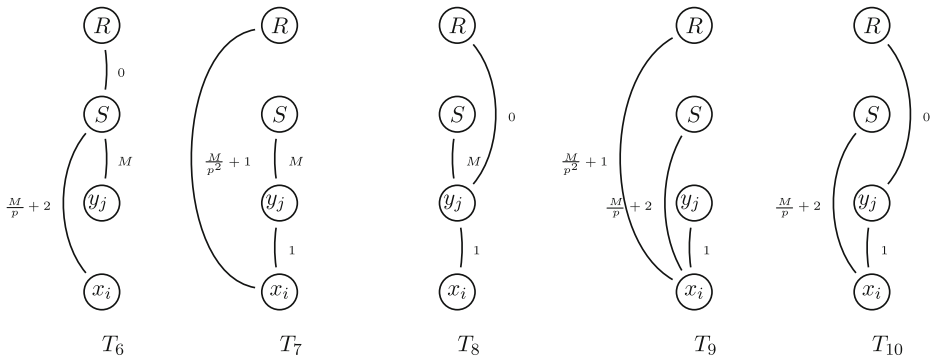


Fig. 7 Other 5 out of 16 trees spanning component of Fig. 5

computed by considering the q sets corresponding to y -vertices of T_1 that are not leaves. The proof of the proposition is now completed. \square

4.2 Stability of the Solutions: Stars and Paths as Anticipatory Solutions

We address in this section the question of “stability of the solutions” that can be stated as follows: “under which conditions and for what kinds of instances and/or particular structures of anticipatory solutions, an optimal solution for MIN SPANNING TREE remains optimal for PROBABILISTIC MIN SPANNING TREE?”. We study in this section some particular but natural cases carrying over assumptions either on the values of vertex-probabilities and/or edge-weights, or on the form of the anticipatory solution.

In what follows we deal with two particular structures of trees, the star (rooted at v_1) and the path. We first consider the star rooted at (the omnipresent) vertex v_1 and prove Proposition 4 just below. Let us note that a similar result is proved in [4], but its proof is not the same than the ours since, in Proposition 4, we handle the case of distinct vertex-probabilities while [4] handles the case of identical ones.

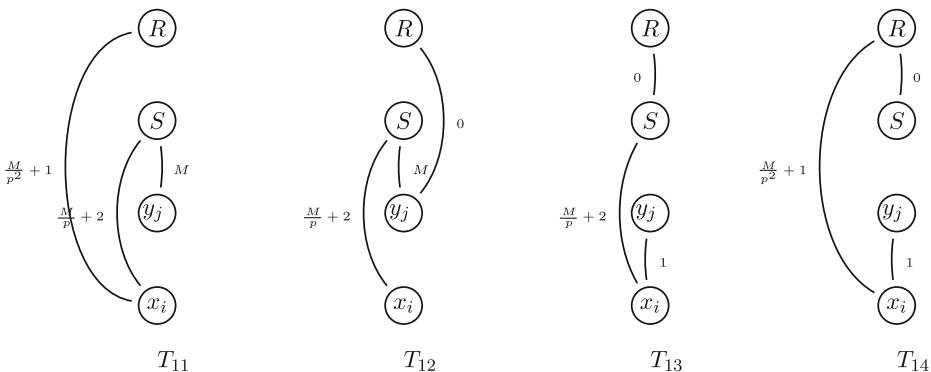
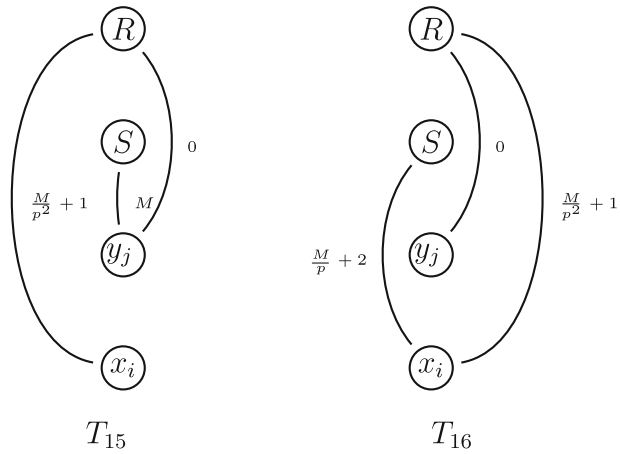


Fig. 8 Other 4 out of 16 trees spanning component of Fig. 5

Fig. 9 And the last 2 out of 16 trees spanning component of Fig. 5



Proposition 4 Let T be a star rooted at v_1 . If T is an optimal solution for MIN SPANNING TREE, then it is also an optimal anticipatory solution for PROBABILISTIC MIN SPANNING TREE.

Proof Recall that by Eq. 17, $E(G, T) = \sum_{v_i \in V \setminus \{v_1\}} C_i$ where C_i is given by Eq. 18. Observe now that, if the vertices of T are numbered in a dfs order (starting from the root) and if the set $A(v_i)$ of the ancestors of a vertex v_i in T is exactly the set $A(v_i) = \{v_1, v_2, \dots, v_{i-1}\}$, then C_i can be written as:

$$C_i = \sum_{j=1}^{i-1} p_i p_j d_{ij} \prod_{l=j+1}^{i-1} (1 - p_l) \quad (20)$$

Since the star T is a minimum spanning tree, it holds that, for any vertex i , $d_{1i} \leq d_{ij}$, for every $j \neq 1, i$. Hence, $C_i \geq p_i \times d_{1i} \times \sum_{j=1}^{i-1} p_j \prod_{l=j+1}^{i-1} (1 - p_l)$.

If we denote by C_i^T the contribution of vertex v_i in the functional $E(G, T)$ of the star T , then, for every i , $C_i^T = p_i \times d_{1i}$. So, in order to complete the proof of the proposition, we have to show that, for any $v_i \in V$, $C_i^T \leq C_i$, where C_i refers to every other spanning tree of G . For this, it suffices to prove that:

$$\sum_{j=1}^{i-1} p_j \prod_{l=j+1}^{i-1} (1 - p_l) \geq 1 \quad (21)$$

We show Eq. 21 by induction on i . For $i = 2$, the lefthand side of Eq. 21 is equal to $p_1 = 1$, so the inequality claimed is true. Suppose it true for $i = n$, i.e.:

$$\sum_{j=1}^{n-1} p_j \prod_{l=j+1}^{n-1} (1 - p_l) \geq 1 \quad (22)$$

Then, at range $n + 1$ it holds:

$$\begin{aligned}
 \sum_{j=1}^n p_j \prod_{l=j+1}^n (1 - p_l) &= \sum_{j=1}^{n-1} p_j \prod_{l=j+1}^n (1 - p_l) + p_n \\
 &= \sum_{j=1}^{n-1} p_j \prod_{l=j+1}^{n-1} (1 - p_l) \times (1 - p_n) + p_n \\
 &= (1 - p_n) \times \sum_{j=1}^{n-1} p_j \prod_{l=j+1}^{n-1} (1 - p_l) + p_n \\
 &\stackrel{\text{Eq. 22}}{\geq} (1 - p_n) + p_n = 1
 \end{aligned}$$

The proof of the proposition is now completed. \square

Unfortunately, in the case where optimal solution for MIN SPANNING TREE is a path, optimality of such a solution for PROBABILISTIC MIN SPANNING TREE cannot be derived as previously in the case of stars.

Indeed consider a complete graph G , the adjacency matrix of which is given in Table 1 and its vertex-probability system is $(1, p, \dots, p)$, with $p < (K - 2)/(K - 1)$ and $K \geq n$. Optimal MIN SPANNING TREE-solution in G is unique and is the path $P = (1, 2, \dots, n - 1)$ with value $n - 1$. The functional $E(G, P)$ of path P is $E(G, P) = (2n - 3)p + (K - n)p^2 - (K - 2)p^3$. On the other hand, the unique optimal anticipatory solution for PROBABILISTIC MIN SPANNING TREE is the tree T^* of Fig. 10 with functional's value $E(G, T^*) = (2n - 3)p + (2 - n)p^2 + p^3 < E(G, P)$, when $p < (K - 2)/(K - 1)$.

4.3 PROBABILISTIC METRIC MIN SPANNING TREE

In this section, we study PROBABILISTIC METRIC MIN SPANNING TREE problem, that is PROBABILISTIC MIN SPANNING TREE in metric complete graphs, i.e., in complete graphs whose edge-weights satisfy the triangular property that can be expressed as follows: if (v_i, v_j, v_k) is a K_3 in G , then $d_{ab} \leq d_{ac} + d_{bc}$, for any permutation (a, b, c) of $\{i, j, k\}$.

Note that, as it can be seen from the proof of Proposition 3, it does not apply in the case of metric graphs. Indeed, the complexity status of PROBABILISTIC METRIC MIN

Table 1 The adjacency matrix of a graph G where optimal solutions for MIN SPANNING TREE and PROBABILISTIC MIN SPANNING TREE do not coincide

	v_1	v_2	v_3	\dots	v_{n-2}	v_{n-1}	v_n
v_1	0	1	2	\dots	2	2	2
v_2	1	0	1	\dots	2	2	2
v_3	2	1	0	\dots	2	2	2
\vdots	\vdots	\vdots	\vdots	\dots	\vdots	\vdots	\vdots
v_{n-2}	2	2	2	\dots	0	1	K
v_{n-1}	2	2	2	\dots	1	0	1
v_n	2	2	2	\dots	K	1	0

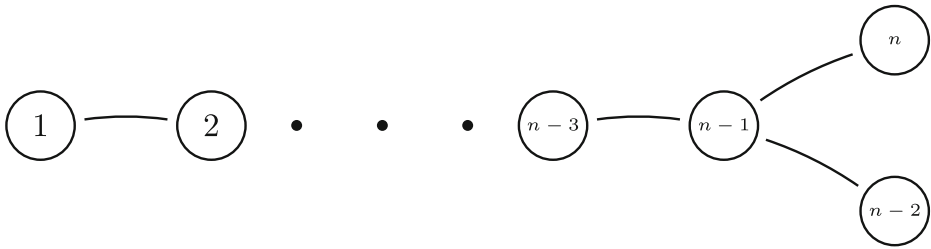


Fig. 10 The optimal solution of the graph of Table 1

SPANNING TREE remains open, even if we feel that this variant is also **NP**-complete. In what follows we give approximation results for the general case of PROBABILISTIC METRIC MIN SPANNING TREE as well as for a natural subcase, namely PROBABILISTIC MIN SPANNING TREE 1,2, where edge weights are either 1, or 2.

We denote by \hat{T} and T^* a tree computed by Kruskal's algorithm (i.e., a minimum spanning tree of G) and an optimal anticipatory solution for PROBABILISTIC METRIC MIN SPANNING TREE (under CLOSEST_ANCESTOR), respectively, and we assume that they are represented as sets of edges. Observe also that, by the metric property the weight of any edge of G is smaller than the weight of any spanning tree of G . Indeed, let (v_i, v_j) be any edge of G , T be some spanning tree of G and $P(v_i, v_j)$ be the unique path from v_i to v_j in T . According to the metric hypothesis, $d_{ij} \leq w(P(v_i, v_j)) \leq m(G, T)$, where $w(P(v_i, v_j))$ is the weight of the path $P(v_i, v_j)$. In what follows, given a set Q of (weighted) edges, we denote by $w(Q)$ its total weight.

4.3.1 General Metric Case

For any $V_i \subseteq V$, we set $G_i(V_i, E_i) = G[V_i]$. Let \hat{T}_i be the spanning trees on G_i resulting from the application of strategy CLOSEST_ANCESTOR on \hat{T} . We prove the following result.

Proposition 5 PROBABILISTIC METRIC MIN SPANNING TREE is approximable in polynomial time within ratio $2(n-2)/(\sqrt{5}-1)$.

Proof In order to prove the result we will first prove that PROBABILISTIC METRIC MIN SPANNING TREE is approximable within ratio $\min\{p(n-2)^2/2 + (n-2), 2/p\}$.

For the first ratio, the solution considered is a deterministic minimum spanning tree \hat{T} . Denote by X the set of vertices connected to the root v_1 and by $A(i)$ the set of ancestors of a vertex v_i in \hat{T} . Then, for every node $v_i \in X$, its contribution to the functional is given by:

$$C_i = p d_{i1} \quad (23)$$

For every node $v_i \in V \setminus (X \cup \{v_1\})$, its contribution C_i can be seen as the sum of two terms C'_i and C''_i , where C'_i represents the part of its contribution with respect to v_1 and is bounded by:

$$C'_i \leq p(1-p)d_{i1} \quad (24)$$

On the other hand, C_i'' the part of the contribution of v_i with respect to another vertex of $V \setminus \{v_1\}$ is:

$$C''(i) \leq p^2 d_{if(i)} + p^2(1-p) \sum_{\substack{v_j \in (A(i) \setminus \{v_1\}) \\ v_j \neq f(v_i)}} d_{ij} \quad (25)$$

where, we denote by $f(v_i)$ the father of v_i in \hat{T} and by $d_{if(i)}$ the weight from v_i to $f(v_i)$. Note also that, by the metric property, no edge $(v_i, v_j) \in E$ can have a weight greater than the weight $w(\hat{T})$ of any spanning tree \hat{T} . Taking this into account as well as the fact that

$$w(\hat{T}) = \sum_{i=2, \dots, n} w(v_i, f(v_i))$$

and summing Eqs. 24, 23 and 25 for $v_i \in V \setminus \{v_1\}$, we get:

$$\begin{aligned} E(G, \hat{T}) &\leq p \sum_{v_i \in X} d_{i1} + \sum_{v_i \in V \setminus (X \cup \{v_1\})} p(1-p)d_{i1} + \sum_{v_i \in V \setminus (X \cup \{v_1\})} p^2 d_{if(i)} \\ &\quad + p^2(1-p) \sum_{v_i \in V \setminus (X \cup \{v_1\})} \sum_{\substack{v_j \in (A(i) \setminus \{v_1\}) \\ v_j \neq f(v_i)}} d_{ij} \\ &\leq pw'(\hat{T}) + \sum_{v_i \in V \setminus (X \cup \{v_1\})} p(1-p)w(\hat{T}) + \sum_{v_i \in V \setminus (X \cup \{v_1\})} p^2 w''(\hat{T}) \\ &\quad + p^2(1-p) \sum_{v_i \in V \setminus (X \cup \{v_1\})} \sum_{\substack{v_j \in (A(i) \setminus \{v_1\}) \\ v_j \neq f(v_i)}} w(\hat{T}) \end{aligned} \quad (26)$$

where $w'(\hat{T})$ is the sum of the weights of the edges of the first level of \hat{T} , i.e., of the edges linking vertices of X to v_1 and $w''(\hat{T})$ is the sum of the weights of some edges of \hat{T} having at least one endpoint in the set $V \setminus (X \cup \{v_1\})$ (corresponding to the first term of Eq. 25). So, quantities $w'(\hat{T})$ and $w''(\hat{T})$ concern distinct sets of edges of \hat{T} and, furthermore:

$$w'(\hat{T}) + w''(\hat{T}) \leq w(\hat{T}) \quad (27)$$

Observe now that the number of terms in the fourth term in Eq. 26 is smaller than the edges of the (complete) subgraph induced by $V \setminus (X \cup \{v_1\})$, i.e., smaller than $|V \setminus (X \cup \{v_1\})|^2/2 = (n - |X| - 1)^2/2$. Then, combining Eqs. 26 and 27 and using the facts that $p^2 \leq p$, $|X| \geq 1$ and $1 - p \leq 1$, we get:

$$\begin{aligned} E(G, \hat{T}) &\leq pw(\hat{T}) + (n - |X| - 1)pw(\hat{T}) + p^2 \frac{(n - |X| - 1)^2}{2} w(\hat{T}) \\ &\leq pw(\hat{T}) + (n - 2)pw(\hat{T}) + p^2 \frac{(n - 2)^2}{2} w(\hat{T}) \end{aligned} \quad (28)$$

Combination of Eqs. 28 and 15, immediately leads to:

$$\frac{E(G, \hat{T})}{E(G, T^*)} \leq 1 + (n - 2) + p \frac{(n - 2)^2}{2} \approx p \frac{(n - 2)^2}{2} + (n - 2) \quad (29)$$

Consider now a left-to-right dfs ordering of the vertices of \hat{T} , the non-elementary path produced by this dfs traversal of \hat{T} and observe that in this traversal any edge is encountered at most twice. This path can be transformed into an elementary Hamiltonian path T' by using suitable shortcuts as shown in Fig. 11b where these shortcuts are the boldfaced lines (edges) $(5, 6), (8, 9), (9, 10), (10, 11), (12, 13)$ and the whole path T' is simply the path $(i, i + 1), i = 1, \dots, 12$ (vertices of \hat{T} in Fig. 11a are numbered in a left-to-right dfs order). By the metric property and the dfs numbering of the vertices of the tree, the weight of a shortcut $(i, j), i < j$, is bounded above by the total weight of the path of \hat{T} that starting from i “climbs” the tree up to the lowest common ancestor k of i and j and then goes down to j . With respect to this shortcut, edges of the path from k to i will count at most twice in T' (either as real edges of T' , or for shortcut’s bound). Furthermore, they will not further be encountered in any other shortcut. In Fig. 11b, the small lines crossing an edge of \hat{T} indicate the number of times this edge encounters in T' . So, since T' is also a spanning tree for G :

$$m(G, T') \leq 2 \text{opt}(G) \quad (30)$$

Since T' is a Hamiltonian path having the omnipresent vertex v_1 as one of its endpoints, any modification of it by the modification strategy `CLOSEST_ANCESTOR` due to the absence of any subset $V_i \subseteq V$, will produce a path T'_i that always remains Hamiltonian for the graph $G'_i = G[V \setminus V_i]$. By the metric hypothesis, for any $V_i \subseteq V$:

$$m(G'_i, T'_i) \leq m(G, T') \quad (31)$$

Then, the functional associated with T' can be written as:

$$\begin{aligned} E(G, T') &= \sum_{V_i \subseteq V} \Pr[V \setminus V_i] m(G'_i, T'_i) \\ &\stackrel{\text{Eq. 31}}{\leq} m(G, T') \times \sum_{V_i \subseteq V} \Pr[V \setminus V_i] \end{aligned} \quad (32)$$

$$= m(G, T') \stackrel{\text{Eq. 30}}{\leq} 2 \text{opt}(G) \stackrel{\text{Eq. 15}}{\leq} \frac{2}{p} \times E(G, T^*) \quad (33)$$

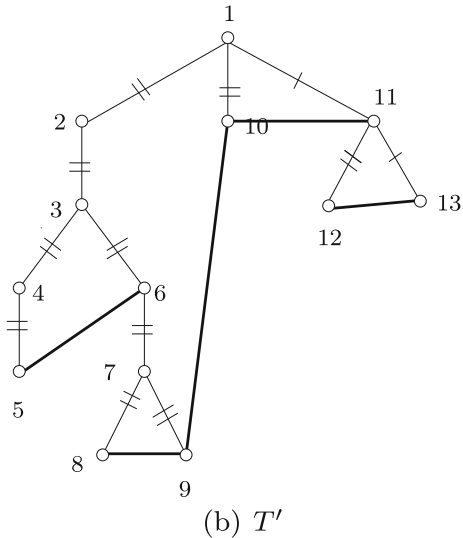
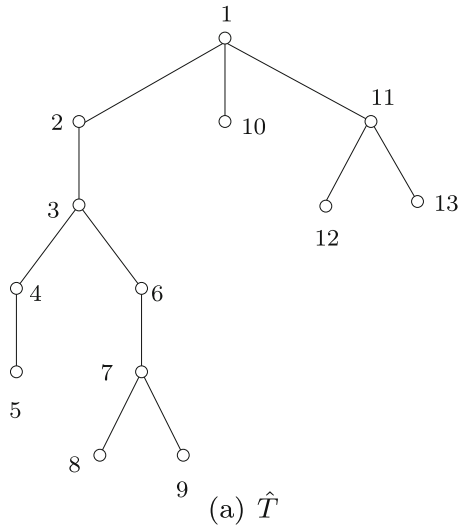
So, T' is a $(2/p)$ -approximation for `PROBABILISTIC METRIC MIN SPANNING TREE` and taking the best (with respect to the functional) among \hat{T} and T' achieves, by Eqs. 29 and 32 approximation ratio $\min\{p(n-2)^2/2 + (n-2), 2/p\}$ as claimed in the beginning of the proof.

Observe now that $p(n-2)^2/2 + (n-2)$ increases with p , while quantity $2/p$ decreases with p . Equality is achieved when these two quantities become equal, i.e., for $p = (\sqrt{5} - 1)/(n - 2)$. In this case, the approximation ratio of both solutions \hat{T} and T' becomes $2(n-2)/(\sqrt{5} - 1)$. \square

4.3.2 PROBABILISTIC MIN SPANNING TREE 1,2

We now focus on a particular but natural and well-studied class of metric complete graphs where edge weights are either 1 or 2. It is easy to see that any such graph is metric.

Fig. 11 A tree \hat{T} and the Hamiltonian path T' produced from it



We first prove the following easy approximation result for PROBABILISTIC MIN SPANNING TREE 1,2 that will be subsequently improved.

Proposition 6 *A minimum spanning tree of G is a $(2 - p)$ -approximation for PROBABILISTIC MIN SPANNING TREE 1,2.*

Proof Consider a minimum spanning tree \hat{T} of G computed, for instance, by Kruskal's algorithm. Then, the following fact holds.

Claim If \hat{T} has k edges of weight 2, then in $E(G, T^*)$, (i.e., the optimal value for PROBABILISTIC MIN SPANNING TREE 1,2 under CLOSEST_ANCESTOR) there exist

at least k vertices with contribution $2p$ (recall that by Eq. 17, the functional can be defined as the sum of the contributions of the vertices, each such contribution expressed by Eq. 20).

Indeed, if \hat{T} has k edges of weight 2, then, by construction of \hat{T} , there exist $k + 1$ distinct connected components V_i in the graph $G'(V, E')$ (that contains only the edges of G of weight 1); edges between any two vertices of any two of these components have weight 2 (suppose, w.l.o.g., that the omnipresent vertex v_1 belongs to V_1). So, the following holds:

$$\forall i \in \{1, \dots, k+1\}, \forall v_j \in V_i, v_k \notin V_i, d_{jk} = 2 \quad (34)$$

Assume now a bfs on \hat{T} , starting from v_1 and denote by u_i the first-visited vertex of V_i (obviously, $u_1 = v_1$). For every $i > 1$, all the vertices visited before u_i (a fortiori those on the path from v_1 to u_i), do not belong to V_i ; hence, by Eq. 34, every edge linking u_i to any of these vertices has weight 2. So, for $i = 2, \dots, k+1$, the contribution $C(u_i)$ of u_i to $E(G, \hat{T})$ is $2p$.

Discussion above holds for any spanning tree of G and a fortiori for T^* ; hence, truth of the claim is immediately derived.

As a consequence of the discussion above, T^* has at least k vertices with contribution $2p$ to the functional; observe furthermore that the contribution of any of the other $n - k - 1$ vertices (recall that vertex v_1 has null contribution) is at least p . So:

$$E(G, T^*) \geq 2p \times k + (n - k - 1) \times p = p \times (n - 1 + k) \quad (35)$$

Revisit now \hat{T} and consider that a vertex is “covered” by the edge of \hat{T} entering it in the bfs visiting. Based upon the discussion above, there are k vertices covered by edges of weight 2 and $n - k - 1$ vertices covered by edges of weight 1. Consider a vertex w covered by an edge of weight 1. It is easy to see from Eq. 20 that its contribution $C(w)$ is at most $p(2 - p)$ (considering that all the other edges incident to it in G have weight at most 2). So:

$$E(G, \hat{T}) \leq p \times (2 - p) \times (n - 1 - k) + 2pk = p(n - 1) \times (2 - p) + p^2 \times k \quad (36)$$

Combining Eqs. 35 and 36 we get after some easy algebra: $E(G, \hat{T})/E(G, T^*) \leq 2 - p$ as claimed. \square

In what follows we refine the result above. For this we consider an execution of Kruskal’s algorithm that starts by introducing in the tree all the edges of weight 1 incident to the vertex v_1 . Let us denote by \tilde{T} the spanning tree so constructed; notice that \tilde{T} is a minimum spanning tree for G .

Proposition 7 *The tree \tilde{T} approximates T^* within ratio:*

$$\frac{1 + (2 - p)(n - 2)}{n - 1 + \frac{(1-p)^2 - (1-p)^n}{p}}$$

Proof Recall that, by Proposition 6, if the value of a minimum spanning tree \tilde{T} is equal to $n + k$ then, for every spanning tree T there exist at least k (exactly k for \tilde{T}) vertices each contributing with $2p$ to the functional associated with T . For

simplicity, assume that this number is exactly k for T also (if not, i.e., if we have more than k such vertices, the bounds derived below are even better). Among the remaining (less than) $n - 1 - k$ vertices of T let us denote by n_1 the number of those that in G are connected to v_1 by edges of weight 1 and by U_2 the set of those that in G are connected to v_1 by edges of weight 2; set finally $n_2 = |U_2|$. Obviously, $n - 1 = n_1 + n_2 + k$.

Consider a vertex $v_i \in U_2$, suppose that its height is i and recall that, by definition, $d_{i1} = 2$. Then, the contribution C_i of v_i (see also Eq. 20) in $E(G, T)$ is at least:

$$C_i \geq p(1 + (1 - p)^{i-1}) \quad (37)$$

where in Eq. 37 it has been assumed that only edge (v_1, v_i) has weight 2, while every other edge on the path from v_1 to v_i in T has weight 1.

Number the vertices of T as they appear in a bfs ordering of T and let $\ell_1, \ell_2, \dots, \ell_{n_2}$ be the bfs numbers of the vertices of U_2 . Fix a vertex $v_i \in U_2$ with bfs number, say, ℓ_i and notice that the $n_2 - i$ vertices of U_2 numbered $\ell_{i+1}, \dots, \ell_{n_2}$ have all height at least equal to the height of v_i . The height of v_i is at most $n - 1 - (n_2 - i) = n - 1 - n_2 + i$; hence, Eq. 37 becomes:

$$C_i \geq p(1 + (1 - p)^{n-n_2+i-2}) \quad (38)$$

From Eq. 38 and denoting by $C(U_2)$ the total contribution of the vertices of U_2 in $E(G, T)$, we get:

$$\begin{aligned} C(U_2) &\geq \sum_{i=n-n_2}^{n-1} p \times (1 + (1 - p)^{i-1}) = pn_2 + p \times \sum_{i=n-n_2}^{n-1} (1 - p)^i \\ &= p \times \left(n_2 + \frac{(1 - p)^{n-n_2} - (1 - p)^n}{p} \right) \end{aligned} \quad (39)$$

What it has been discussed until now holds for any spanning tree T , a fortiori for the optimal anticipatory tree T^* . So, using Eq. 39, the functional $E(G, T^*)$ becomes:

$$E(G, T^*) \geq p \times n_1 + p \times \left(n_2 + \frac{(1 - p)^{n-n_2} - (1 - p)^n}{p} \right) + 2p \times k \quad (40)$$

On the other hand, consider the tree \tilde{T} and recall that, by its construction, n_1 of its vertices are directly linked to v_1 . Then, their contribution to $E(G, \tilde{T})$ is equal to pn_1 . According to the proof of Proposition 6, the contribution of any of the n_2 vertices linked in G to v_1 by edges of weight 2 but covered in \tilde{T} by edges of weight 1 is at most $p(2 - p)$ and the contribution of any of the remaining k vertices (covered in \tilde{T} by edges of weight 2) is $2p$. Putting all this together, we derive:

$$E(G, \tilde{T}) \leq p \times n_1 + p(2 - p) \times n_2 + 2p \times k \quad (41)$$

Combining Eqs. 40 and 41 we get:

$$\frac{E(G, \tilde{T})}{E(G, T^*)} \leq \frac{p \times n_1 + p(2-p) \times n_2 + 2p \times k}{p \times n_1 + p \times \left(n_2 + \frac{(1-p)^{n-n_2} - (1-p)^n}{p} \right) + 2p \times k} \quad (42)$$

$$\leq \frac{n_1 + (2-p) \times n_2}{n_1 + n_2 + \frac{(1-p)^{n-n_2} - (1-p)^n}{p}} \quad (43)$$

$$\leq \frac{n_1 + (2-p)(n-1-n_1)}{n-1 + \frac{(1-p)^{n_1-1} - (1-p)^n}{p}} \leq \frac{1 + (2-p)(n-2)}{n-1 + \frac{(1-p)^2 - (1-p)^n}{p}} \quad (44)$$

where the rightmost expression of Eq. 43 is due to the fact that ratio's expression is decreasing with $k \geq 0$, the first expression of Eq. 44 is due to the fact that since k is set to 0, $n-1 = n_1 + n_2$ and the second expression of Eq. 44 is due to the fact that the first expression is decreasing with n_1 but k and n_1 cannot be simultaneously equal to 0 (if $n_1 = 0$, then no vertex of G is linked to v_1 with an edge of weight 1, so $k \geq 1$), hence, $n_1 \geq 1$. This proves the ratio claimed.

For tightness, consider the graph of Fig. 12. There, $\forall i \in \{3, \dots, n\}$, $d_{1i} = 2$; every other edge has weight 1. So, any tree where v_1 is linked to v_2 is a tree \tilde{T} , potentially built by an execution of Kruskal's algorithm as that considered in the proposition. For instance, such a tree can be the tree of Fig. 13a. It is easy to see that the value of the functional of this tree is equal to $p(1 + (2-p)(n-2))$. On the other hand, the optimal anticipatory solution T^* is shown in Fig. 13b; the value of the functional associated with T^* is equal to $p(n-1) + p \sum_{i=1}^{n-2} (1-p)^i = p(n-1) + ((1-p) - (1-p)^{n-1})$. \square

One can see that when p is fixed (i.e., independent on n), the approximation ratio achieved is strictly better than 2. On the other hand, when $p \sim 1/n$, then since $\lim_{n \rightarrow +\infty} (1-p)^n/p = n/e$, the ratio claimed in Proposition 7 tends to 1.225, for large values of n . If $p \sim 1/n^k$, $k > 1$, then for large values of n , this ratio tends to 1. Finally, if $p \sim 1/n^k$, $k < 1$, then (always for large values of n) the ratio is asymptotically equal to 2.

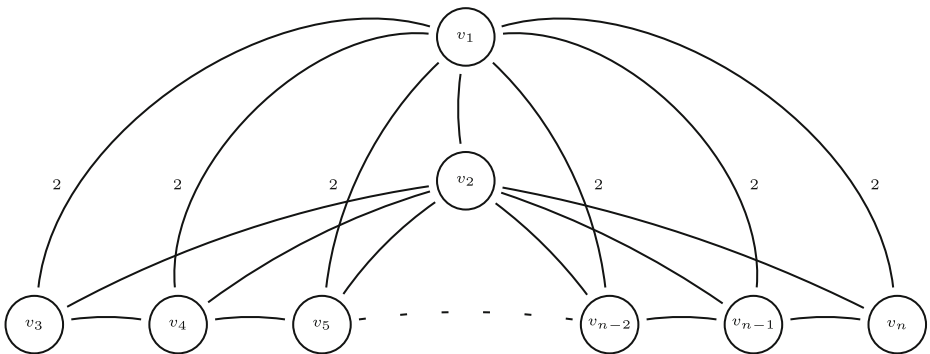


Fig. 12 A graph where the ratio claimed in Proposition 7 is tight

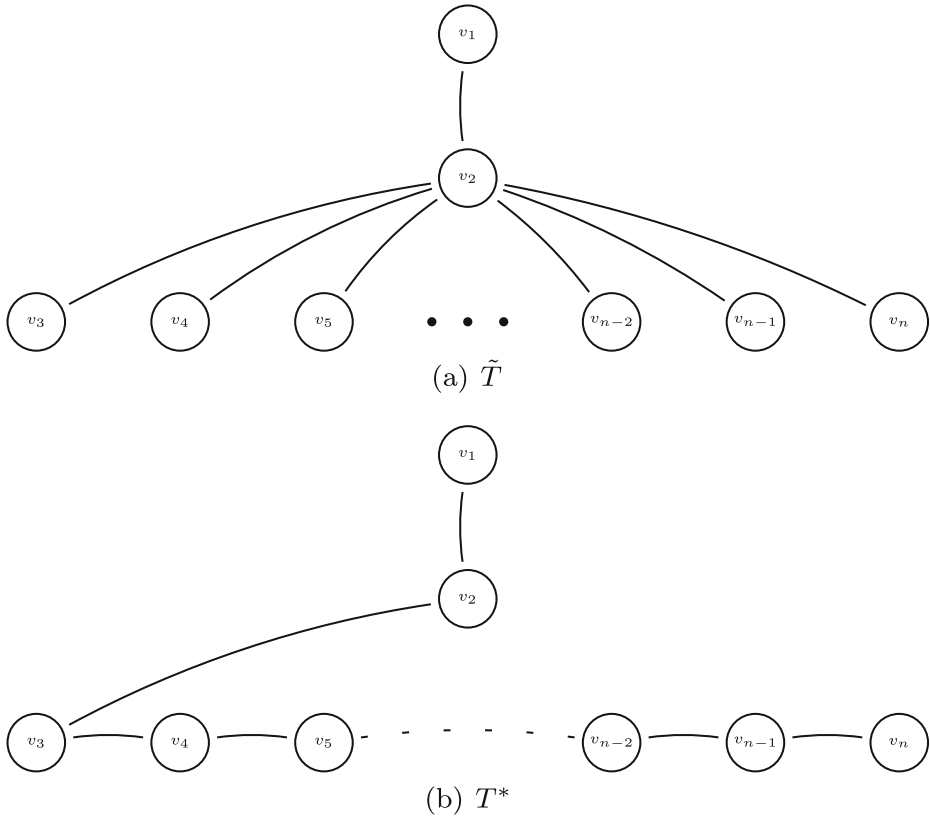


Fig. 13 The trees \tilde{T} and T^* from the graph of Fig. 12

5 PROBABILISTIC MIN SPANNING TREE under Modification Strategy ROOT

We will now analyze PROBABILISTIC MIN SPANNING TREE under another modification strategy. In this strategy, denoted by ROOT, present vertices whose fathers are absent are connected to the omnipresent root (see Fig. 14 for an example). Clearly, its complexity is $O(n)$. ROOT is very natural in terms of network design where, for instance, in case of failure of intermediate servers, surviving servers are connected to the central one in order to ensure total global connectivity of the remaining network.

5.1 The Complexity of PROBABILISTIC MIN SPANNING TREE

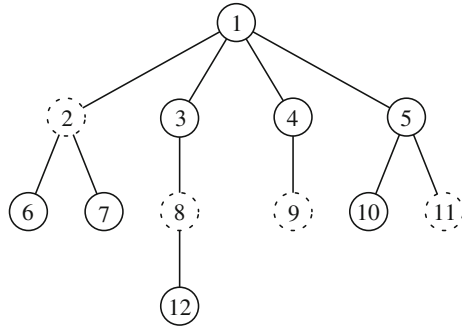
With respect to ROOT, the contribution C_i of a vertex v_i can be written as follows:

$$C_i = C(v_i, T) = p_i (p_{f(i)} d_{i, f(i)} + (1 - p_{f(i)}) d_{i, 1})$$

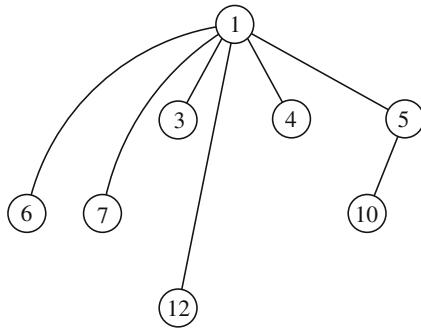
where $v_{f(i)}$ is the father of v_i in T . As previously, summing the C_i 's, for each vertex of the graph, leads to the functional associated with the strategy ROOT:

$$E(G, T) = \sum_{v_i \in V} (p_i (p_{f(i)} d_{i, f(i)} + (1 - p_{f(i)}) d_{i, 1})) \quad (45)$$

Fig. 14 An anticipatory spanning tree T and its adjustment



(a) An initial tree $T \dots$



(b) \dots and its modification by ROOT

The value of $E(G, T)$ can be computed in polynomial time. We will now show that PROBABILISTIC MIN SPANNING TREE under ROOT belongs to **P**, by specifying a polynomial algorithm that solves this problem optimally.

Proposition 8 PROBABILISTIC MIN SPANNING TREE under ROOT can be optimally solved in polynomial time.

Proof Transform the initial graph $G(V, E)$ into a directed one $G_d(V, A)$ having the same vertex-set as follows: for each edge (v_i, v_j) of E ($i, j \neq 1$), add two arcs (v_i, v_j) and (v_j, v_i) in G_d with weights:

$$d'_{i,j} = p_i(p_j d_{i,j} + (1 - p_j) d_{i,1})$$

$$d'_{j,i} = p_j(p_i d_{j,i} + (1 - p_i) d_{j,1})$$

and for each edge (v_1, v_i) of E , add one arc (v_1, v_i) in G_d with the same weight. Note that the number of arcs in G_d is at most $2|E|$.

Run in G_d Edmonds' polynomial algorithm for the ARBORESCENCE problem ([14, 15, 20]; this problem consists of finding a minimum-weight spanning tree in a directed graph), taking v_1 as root. It returns a rooted tree of minimum total edge-weight spanning all the vertices of G_d . This algorithm consists of determining the minimum weight arc entering each vertex (except for the root), and adding this edge to the solution. At this point, the solution has $n - 1$ arcs, but may contain loops. If

the solution does not contain loops, then it is an optimal arborescence, otherwise loops are successively broken in an optimal way. Retain this tree as anticipatory solution for G (simply delete orientations of arcs and unify the so obtained duplicate edges). Given the edge-weight function defined in G_d , this tree also minimizes the functional $E(G, T, \text{ROOT})$, i.e., it is optimal for PROBABILISTIC MIN SPANNING TREE. \square

5.2 Stability of the Solutions

5.2.1 Same Probabilities for All Vertices

We first focus on the case where all vertices have the same presence-probability, p (except for v_1 which still has probability 1). We show that in such graphs, the deterministic and probabilistic solutions coincide, which is not true anymore when probabilities are distinct.

Proposition 9 *Let G be an instance of PROBABILISTIC MIN SPANNING TREE problem under strategy ROOT, where all vertices, except for the root, have the same probability p . On such an instance, the optimal tree \hat{T} and optimal a priori tree T^* coincide.*

Proof In an instance where all vertices are equiprobable, the general functional formula given in Eq. 45 can be reformulated as follows:

$$E(G, T) = \sum_{v_i \in V} (p (pd_{i, f(i)} + (1 - p) d_{i, 1})) \quad (46)$$

In this case, notice that the second member of the sum does not depend on the structure of the tree. Thus, one can rewrite the functional as:

$$E(G, T) = p \left(\sum_{(v_i, v_j) \in T} pd_{i, j} + \sum_{v_j \in V} (1 - p) d_{1, j} \right) \quad (47)$$

The first term in the parentheses of Eq. 47 is p times the value of the spanning tree T , while the second term is the weight of the star S rooted at v_1 (that is constant for a given graph), i.e.:

$$E(G, T) = p (pd(T) + (1 - p) d(S)) \quad (48)$$

So, in order to minimize $E(G, T)$ in Eq. 48, it suffices to minimize $d(T)$. \square

5.2.2 Star as Deterministic Solution

Let us now consider the particular case where the anticipatory solution is a star rooted at v_1 . Then, the following result holds.

Proposition 10 *Let T be a star rooted at v_1 . If T is an optimal solution for MIN SPANNING TREE then it is also an optimal anticipatory solution for PROBABILISTIC MIN SPANNING TREE under ROOT.*

Proof Consider G_d as it has been built in the proof of Proposition 8. If the optimal solution for MIN SPANNING TREE in G is a star, then:

$$\forall i, j \neq 1, d_{1,j} \leq d_{i,j} \quad (49)$$

According to the definition of the edge-weight function d' in G_d , it can be easily seen that the property expressed by Eq. 49 extends also to G_d : $\forall i, j \neq 1, d'_{1,j} \leq d'_{i,j}$.

Since edges of G incident to v_1 are not duplicated when they become arcs in G_d , in the case of stars rooted at v_1 the minimum-weight arc for each vertex v_i will be the arc (v_1, v_i) and no loop will appear. Thus, in this case, the probabilistic optimum is the same star. \square

6 ROOT vs. CLOSEST_ANCESTOR

It has been shown that ROOT leads to a probabilistic model much easier to analyze and optimize than that induced by CLOSEST_ANCESTOR. But the optimum returned by the former might not be as good in terms of expected value as that returned by CLOSEST_ANCESTOR. Naturally, the choice between ROOT and CLOSEST_ANCESTOR will depend on the real world application so-modeled. But still, even if PROBABILISTIC MIN SPANNING TREE under ROOT is polynomial, comparing these two strategies in terms of quality of their respective returned solutions seems to be interesting to evaluate the (eventual) price of using ROOT (that derives a polynomial PROBABILISTIC MIN SPANNING TREE) vs. CLOSEST_ANCESTOR.

Proposition 11 *Let $E(G, T_R^*)$ and $E(G, T_{CA}^*)$ be the expected values of optimal anticipatory optima associated with ROOT and CLOSEST_ANCESTOR, respectively. Even in metric graphs, the ratio between these two values is unbounded.*

Proof Let us consider a graph G of order n , some ordering on its vertices and assume that the optimal solution for MIN SPANNING TREE is a path. In this graph, vertices are equiprobable and weights are as follows:

$$\begin{aligned} d_{1,i} &= n + i - 1 & \forall i \neq 1 \\ d_{i,j} &= j - i & \forall i, j \quad 1 < i < j \leq n \end{aligned} \quad (50)$$

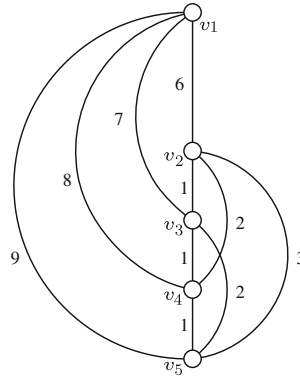
It can be easily seen that such graph is metric and that the path $P = (v_1, v_2, \dots, v_n)$ is a minimum spanning tree (see Fig. 15 for an example with $n = 5$).

From Proposition 9, P is also the optimal anticipatory solution for PROBABILISTIC MIN SPANNING TREE under ROOT, since vertices are equiprobable. The weight of P is $2n - 1$. So, revisiting Eq. 48, its expected value is:

$$\begin{aligned} E(G, T_R^*) &= p \left(p(2n - 1) + (1 - p) \sum_{(v_i) \in V} (n + j - 1) \right) \\ &= p \left(p(2n - 1) + (1 - p) \left(n - 1 + \frac{n(n - 1)}{2} \right) \right) \end{aligned} \quad (51)$$

Consider now P as the anticipatory solution for CLOSEST_ANCESTOR. Recall that this strategy reconnects each vertex to its closest present ancestor, so that, in G , any

Fig. 15 A graph G with $n = 5$



adjusted tree will be a path, with weight at most $d(P) = 2n - 1$. Figure 16 illustrates this assumption and depicts all the paths generated by `CLOSEST_ANCESTOR`, each of which has weight at most $2n - 1 = 7$.

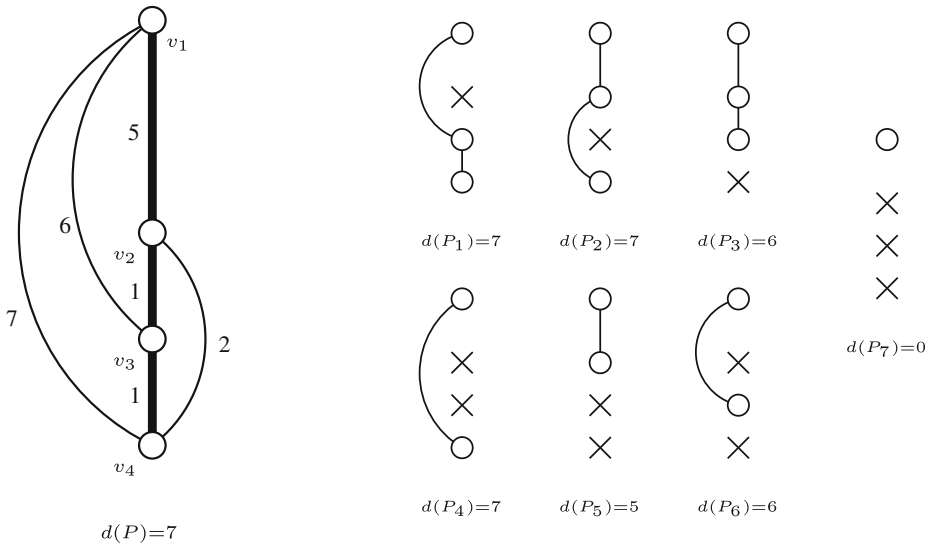
Thus, a loose (but sufficient) lower bound for $E(G, T_{CA}^*)$ is:

$$E(G, T_{CA}^*) \leq 2n - 1 \quad (52)$$

Combining Eqs. 51 and 52, we finally get for the ratio between the expected values associated with these two strategies:

$$\frac{E(G, T_R^*)}{E(G, T_{CA}^*)} \geq p^2 + \frac{p(1-p)(n-1)(1+\frac{n}{2})}{2n-1} \quad (53)$$

that is $O(n)$ for large values of n when p is constant. \square



(a) A graph G with $n = 4$, the path P

(b) Modification by `CLOSEST_ANCESTOR` for each possible subgraph

Fig. 16 An anticipatory solution P and all its possible adjustments

In other words, there exist graphs where the expected value of the optimum with respect to `ROOT` is much larger than the expectation for `CLOSEST_ANCESTOR` when applied to this solution.

7 Conclusion

In this paper we have handled the `PROBABILISTIC MIN SPANNING TREE` problem in the framework of probabilistic combinatorial optimization. We have proposed two fast modification strategies (`CLOSEST_ANCESTOR` and `ROOT`) for reconstructing a second-stage tree and we have studied particular cases of anticipatory solutions. We have shown that the problem of optimizing the expectation of the second-stage cost by selecting an appropriate first-stage (anticipatory) solution is **NP-hard** under strategy `CLOSEST_ANCESTOR` while it is polynomial under `ROOT`. Under `CLOSEST_ANCESTOR` strategy we have given several approximation results for `PROBABILISTIC METRIC MIN SPANNING TREE` and `PROBABILISTIC MIN SPANNING TREE 1,2`.

There are several open questions subject for further research. To our opinion, the major among them are the complexities of `PROBABILISTIC METRIC MIN SPANNING TREE` and `PROBABILISTIC MIN SPANNING TREE 1,2` (under `CLOSEST_ANCESTOR`; we conjecture that they are both **NP-hard**) and the improvement of their approximation ratios.

Acknowledgement The very useful and pertinent comments and suggestions of an anonymous referee are gratefully acknowledged.

References

1. Averbakh, I., Berman, O., Simchi-Levi, D.: Probabilistic a priori routing-location problems. *Naval Res. Logistics* **41**, 973–989 (1994)
2. Bertsimas, D.J.: Probabilistic Combinatorial Optimization Problems. Phd thesis, Operations Research Center, MIT, Cambridge Mass., USA (1988)
3. Bertsimas, D.J.: On probabilistic traveling salesman facility location problems. *Transp. Sci.* **3**, 184–191 (1989)
4. Bertsimas, D.J.: The probabilistic minimum spanning tree problem. *Networks* **20**, 245–275 (1990)
5. Bertsimas, D.J., Jaillet, P., Odoni, A.: A priori optimization. *Oper. Res.* **38**(6), 1019–1033 (1990)
6. Bianchi, L., Knowles, J., Bowler, N.: Local search for the probabilistic traveling salesman problem: correlation to the 2-p-opt and 1-shift algorithms. *Eur. J. Oper. Res.* **161**(1), 206–219 (2005)
7. Birge, J., Louveaux, F.: Introduction to Stochastic Programming. Springer, Berlin (1997)
8. Bourgeois, N., Della Croce, F., Escoffier, B., Murat, C., Paschos, V.Th.: Probabilistic coloring of bipartite and split graphs. *J. Comb. Optim.* **17**(3), 274–311 (2009)
9. Butelle, F.: Contribution à l’algorithmique distribuée de contrôle: arbres couvrants avec et sans contraintes. PhD thesis, Université Paris VIII (1994)
10. Chazelle, B.: A minimum spanning tree algorithm with inverse-ackerman type complexity. *J. Assoc. Comput. Mach.* **47**(6), 1028–1047 (2000)
11. Cheriton, D., Tarjan, R.E.: Finding minimum spanning trees. *SIAM J. Comput.* **5**, 724–742 (1976)
12. Cormen, T.H., Leiserson, C.E., Rivest, R.L., Stein, C.: Introduction to Algorithms, 2nd edn. McGraw-Hill (2001)
13. Dantzig, G.W.: Linear programming under uncertainty. *Manage. Sci.* **1**, 197–206 (1951)
14. Edmonds, J.: Optimum branchings. *J. Res. Natl. Bur. Stand.* **71B**, 233–240 (1967)
15. Georgiadis, L.: Arborescence optimization problems solvable by edmonds’ algorithm. *Theor. Comput. Sci.* **301**, 427–437 (2003)
16. Jaillet, P.: Probabilistic Traveling Salesman Problem. Technical Report 185, Operations Research Center, MIT, Cambridge Mass., USA (1985)

17. Jaillet, P.: A priori solution of a traveling salesman problem in which a random subset of the customers are visited. *Oper. Res.* **36**(6), 929–936 (1988)
18. Jaillet, P.: Shortest path problems with node failures. *Networks* **22**, 589–605 (1992)
19. Jaillet, P., Odoni, A.: The probabilistic vehicle routing problem. In: Golden, B.L., Assad, A.A. (eds.) *Vehicle Routing: Methods and Studies*. North Holland, Amsterdam (1988)
20. Korte, B.: *Combinatorial Optimization: Theory and Algorithms*, vol. 21 of *Algorithms and Combinatorics*, 3rd edn. Springer (2006)
21. Kouvelis, P., Yu, G.: *Robust Discrete Optimization and its Applications*. Kluwer Academic Publishers, Boston (1997)
22. Kruskal, J.B.: On the shortest spanning subtree of a graph and the traveling salesman problem. *Proc. Am. Math. Soc.* **7**(1), 48–50 (1956)
23. Murat, C., Paschos, V.Th.: The probabilistic longest path problem. *Networks* **33**, 207–219 (1999)
24. Murat, C., Paschos, V.Th.: A priori optimization for the probabilistic maximum independent set problem. *Theor. Comput. Sci.* **270**, 561–590 (2002)
25. Murat, C., Paschos, V.Th.: The probabilistic minimum vertex-covering problem. *Int. Trans. Opl. Res.* **9**(1):19–32 (2002)
26. Murat, C., Paschos, V.Th.: On the probabilistic minimum coloring and minimum k -coloring. *Discrete Appl. Math.* **154**, 564–586 (2006)
27. Paschos, V.Th., Telelis, O.A., Zissimopoulos, V.: Steiner forests on stochastic metric graphs. In: Dress, A., Xu, Y., Zhu, B. (eds.) *Proc. Conference on Combinatorial Optimization and Applications, COCOA'07*, vol. 4616 of *Lecture Notes in Computer Science*, pp. 112–123. Springer-Verlag (2007)
28. Paschos, V.Th., Telelis, O.A., Zissimopoulos, V.: Probabilistic models for the STEINER TREE problem. *Networks* **56**(1), 39–49 (2010)
29. Prekopa, A.: *Stochastic Programming*. Kluwer Academic Publishers, The Netherlands (1995)
30. Tarjan, R.E.: Efficiency of a good but not linear set-union algorithm. *J. Assoc. Comput. Mach.* **22**, 215–225 (1975)
31. Yao, A.: An $O(|E| \log \log |V|)$ algorithm for finding minimum spanning trees. *Inf. Process. Lett.* **4**, 21–23 (1975)