**URAL MATHEMATICAL JOURNAL, Vol. 11, No. 2, 2025, pp. 100–118**

DOI: 10.15826/umj.2025.2.007

# EDGE-DISJOINT SPANNING TREES OF ARBITRARY BOUNDED DIAMETER ON RANDOM INPUTS

**Edward Kh. Gimadi**[†], **Oxana Yu. Tsidulko**[††]

Sobolev Institute of Mathematics,
Siberian Branch of the Russian Academy of Sciences,
4 pr. Akad. Koptyug, Novosibirsk, 630090, Russian Federation

[†]gimadi@math.nsc.ru [††]tsidulko@math.nsc.ru

**Abstract:** We consider the following NP-hard generalization of the Minimum Spanning Tree problem. Given an undirected $n$-vertex edge-weighted complete graph and integers $d$ and $m$, find $m$ edge-disjoint spanning trees of diameter at most $d$ with minimum total weight. We propose a new polynomial-time approximation algorithm for the problem and study its performance guarantees on random inputs, that is, when the edge weights of the graph are i. i. d. random variables. We show that under mild conditions on the distribution parameters the proposed algorithm is asymptotically optimal for the case of continuous and discrete uniform distribution on $[a_n, b_n]$, $a_n > 0$, the shifted exponential distribution with shift $a_n > 0$, and distributions dominating the above. In contrast to a number of previous results for related problems, the new algorithm is asymptotically optimal not only if $d$ tends to infinity with $n$, but for constant $d$ as well.

**Keywords:** Edge-disjoint spanning trees, Bounded diameter, Random inputs, Asymptotically optimal algorithm, Probabilistic analysis.

## 1. Introduction

The Minimum Spanning Tree (MST) problem is one of the fundamental problems in discrete optimization. Given a connected undirected edge-weighted graph $G = (V, E)$, the problem is to find a spanning tree, that is, a tree with vertex set $V$, of minimum total weight. The MST problem has long been known to admit polynomial-time algorithms, for example, $O(|E| \log |V|)$-time algorithm by Boruvka [5], $O(|E| \log |V|)$-time algorithm by Kruskal [21], $O(|V|^2)$-time algorithm by Prim [24].

An important generalization of the MST problem is finding an MST of bounded diameter (BDMST). The diameter of a graph is the maximum number of edges on the shortest path between any pair of vertices of the graph. Hence, given an edge-weighted connected $n$-vertex input graph and an integer $d$, $1 < d \leq n-1$, the BDMST consists in finding a minimum-weight spanning tree of diameter at most $d$. BDMST has applications in telecommunication and wireless sensor network design, where one needs to connect a number of devices by a network at minimum cost while providing a bounded maximum communication delay [3, 19, 26, 27]; in sparse bit-vector compression [20]; in solving the joint image alignment problem for dense Active Appearance Models [1], etc. In some applications, e.g., [1, 20], an approximate solution to the so-called Given-Diameter MST problem is actually used, in which the desired tree is required to have diameter exactly equal to the given value $d$. The BDMST is known to be polynomially solvable for $d = 2, 3, n-1$, and NP-hard otherwise [7]. In general graphs it is also NP-hard to approximate the BDMST to within an approximation factor better than $O(\log n)$ for any fixed $d$, $4 \leq d \leq n-2$ [4].

We now consider a further generalization of the BDMST problem: given an edge-weighted connected graph $G = (V, E)$, find $m$ edge-disjoint spanning trees of diameter upper bounded by $d$ and minimum total weight. We refer to this problem as $m$-$d$-MST. For the case when there is no restriction on the diameter, the $m$-MST problem can be solved in $O(|E| \log |E| + m^2|V|^2)$

time [25]. If $d = 2$, each requested spanning tree should be a star-tree with one central vertex of degree $n-1$ and $n-1$ vertices of degree 1. Thus, for $m \geq 2$ and $d = 2$, the $m$-$d$-MST does not have a feasible solution, since once the edges incident to a central vertex are used in one spanning tree, there are no remaining edges to span it in other trees. If $d = 3$ and $m \geq 2$, for the $m$-$d$-MST to have a feasible solution, each spanning tree should have diameter exactly 3, that is, each such spanning tree $T_i$ should have a central edge $e_i$, and every other vertex must be adjacent to one of $e_i$'s endpoints. Thus, at least for constant $m$, one can in polynomial time try all possible subsets of $m$ vertex-disjoint central edges and complete them into a feasible solution by adding the corresponding shortest edges from the remaining vertices. In general, however, $m$-$d$-MST is NP-hard [22].

Instead of the worst-case scenario, in this paper we consider the problem on random inputs. That is, we assume that the input $n$-vertex graph is complete and the weights of its edges are i. i. d. random variables that follow a certain distribution $\mathcal{D}$. In this setting we are interested in polynomial-time asymptotically optimal algorithms, which, with probability at least $1-\delta$, provide $(1+\varepsilon)$-approximate solutions such that $\delta$ and $\varepsilon$ tend to zero as $n$ grows.

In this area, one can distinguish two groups of papers in relation to the considered distribution $\mathcal{D}$ depending on whether its support is shifted away from zero [10–15] or not shifted away from zero [2, 6]. In the latter case, it is more difficult to construct a tight lower bound on the weight of the optimal solution, so the algorithms and their analysis are more sophisticated. For example, in [6] it was shown that the MST on random inputs with i. i. d. edge weights drawn from the uniform distribution $\mathrm{U}(0,1)$ has weight, with high probability, tending to $\zeta(3) = \sum_{i=1}^{\infty} 1/i^3$ as $n$ tends to infinity. Paper [2] shows that the optimal 1-$d$-MST (or simply BDMST) on random inputs with the exponential distribution $\mathrm{Exp}(1)$ of mean 1 also has weight tending to $\zeta(3)$, if $d > \log_2 \log_2 n + \omega(1)$, where $\omega(1)$ is any function tending to infinity with $n$; and if $d < \log_2 \log_2 n - \omega(1)$, then the weight is doubly-exponentially large in $\log_2 \log_2 n - d$. In [2], two greedy asymptotically optimal algorithms are also proposed for the BDMST on $\mathrm{Exp}(1)$ random inputs, one for each of the two cases of $d$ mentioned above. Yet, it is difficult to generalize these algorithms and their analysis to the case of $m$-$d$-MST.

For distributions $\mathcal{D}$ with support $[a,b]$ or $[a,\infty)$, where $a > 0$ (e.g., the uniform distribution $\mathrm{U}(a,b)$, a shifted exponential distribution $\mathrm{Exp}(a,\lambda)$, or a truncated normal distribution $\mathcal{N}(a,\sigma)$), it turns out that in many cases the obvious lower bound $a(n-1)m$ on the weight of $m$-$d$-MST is asymptotically tight, in the sense that it tends to the weight of the optimal solution as $n$ grows. Using techniques developed in [10–14], asymptotically optimal algorithms have been constructed for several variants of $m$-$d$-MST. In particular, [11] considered the 1-$d$-MST, papers [9, 12, 14] considered the Given-Diameter $m$-$d$-MST, where the diameter should be exactly equal to $d$, and papers [10, 13] considered the variant of $m$-$d$-MST, where the diameter is required to be at least $d$. These papers proposed asymptotically optimal algorithms for the corresponding problems in case $\mathcal{D}$ is $\mathrm{U}(a,b)$, $\mathrm{Exp}(a,\lambda)$, $\mathcal{N}(a,\sigma)$, and certain other distributions, if $d = \omega(1)$ tends to infinity with $n$. In a resent paper [15], a new approach for the 1-$d$-MST was proposed that constructs asymptotically optimal solutions on uniform random inputs for any $d \geq 4$. In this paper we show how to extend that result to the case of $m$-$d$-MST and to a wider class of random inputs.

The rest of the paper is organized as follows. Section 2 defines the problem and the considered classes of random inputs; Section 3 describes the approximation algorithm. In Section 4.1, we present a general probabilistic analysis for the constructed algorithm. Section 4.2 provides technical lemmas estimating expectations of certain random variables. Finally, in Section 4.3, we prove conditions under which the constructed algorithm is asymptotically optimal on the considered classes of random inputs.

## 2. Problem statement

In the $m$-$d$-MST problem, given a complete undirected $n$-vertex graph $G = (V, E)$, edge weights $w : E \to \mathbb{R}_{\geq 0}$, an integer $d \in \{4, \dots, n-1\}$, and an integer $m \leq n/2$, the goal is to find $m$ edge-disjoint spanning trees $T_1, \dots, T_m$, each of diameter at most $d$, with minimum total weight:

$$\sum_{i=1}^{m} w(T_i) = \sum_{i=1}^{m} \sum_{e \in T_i} w(e) \to \min .$$

We consider the $m$-$d$-MST on random inputs from the following three classes: $\mathrm{U}(a_n, b_n)$, d-$\mathrm{U}(a_n, b_n)$, and $\mathrm{Exp}(a_n, \lambda_n)$, in which the edge weights of the $n$-vertex input graph are i. i. d. random variables $\rho$ with

1) *continuous uniform distribution* on the interval $[a_n, b_n]$, $0 < a_n < b_n$, with CDF

$$F_{\mathrm{U}}(x) = \begin{cases} 0 & \text{if } x \leq a_n, \\ \dfrac{x - a_n}{b_n - a_n} & \text{if } x \in [a_n, b_n], \\ 1 & \text{if } \ x \geq b_n, \end{cases} \tag{2.1}$$

2) *discrete uniform distribution d-U$(a_n, b_n)$* on the positive integer segment $[a_n, b_n]_{\mathbb{N}} := \mathbb{N} \cap [a_n, b_n]$, $a_n, b_n \in \mathbb{N}$, $0 < a_n < b_n$, where the probability of hitting each number in the segment is the same, that is,

$$\mathbf{P}\{\rho = x\} = \begin{cases} \dfrac{1}{b_n - a_n + 1} & \text{if } x \in [a_n, b_n]_{\mathbb{N}}, \\ 0, & \text{otherwise}, \end{cases} \qquad F_{\text{d-U}}(x) = \begin{cases} 0 & \text{if} \quad x \leq a_n, \\ \dfrac{\lfloor x \rfloor - a_n + 1}{b_n - a_n + 1} & \text{if} \quad x \in [a_n, b_n], \\ 1 & \text{if} \quad x \geq b_n, \end{cases}$$

3) shifted exponential distribution $\mathrm{Exp}(a_n, \lambda_n)$ on the interval $[a_n, \infty)$, $a_n > 0$, with CDF

$$F_{\mathrm{Exp}}(x) = \begin{cases} 1 - \exp\Big( - \dfrac{x - a_n}{\lambda_n} \Big) & \text{if} \quad x \geq a_n, \\ 0, & \text{if} \quad x < a_n. \end{cases} \tag{2.2}$$

We will also consider random inputs with distributions that dominate one of the three distributions listed above.

**Definition 1.** *A distribution with CDF $F_1(x)$ is said to* dominate *a distribution with CDF $F_2(x)$ if for all $x \in (-\infty, \infty)$: $F_1(x) \geq F_2(x)$.*

For example, the truncated normal distribution $\mathcal{N}(a_n, \sigma_n^2)$ with PDF

$$f_{\mathcal{N}}(x) = \begin{cases} \dfrac{1}{\sqrt{2\pi\sigma_n^2}} \exp\Big( - \dfrac{(x - a_n)^2}{2\sigma_n^2} \Big) & \text{if} \quad x > a_n, \\ 0, & \text{otherwise}, \end{cases}$$

dominates the shifted exponential distribution $\mathrm{Exp}(a_n, \lambda_n)$ with $\lambda_n = 2\sigma_n$, see Proposition 3 in [13].

## 3. Approximation algorithm

**The problem with the previous algorithms.** Papers [9, 12, 14] proposed approximation algorithms mainly for the Given-Diameter $m$-$d$-MST, which can be easily transformed into algorithms for the $m$-$d$-MST with a diameter bounded from above. The output of those approximation algorithms was a collection of spanning caterpillar trees $T_1, \dots, T_m$ constructed as follows.

In Step 1, the vertices of the input graph were randomly partitioned into $m+1$ parts: $V_1, \dots, V_m$ and $V' = V \setminus \bigcup_{j=1}^{m} V_j$ such that $|V_1| = \dots = |V_m| = d+1$. For each $i = 1, \dots, m$, the base tree $S_i$, which was a simple path of $d$ edges on vertices of $V_i$, was built by greedily applying the principle "go to the closest non-visited vertex" of $V_i$. In Step 2, for each tree $T_i := S_i$ the shortest edge $\{v, u\}$ from each vertex $v \in \bigcup_{j=1}^{m} V_j \setminus V_i$ to an inner vertex $u$ of a certain half of $S_i$ was added to $T_i$. Finally, in Step 3, each tree $T_i$ was augmented by adding for each vertex $v \in V'$ an edge $\{v, u\}$ to the closest inner vertex $u$ of $S_i$.

The analysis in papers [9, 12, 14] shows that, with high probability, the relative error of these algorithms satisfies

$$\varepsilon = O\left(\frac{\mathbf{E}(W) - m(n-1)a_n}{m(n-1)a_n}\right),$$

where $m(n-1)a_n$ is the obvious lower bound[1] for the weight of the optimal solution, and $\mathbf{E}(W)$ is the expectation of the weight of the constructed solution, $W = \sum_{i=1}^{m} w(T_i)$. Further analysis in those papers yields $\varepsilon = O\left(\beta_n/a_n \cdot 1/d\right)$, where $a_n, \beta_n > 0$ are the parameters of the edge-weight distribution: uniform $U(a_n, \beta_n)$, shifted exponential $\mathrm{Exp}(a_n, \beta_n)$, truncated normal $N(a_n, \beta_n)$, etc. Such $\varepsilon$ tends to zero if $d = \omega(1)$ tends to infinity with $n$, whereas requiring $\beta_n/a_n = o(1)$ is unrealistic. For constant $d$ this estimate does not yield asymptotic optimality. Note that the weight of *each edge* added to the solution in each of the three steps of these algorithms equals the minimum of $\Theta(d)$ i. i. d. random values, and in the case of constant $d$, the expectation of such a minimum is, roughly speaking, "not small enough" compared to the lower bound $a_n$ to yield an overall asymptotically optimal solution. To achieve asymptotic optimality, the solution is allowed to contain only $o(mn)$ of such "not small enough" edges, while in the case of constant $d$ the algorithms from [9, 12, 14] construct solutions in which all the edges are "not small enough".

**The new algorithm.** In this paper, we mitigate the above problem and, elaborating on the idea introduced in [15] for a single tree, propose Algorithm 1 that constructs $m$ trees $T_1, \dots, T_m$ starting from more bushy vertex-disjoint base trees $S_1, \dots, S_m$, each having $\Theta(\sqrt{n})$ instead of $\Theta(d)$ vertices. Most edges of $S_1, \dots, S_m$ will be chosen as the minimum-weight edge out of $\omega(1)$ edges. For each $i = 1, \dots, m$, $T_i := S_i$ will then be populated with an edge $\{v, u\}$ from each $v \in V \setminus V(S_i)$ to the closest vertex $u$ in a certain quarter (or half) of the inner vertices of $S_i$. And since there are $\Theta(\sqrt{n})$ inner vertices in $S_i$, each such edge $\{v, u\}$ will be chosen as the minimum-weight edge out of $\omega(1)$ edges. Therefore, most of the edges added to the solution will be "small enough", which in Section 4 will allow us to show asymptotic optimality of the algorithm for the considered classes of random inputs, regardless of the value of $d \in \{4, \dots, n-2\}$.

[1] Although the estimation for $\varepsilon$ uses the weight of the lower bound instead of the weight of the optimal solution, from the results of the current paper (Theorem 1) we know that there exists a feasible solution whose weight tends to the lower bound for the considered classes of random inputs, and, thus, so does the optimum.

**Algorithm 1:** Approximation algorithm for the $m$-$d$-MST

**Input :** complete undirected graph $G = (V, E)$, $|V| = n$, weights of edges $w : E \to \mathbb{R}_{\geq 0}$, and positive integers $m$ and $d$ : $4 \leq d \leq n - 1$.

**Result:** $m$ edge-disjoint spanning trees $T_1, \ldots, T_m$ each of diameter at most $d$.

**Step 0.** Randomly and uniformly split the vertex set $V$ into parts $V_1$ and $V_2$ so that $V_1 \cup V_2 = V$, $V_1 \cap V_2 = \emptyset$, $|V_1| = \lfloor n/2 \rfloor$, and $|V_2| = \lceil n/2 \rceil$;

Set the edge sets $T_i := \emptyset$, $S_i := \emptyset$ for all $i = 1, \ldots, m$, and the set of used edges $T := \emptyset$;

Set $D := \lfloor \min\{d, \sqrt{n}\}/2 \rfloor$ and $\ell := \lceil \sqrt{n}/(2D) \rceil$;

**Step 1.** /* Construct vertex-disjoint base trees $S_1, \ldots, S_m$ of diameter $2D$ */

**foreach** $i = 1, \ldots, m$ **do**

  Arbitrarily select: vertex $v_0^i \in V_{1+(i \bmod 2)} \setminus V(T)$ as the central vertex of $S_i$, vertices $v_{1,1,1}^i, \ldots, v_{1,\ell,1}^i \in V_1 \setminus (V(T) \cup \{v_0^i\})$, and $v_{2,1,1}^i, \ldots, v_{2,\ell,1}^i \in V_2 \setminus (V(T) \cup \{v_0^i\})$ ;

  $S_i := \bigcup_{p=1}^{\ell} \bigcup_{r=1}^{2} \{\{v_0^i, v_{r,p,1}^i\}\}$ and $T := T \cup S_i$ ;

  **foreach** $r = 1, 2$ **do**

    **foreach** $p = 1, \ldots, \ell$ *and* $k = 1, \ldots, D - 1$ **do**

      $v_{r,p,k+1}^i := \arg\min\{w(\{v_{r,p,k}^i, u\}) \mid u \in V_r \setminus V(T)\}$;

      $S_i := S_i \cup \{\{v_{r,p,k}^i, v_{r,p,k+1}^i\}\}$ and $T := T \cup S_i$;

  $T_i := S_i$ and $U_i := \{v_{r,p,k}^i \mid r = 1, 2,\ p = 1, \ldots, \ell,\ k = 1, \ldots, D - 1\}$ ;

**Step 2.** /* Populate each tree $T_i$ with edges spanning vertices of all $S_j$, $i \neq j$, keeping all trees edge-disjoint and of diameter $2D$ */

**foreach** $i = 1, \ldots, m$ **do**

  For each vertex $v \in V(S_i) \setminus \{v_0^i\}$ where $v = v_{r,p,k}^i$, set $\mathrm{num}(v) := p + (k - 1)\ell$;

**foreach** $i = 1, \ldots, m$ **do**

  **foreach** $j = 1, \ldots, m$, $j \neq i$ **do**

    **foreach** $r = 1, 2$ **do**

      **foreach** $v \in V(S_j) \cap V_r$ **do**

        **if** $v = v_0^j$ **then**

          $x^* := \arg\min\{w(\{v_0^j, x\}) \mid x \in U_i \cap V_{3-r}\}$ ;

        **else**

          $x^* := \arg\min\{w(\{x, v\}) \mid x \in U_i \cap V_{3-r}$, and $\mathrm{num}(x) \equiv \mathrm{num}(v) + [i < j] \pmod 2\}$;

        $T_i := T_i \cup \{\{x^*, v\}\}$;

**Step 3.** /* Populate each tree $T_i$ with edges spanning vertices of $V \setminus V(T_i)$ */

**foreach** $i = 1, \ldots, m$ **do**

  **foreach** $r = 1, 2$ **do**

    **foreach** $v \in V_r \setminus V(T_i)$ **do**

      $x^* := \arg\min\{w(\{x, v\}) \mid x \in U_i \cap V_{3-r}\}$ and $T_i := T_i \cup \{\{x^*, v\}\}$;

**Return:** $T_1, \ldots, T_m$.

Algorithm 1 consists of Steps 0–3. In Step 0, the vertices of the initial graph are uniformly and randomly partitioned into $V_1$ and $V_2$, $|V_1| = \lfloor n/2 \rfloor$ and $|V_2| = \lceil n/2 \rceil$. This is a technical step that will force the weights of edges added in Step 1 and Steps 2–3 in the constructed trees to be mutually independent random variables, as follows. For $r = 1, 2$, let

$$E_{rr} = \big\{\{u, v\} \in E \mid u, v \in V_r\big\}$$

and

$$E_{12} = \big\{\{u, v\} \in E \mid u \in V_1, v \in V_2\big\}.$$

The edges added to the trees in Step 1 (except for those chosen randomly in lines 6–7) are selected from $E_{11} \cup E_{22}$, while the edges added in Steps 2–3 are from $E_{12}$; by construction, the weights of edges from $E_{11} \cup E_{22}$ are independent of the weights of edges in $E_{12}$.

In Step 1, $m$ vertex-disjoint base trees $S_i$, $i = 1, \dots, m$, are constructed. Each $S_i$ consists of a central vertex $v_0^i \in V \setminus \bigcup_{j=1}^{i-1} V(S_j)$ and $2\ell$ simple $D$-edge paths originating from $v_0^i$: $\ell$ paths on the vertices of $V_1$ and $\ell$ paths on the vertices of $V_2$; $D = \lfloor \min\{d, \sqrt{n}\}/2 \rfloor$ and $\ell = \lceil \sqrt{n}/(2D) \rceil$. Note that vertex $v_0^i$ is chosen from $V_2$ if $i \equiv 0 (\mathrm{mod}\ 2)$ and from $V_1$, otherwise (line 6). The first edges of the $2\ell$ paths incident to $v_0^i$ are chosen at random in lines 6–7, then each path is greedily constructed by applying the principle "go to the closest non-visited vertex" in lines 9–11.

In Step 2, for each $i = 1, \dots, m$, tree $T_i := S_i$ is extended so as to span all the vertices of $\bigcup_{i=1}^{m} V(S_i)$. To that end, for each $j \neq i$, $1 \leq j \leq m$, from each vertex of $S_j$, an edge to a certain closest vertex of $U_i$ is added to $T_i$ in lines 17–24, where $U_i$ is the set of vertices of $S_i$ excluding the leaves and the center (line 12). All the edges added in this step are restricted to belong to $E_{12}$. Moreover, in order to keep the resulting trees edge-disjoint, the vertices of each $S_i$, $i = 1, \dots, m$, are enumerated in a breadth-first order (line 15), and the edges between $V(S_j)$ and $V(S_i)$ added to $T_i$ are chosen only among the edges with endpoints of the same parity if $i > j$, and with the opposite parity, otherwise (lines 23–24). Line 23 uses Iverson's notation $[i < j]$, where the bracket value is 1 if the condition holds, otherwise it is 0. This parity trick also forces the weights of the edges added in Step 2 to be independent random variables, since each such edge $e$ is selected as the shortest edge from a unique subset of edges $E_e$, where $E_e \cap E_{e'} = \emptyset$ for $e \neq e'$.

After Step 2, $V(T_1) = \dots = V(T_m)$. In Step 3, for each $i = 1, \dots, m$, tree $T_i$ is populated with an edge $\{v, u\} \in E_{12}$ from each vertex $v \in V \setminus V(T_i)$ to the closest vertex $u$ in $U_i$, where $U_i$ is the set of vertices of $S_i$ excluding the leaves and the center (line 12).

The following two lemmas show the correctness and the time complexity of Algorithm 1, while the quality of its solution will be studied in Section 4.

**Lemma 1.** *Algorithm 1 constructs a feasible solution for the m-d-MST, if* $m \leq n/(2D\ell + 1)$, $d \geq 4$ *and* $n > 16$.

P r o o f. First, we show that each constructed tree is a spanning tree of diameter at most $d$. In Step 1 of Algorithm 1, we construct the base $S_i$ of each $T_i$, $i = 1, \dots, m$, such that $S_i$ is a spider tree with central vertex $v_0^i$ from which $2\ell$ simple vertex-disjoint paths originate. Each such path consists of $D$ edges: one edge is added at line 7, and $D - 1$ are added at lines 9–11. Thus, for each $i = 1, \dots, m$, the diameter of $T_i = S_i$ after Step 1 is

$$\mathrm{diam}(T_i) = \mathrm{diam}(S_i) = 2D = 2\lfloor \min\{d, \sqrt{n}\}/2 \rfloor \leq d.$$

In Steps 2 and 3, for $i = 1, \dots, m$ each $T_i$ is extended so as to span the remaining vertices: for each $v \in V \setminus V(T_i)$ a certain edge $\{u, v\}$ is added to $T_i$, such that $u \in U_i$ is a vertex of $S_i$ apart from the leaves and the center. Let $\mathrm{dist}(u, v)$ be the number of edges in the shortest path between vertices

$u$ and $v$ in $T_i$. Then for each $u \in U_i$, $\mathrm{dist}(u, v_i^0) \le D-1$, and for an edge $\{u, v\}$ added in Step 2 or 3 and an arbitrary vertex $v' \in V(S_i)$ it holds that

$$\mathrm{dist}(v, v') = 1 + \mathrm{dist}(u, v') \le 1 + \mathrm{dist}(u, v_i^0) + \mathrm{dist}(v', v_i^0) \le 2D \le d,$$

while for another edge $\{u_2, v_2\}$ added in Step 2 or 3 with $u_2 \in U_i$, we have

$$\mathrm{dist}(v, v_2) = 2 + \mathrm{dist}(u, u_2) \le 2 + \mathrm{dist}(u, v_i^0) + \mathrm{dist}(u_2, v_i^0) \le 2D \le d.$$

Thus, after Steps 2 and 3, $T_i$ is a spanning tree and $\mathrm{diam}(T_i) = 2D \le d, \quad i = 1, \ldots, m$.

Now we show that all the constructed trees are edge-disjoint. In Step 1 we maintain and update the set $T$ of all currently used edges in a timely manner, so that when a new edge is added to one of the trees, it never belongs to $T$, and when the center vertex of a new tree is selected, it never belongs to $V(T)$. Therefore, $S_1, \ldots, S_m$ are vertex-disjoint and, thus, edge-disjoint. Note that since $m \le n/(2D\ell+1)$, there are enough vertices to construct all $m$ base trees, and Step 1 is well defined.

For Step 2, consider an edge $e = \{v_i, v_j\}$ with $v_i \in V(S_i)$ and $v_j \in V(S_j)$ added to the solution for some $i < j$, $1 \le i, j \le m$. If $v_j = v_0^j$, then $v_i \in U_i, v_i \ne v_0^i$ and $e \in T_i$ due to line 21; similarly, if $v_i = v_0^i$, then $v_j \ne v_0^j$ and $e \in T_j$. Otherwise, due to line 23, edge $e$ can belong to exactly one of the trees $T_i$ or $T_j$ in two non-overlapping cases: $e \in T_j$ if $\mathrm{num}(v_j) \equiv \mathrm{num}(v_i) \pmod 2$ and $e \in T_i$ if $\mathrm{num}(v_i) \equiv \mathrm{num}(v_j) + 1 \pmod 2$.

Note that for Step 2 to be well defined, for each $U_i \cap V_r$ with $i = 1, \ldots, m$, $r = 1, 2$, there has to be at least one vertex in $U_i$ with an even and one with an odd number, thus, $(D-1)\ell \ge 2$. Therefore, if $m > 1$ and $D = 2$, it should be that $\ell = \lceil \sqrt{n}/4 \rceil \ge 2$ and, thus, $n > 16$.

Finally, consider an edge $e_i = \{v, x_i\}$ added to $T_i$ in Step 3 with $v \in V \setminus \bigcup_{i=1}^m V(T_i)$ and $x_i$ being a non-leaf vertex of $S_i$, $i \in \{1, \ldots, m\}$. Since $S_1, \ldots, S_m$ are vertex-disjoint, $e_i \notin T_j$ for $j \ne i$. □

**Lemma 2.** *Algorithm 1 runs in $O(n^2)$ time.*

P r o o f. Recall that $(1 + 2D\ell)m \le n$, $D = \lfloor \min\{d, \sqrt{n}\}/2 \rfloor$, $\ell = \lceil \sqrt{n}/(2D) \rceil$. Step 0 of Algorithm 1 can be done in $O(n)$ time. In Step 1, to construct each $S_i$, $i = 1, \ldots, m$, selecting the first $2\ell$ edges at random can be done in $O(n)$ time, and then adding each of the remaining $2(D-1)\ell$ edges takes $O(n)$ time. Thus, Step 1 takes $O(2D\ell mn) = O(n^2)$ time. In Step 2, $m(m-1)(D\ell+1)$ edges are added, and adding each edge takes $O(D\ell)$ time. Thus, Step 2 takes $O(m^2\ell^2 D^2) = O(n^2)$ time. In Step 3, adding each edge takes $O(D\ell)$ time, which results in an overall $O(mD\ell(n - mD\ell - m)) = O(mD\ell n) = O(n^2)$ time. Therefore, the total time complexity of Algorithm 1 is $O(n^2)$. □

## 4. Probabilistic analysis of Algorithm 1

To evaluate the accuracy of the algorithm, we perform a probabilistic analysis, where instead of describing the performance of the algorithm for the worst-case input scenario, we focus on the performance of the algorithm on "typical" instances. That is, we assume the instances of the problem belong to a certain probability space. In this paper, we consider three classes of inputs: $\mathrm{U}(a_n, b_n)$, d-$\mathrm{U}(a_n, b_n)$, and $\mathrm{Exp}(a_n, \lambda_n)$, in which the edge weights of the $n$-vertex input graph are i. i. d. random variables $\rho$ with the continuous uniform, discrete uniform, and shifted exponential distribution, see (2.1)–(2.2).

**Definition 2.** *Let $W_A(I_n)$ and $OPT(I_n)$ be, respectively, the approximate value obtained by approximation algorithm $A$ and the optimum value of the objective function of some minimization problem on an $n$-vertex graph instance $I_n$. Algorithm $A$ is said to have* ***performance guarantees*** *$(\varepsilon_n, \delta_n)$ on a class of random instances, if for all instances $I_n$ from this class*

$$\mathbf{P}\Big\{W_A(I_n) > \big(1+\varepsilon_n\big)OPT(I_n)\Big\} \le \delta_n,$$

*where $\varepsilon_n$ is* the relative error *of algorithm $A$, $\delta_n$ is* ***the failure probability*** *of algorithm $A$, defined as the proportion of cases in which the algorithm does not achieve the relative error $\varepsilon_n$ or fails to produce any answer at all.*

**Definition 3.** *An approximation algorithm $A$ is called* ***asymptotically optimal*** *on a class of instances if there exist performance guarantees $(\varepsilon_n, \delta_n)$ such that for all instances $I_n$ of this class, $\varepsilon_n \to 0$ and $\delta_n \to 0$ as $n \to \infty$.*

The goal of this section is to show conditions under which Algorithm 1 from Section 3 is asymptotically optimal on the classes of U$(a_n, b_n)$, d-U$(a_n, b_n)$, and Exp$(a_n, \lambda_n)$ random inputs.

### 4.1. General analysis of Algorithm 1

In this section, we describe the weight of the solution constructed by Algorithm 1 in terms of random variables, and in Lemma 3 we derive general upper bounds for the performance guarantees $\varepsilon_n$ and $\delta_n$ of Algorithm 1. The only property of the edge-weights distribution $\mathcal{D}$ used here is that its support is shifted away from zero: $[a_n, b_n]$ or $[a_n, \infty)$ with $a_n > 0$. More explicit forms for the bounds obtained in Lemma 3 will be derived for specific distributions in Section 4.3.

Let

$$W = \sum_{i=1}^{m} \sum_{e \in T_i} w(e)$$

be the total weight of the spanning trees $T_1, \ldots, T_m$ returned by Algorithm 1; let $W_i$ be the total weight of edges added to the trees in Step $i$ of Algorithm 1, $i \in \{1,2,3\}$, so that $W = W_1+W_2+W_3$.

Let $\rho$ be a random variable with initial distribution $\mathcal{D}$, and let $\rho_k$ be a random variable equal to the minimum of $k$ i. i. d. random variables of type $\rho$. By $\{\rho_k^j\}_{j\in J}$ we will denote a set of $|J|$ independent random variables of type $\rho_k$ with the same value of $k$, where the superscript $j$ is a (composite) index that allows us to distinguish between different independent random variables of type $\rho_k$. Note that a random variable of type $\rho_1$ has the same distribution as a random variable of type $\rho$. For the weights $W_1$, $W_2$, and $W_3$, we can show the following.

The total weight of edges added in Step 1 satisfies

$$\begin{aligned} W_1 = \sum_{i=1}^{m}\sum_{j=1}^{2\ell} \rho_1^{1,i,j} + \sum_{i=1}^{m}\sum_{p=1}^{\ell}\sum_{k=1}^{D-1} \rho^{1,1,i,p,k}_{\lfloor n/2\rfloor - D\ell(i-1) - \lfloor i/2\rfloor - (p-1)D - k} \\ + \sum_{i=1}^{m}\sum_{p=1}^{\ell}\sum_{k=1}^{D-1} \rho^{1,2,i,p,k}_{\lceil n/2\rceil - D\ell(i-1) - \lceil i/2\rceil - (p-1)D - k}. \end{aligned} \tag{4.3}$$

Here, the first sum corresponds to the weights of $2m\ell$ edges incident to $v_0^1, \ldots, v_0^m$ and chosen in line 6 of Algorithm 1. Since these edges were chosen uniformly at random, their weights have the initial distribution $\mathcal{D}$ and are of type $\rho_1$. The second and the third sums correspond to the total weight of all edges added to the solution in lines 9–11 with both endpoints in $V_1$ and with both endpoints in $V_2$, respectively. For example, for the second sum, note that when adding a new edge of minimum weight from a vertex $v^i_{1,p,k} \in V(S_i) \cap V_1$ to the vertex $v^i_{1,p,k+1} \in V_1 \setminus V(T)$ in line 9, $v^i_{1,p,k+1}$ is chosen out of $\lfloor n/2 \rfloor - D(i-1)\ell - \lfloor i/2 \rfloor - (p-1)D - k$ vertices, since $|V_1| = \lfloor n/2 \rfloor$ and by the time the algorithm chooses $v^i_{1,p,k+1}$: $\lfloor i/2 \rfloor$ centers (due to line 6) of previously constructed and possibly current base trees are already in $V_1 \cap V(T)$, $D\ell(i-1)$ non-center vertices of previously constructed base trees are already in $V_1 \cap V(T)$, and $(p-1)D+k$ non-center vertices of the current $S_i$ are in $V_1 \cap V(T)$. Similar arguments apply for the third sum. Finally, note that the set of edges from which edge $\{v^i_{r,p,k}, v^i_{r,p,k+1}\}$ is chosen in line 9 does not intersect with the edge sets from which edges on previous iterations were chosen. Thus, all random variables summed in $W_1$ are mutually independent.

The total weight of edges added in Step 2 satisfies

$$\begin{aligned} W_2 = &\sum_{i=1}^{m} \sum_{j=1, j\neq i}^{m} \rho^{2,i,j}_{\ell(D-1)} + \sum_{i=1}^{m} \sum_{j=1, j\neq i}^{m} \sum_{r=1}^{2} \sum_{k=1, k\%2=1}^{D\ell} \Big( \rho^{2,i,j,r,k}_{\lceil (D-1)\ell/2 \rceil} \cdot [i<j] + \rho^{2,i,j,r,k}_{\lceil (D-1)\ell/2 \rceil} \cdot [i>j] \Big) \\ &+ \sum_{i=1}^{m} \sum_{j=1, j\neq i}^{m} \sum_{r=1}^{2} \sum_{k=1, k\%2=0}^{D\ell} \Big( \rho^{2,i,j,r,k}_{\lfloor (D-1)\ell/2 \rfloor} \cdot [i<j] + \rho^{2,i,j,r,k}_{\lfloor (D-1)\ell/2 \rfloor} \cdot [i>j] \Big) \\ &= \sum_{i=1}^{m} \sum_{j=1, j\neq i}^{m} \rho^{2,i,j}_{\ell(D-1)} + \sum_{i=1}^{m} \sum_{j=1, j\neq i}^{m} \sum_{r=1}^{2} \sum_{k=1, k\%2=1}^{D\ell} \rho^{2,i,j,r,k}_{\lceil (D-1)\ell/2 \rceil} \\ &+ \sum_{i=1}^{m} \sum_{j=1, j\neq i}^{m} \sum_{r=1}^{2} \sum_{k=1, k\%2=0}^{D\ell} \rho^{2,i,j,r,k}_{\lfloor (D-1)\ell/2 \rfloor}. \end{aligned} \tag{4.4}$$

Note that for $i = 1, \dots, m$, $r = 1, 2$, the set $U_i \cap V_{3-r}$ consists of $(D-1)\ell$ vertices, out of which $\lceil (D-1)\ell/2 \rceil$ vertices have an odd number and $\lfloor (D-1)\ell/2 \rfloor$ have an even number. Variables $\rho^{2,i,j}_{\ell(D-1)}$ in the first sum of (4.4) stand for the weights of edges added in line 21, where an edge from $v^j_0$ to the closest out of $(D-1)\ell$ vertices of $U_i$ from the opposite part (whether it is $V_1$ or $V_2$) is added to the solution. In the second and the third sums, the term $\rho^{2,i,j,r,k}$ stands for the weight of an edge selected in lines 23–24 from the vertex $v \in S_j \cap V_r$ with $\mathrm{num}(v) = k$ to the closest inner vertex of $U_i \cap V_{3-r}$ of the corresponding parity. Note that for each two edges $e$ and $e'$ added to the solution in Step 2, not only the edges $e$ and $e'$ themselves, but also the sets of edges out of which $e$ and $e'$ were chosen as the ones with minimum weight do not intersect. Thus, all the random variables summed in $W_2$ are mutually independent. Moreover, since in Step 2 only edges between $V_1$ and $V_2$ were considered, their weights are independent of the weights of edges considered in Step 1 with both endpoints either in $V_1$ or $V_2$.

The total weight of edges added in Step 3 satisfies

$$W_3 = \sum_{i=1}^{m} \sum_{j=1}^{\lfloor n/2 \rfloor - mD\ell - \lfloor m/2 \rfloor} \rho^{3,1,i,j}_{(D-1)\ell} + \sum_{i=1}^{m} \sum_{j=1}^{\lceil n/2 \rceil - mD\ell - \lceil m/2 \rceil} \rho^{3,2,i,j}_{(D-1)\ell}. \tag{4.5}$$

In Step 3 for each $i = 1, \dots, m$, $T_i$ is populated with edges spanning the remaining set of vertices $V' = V \setminus V(T_i)$. Note that

$$|V'| = n - m(1 + 2D\ell), \quad |V' \cap V_1| = \lfloor n/2 \rfloor - mD\ell - \lfloor m/2 \rfloor,$$
$$|V' \cap V_2| = \lceil n/2 \rceil - mD\ell - \lceil m/2 \rceil.$$

Each term $\rho^{3,1,i,j}_{(D-1)\ell}$ in the first sum of (4.5) corresponds to the weight of an edge added to $T_i$, which was chosen in line 29 as the edge of least weight connecting $v_j \in V' \cap V_1$ with one of $(D-1)\ell$ vertices of $U_i \cap V_2$. Similarly, each term $\rho^{3,2,i,j}_{(D-1)\ell}$ in the second sum of (4.5) stands for the weight of an edge added to $T_i$, which was chosen in line 29 as the edge of least weight connecting $v_j \in V' \cap V_2$ with one of $(D-1)\ell$ vertices of $U_i \cap V_1$. Note again that for each edge $e$ added to the solution in Step 3, the set of edges from which $e$ was chosen as the one with minimum weight does not intersect with the sets of edges from which any other edge $e' \in \bigcup_{i=1}^{m} T_i$ was chosen. Thus, all random variables summed in $W_3$ are mutually independent, and are independent of the random variables summed in $W_1$ and $W_2$.

In what follows, it will be more convenient for us to use the normalized random variables:

$$\xi = \frac{\rho - a_n}{\beta_n} \quad \text{and} \quad \xi_k = \frac{\rho_k - a_n}{\beta_n}, \tag{4.6}$$

where

$$\beta_n = \begin{cases} b_n - a_n & \text{for inputs in } \mathrm{U}(a_n, b_n), \\ b_n - a_n + 1 & \text{for inputs in d-}\mathrm{U}(a_n, b_n), \\ \lambda_n & \text{for inputs in } \mathrm{Exp}(a_n, \lambda_n). \end{cases}$$

We will use the notations $W'$, $W_1'$, $W_2'$, and $W_3'$ for the corresponding sums of normalized random variables.

**Lemma 3.** *Algorithm 1 for the m-d-MST on an n-vertex complete graph with weights of edges from* $\mathrm{U}(a_n, b_n)$*, discrete* $\mathrm{U}(a_n, b_n)$ *or* $\mathrm{Exp}(a_n, \lambda_n)$ *has the following estimates for the relative error* $\varepsilon_n$ *and the failure probability* $\delta_n$*:*

$$\varepsilon_n = \frac{2\beta_n}{m(n-1)a_n}\mathcal{E}, \tag{4.7}$$

$$\delta_n = \mathbf{P}\Big\{\widetilde{W}' \geq \mathcal{E}\Big\}, \tag{4.8}$$

*where* $\mathcal{E}$ *is an upper bound on* $\mathbf{E}W'$ *and* $\widetilde{W}' = W' - \mathbf{E}W'$.

P r o o f. Using a simple lower bound $OPT \geq m(n-1)a_n$, we get the following inequality chain:

$$\begin{aligned} \mathbf{P}\big\{W > (1+\varepsilon_n)OPT\big\} &\leq \mathbf{P}\big\{m(n-1)a_n + \beta_n W' > (1+\varepsilon_n)m(n-1)a_n\big\} \\ = \mathbf{P}\Big\{W' > \frac{\varepsilon_n m(n-1)a_n}{\beta_n}\Big\} &= \mathbf{P}\Big\{W' - \mathbf{E}W' > \frac{\varepsilon_n m(n-1)a_n}{\beta_n} - \mathbf{E}W'\Big\} \\ \leq \mathbf{P}\Big\{\widetilde{W}' > \frac{\varepsilon_n m(n-1)a_n}{\beta_n} - \mathcal{E}\Big\} &\leq \mathbf{P}\big\{\widetilde{W}' \geq \mathcal{E}\big\} = \delta_n, \end{aligned}$$

where in the last inequality we substituted expression (4.7) for $\varepsilon_n$. □

To obtain a more explicit form of the upper bound (4.7) for the relative error $\varepsilon_n$, we need to derive an upper bound $\mathcal{E}$ on the expectation $\mathbf{E}W'$. This will be done for distributions

$$\mathcal{D} \in \big\{\mathrm{U}(a_n, b_n), \text{d-}\mathrm{U}(a_n, b_n), \mathrm{Exp}(a_n, \lambda_n)\big\}$$

in Lemmas 4 and 7. In turn, to obtain a more explicit form of the upper bound (4.8) for the failure probability $\delta_n$ we apply Petrov's theorem (Proposition 1 below) from [23], and show how to choose appropriate constants $h_k$ required in Petrov's theorem in Lemmas 5, 6, and 8 for the considered distributions.

**Proposition 1** (Petrov's theorem [23]). *Consider independent random variables* $X_1, \dots, X_n$. *Let there be positive constants* $T > 0$ *and* $h_1, \dots, h_n > 0$ *such that, for all* $k = 1, \dots, n$ *and* $0 \le t \le T$, *it holds that*

$$\mathbf{E} e^{tX_k} \le e^{h_k t^2/2}. \tag{4.9}$$

*Let* $S = \sum_{k=1}^n X_k$ *and* $H = \sum_{k=1}^n h_k$. *Then*

$$\mathbf{P}\{S > x\} \le \begin{cases} \exp\{-x^2/(2H)\} & \text{if } \ 0 \le x \le HT, \\ \exp\{-Tx/2\} & \text{if } \ x \ge HT. \end{cases}$$

## 4.2. Properties of the random variable $\xi_k$ for selected distributions

In this section, we gather technical lemmas estimating $\mathbf{E}\xi_k$ and $\mathbf{E}\exp(t(\xi_k - \mathbf{E}\xi_k))$, where $\xi_k$ is a random variable, normalized as in (4.6), equal to the minimum of $k \in \mathbb{N}$ independent random variables with distribution

$$\mathcal{D} \in \{\mathrm{U}(a_n, b_n), \text{d-U}(a_n, b_n), \mathrm{Exp}(a_n, \lambda_n)\}, \quad a_n > 0.$$

The estimates for $\mathbf{E}\xi_k$ are fairly straightforward. Estimates for $\mathbf{E}\exp(t(\xi_k - \mathbf{E}\xi_k))$ in the cases of $\mathcal{D} \in \{\mathrm{U}(a_n, b_n), \mathrm{Exp}(a_n, \lambda_n)\}$ were previously obtained in [8, 17], and we recall these results below without proofs. Although the case $\mathcal{D} = \text{d-U}(a_n, b_n)$ was already studied in [9], we provide here a more accurate and complete proof for this case. Below we will need the following standard facts on integrals, left and midpoint sums, and concave functions.

**Proposition 2.** *Let* $f(x)$ *be a nonnegative integrable function on* $[a, b+1]$. *Then*

$$\sum_{i=a}^{b} f(i) \le \int_a^{b+1} f(x)\,dx \quad \text{if } f(x) \text{ is nondecreasing on } [a, b+1]; \tag{4.10}$$

$$\frac{f(a) + f(b)}{2}(a - b) \le \int_a^b f(x)\,dx \quad \text{if } f(x) \text{ is concave on } [a, b]; \tag{4.11}$$

$$\sum_{i=a}^{b} f(i) \le \frac{f(a) + f(b)}{2} + \int_a^b f(x)\,dx \quad \text{if } f(x) \text{ is nonincreasing and concave on } [a, b]. \tag{4.12}$$

The next lemma estimates $\mathbf{E}\xi_k$ for $\mathcal{D} \in \{\mathrm{U}(a_n, b_n), \text{d-U}(a_n, b_n), \mathrm{Exp}(a_n, \lambda_n)\}$.

**Lemma 4.** $\mathbf{E}\xi_k = 1/(k+1)$ *for* $\mathcal{D} = \mathrm{U}(a_n, b_n)$, $\mathbf{E}\xi_k \le 1/(k+1)$ *for* $\mathcal{D} = \text{d-U}(a_n, b_n)$, *and* $\mathbf{E}\xi_k = 1/k$ *for* $\mathcal{D} = \mathrm{Exp}(a_n, \lambda_n)$.

P r o o f. For $\mathcal{D} = \mathrm{U}(a_n, b_n)$, we have

$$\mathbf{E}\xi_k = \int_{-\infty}^{\infty} x\, d\big(1 - (1 - F_{\mathrm{U}(0,1)}(x))^k\big) = k \int_0^1 x(1-x)^{k-1}\,dx = k \int_0^1 (1-t)t^{k-1}\,dt = \frac{1}{k+1}.$$

For $\mathcal{D} = \mathrm{Exp}(a_n, \lambda_n)$ we have

$$\mathbf{E}\xi_k = \frac{1}{\lambda_n} \int\limits_{a_n}^{\infty} x d\big(1 - (1 - F_{\mathrm{Exp}(a_n,\lambda)}(x))^k\big) - \frac{a_n}{\lambda_n} = -\frac{1}{\lambda_n} \int\limits_{a_n}^{\infty} x\, d\exp\left(\frac{-k(x - a_n)}{\lambda_n}\right) - \frac{a_n}{\lambda_n}$$
$$= \frac{1}{\lambda_n} \int_{a_n}^{\infty} \exp\left(\frac{-k(x - a_n)}{\lambda_n}\right) dx = \frac{1}{k}.$$

For $\mathcal{D} = \text{d-U}(a_n, b_n)$, the computations are slightly more involved. Let $\xi$ be a normalized (4.6) random variable with distribution $\text{d-U}(a_n, b_n)$.

Note that $\mathbf{P}\{\xi \le x/\beta_n\} = (x+1)/\beta_n$ and $\mathbf{P}\{\xi \ge (x+1)/\beta_n\} = (\beta_n - x - 1)/\beta_n$ for $x \in \{0, 1, \dots, \beta_n - 1\}$, while $\mathbf{P}\{\xi \ge x/\beta_n\} = 0$ for $x \ge \beta_n$. Then we have

$$\begin{aligned}
\mathbf{E}\xi_k &= \sum_{x=0}^{\beta_n-1} \frac{x}{\beta_n}\mathbf{P}\left\{\xi_k = \frac{x}{\beta_n}\right\} = \sum_{x=0}^{\beta_n-1} \frac{x}{\beta_n}\left(\mathbf{P}\Big\{\xi_k \ge \frac{x}{\beta_n}\Big\} - \mathbf{P}\Big\{\xi_k \ge \frac{x+1}{\beta_n}\Big\}\right) \\
&= \frac{1}{\beta_n}\left(\sum_{x=0}^{\beta_n-2}(x+1)\mathbf{P}\Big\{\xi_k \ge \frac{(x+1)}{\beta_n}\Big\} - \sum_{x=0}^{\beta_n-1} x\mathbf{P}\Big\{\xi_k \ge \frac{x+1}{\beta_n}\Big\}\right) \\
&= \frac{1}{\beta_n}\sum_{x=0}^{\beta_n-1}(x+1-x)\mathbf{P}\Big\{\xi_k \ge \frac{x+1}{\beta_n}\Big\} = \frac{1}{\beta_n}\sum_{x=0}^{\beta_n-1}\Big(\mathbf{P}\Big\{\xi \ge \frac{x+1}{\beta_n}\Big\}\Big)^k \\
&= \frac{1}{\beta_n}\sum_{x=0}^{\beta_n-1}\Big(\frac{\beta_n - x - 1}{\beta_n}\Big)^k = \frac{1}{\beta_n^{k+1}}\sum_{i=0}^{\beta_n-1} i^k \le \Big(\frac{1}{\beta_n}\Big)^{k+1}\int_0^{\beta_n} x^k\,dx = \frac{1}{k+1}.
\end{aligned}$$

□

The next lemma shows how to choose constants $h_k$, so that the variables

$$\widetilde{\xi}_k = \xi_k - \mathbf{E}\xi_k$$

satisfy condition (4.9) of Petrov's theorem for $\mathcal{D} \in \{\text{U}(a_n, b_n), \text{Exp}(a_n, \lambda_n)\}$.

**Lemma 5.** *Let* $\widetilde{\xi}_k = \xi_k - \mathbf{E}\xi_k$, $k \in \mathbb{N}$. *For* $k \in \mathbb{N}$, *let*

$$h_k = \begin{cases} 3/(k+1)^2 & \textit{if} \quad \mathcal{D} = \text{U}(a_n, b_n), \\ 3/k^2 & \textit{if} \quad \mathcal{D} = \text{Exp}(a_n, \lambda_n), \end{cases} \quad \textit{and} \quad T = \begin{cases} 1 & \textit{if} \quad \mathcal{D} = \text{U}(a_n, b_n), \\ 1/2 & \textit{if} \quad \mathcal{D} = \text{Exp}(a_n, \lambda_n). \end{cases}$$

*Then for all* $t \in [0, T]$ *and each* $k \in \mathbb{N}$, *it holds that*

$$\mathbf{E}\exp(t\widetilde{\xi}_k) \le \exp\Big(\frac{h_k t^2}{2}\Big).$$

P r o o f. The proof for $\mathcal{D} = \text{U}(a_n, b_n)$ is given in [8, Lemma 4]. For $\mathcal{D} = \text{Exp}(a_n, \lambda_n)$ and nonnormalized random variables $\eta_k = \lambda_n \xi_k$, in [17, Lemma 10], it was shown that for all $t \in [0, 1/(2\lambda_n)]$,

$$\mathbf{E}\exp(t(\eta_k - \mathbf{E}\eta_k)) \le \exp\left(\frac{(3\lambda_n^2/k^2)t^2}{2}\right)$$

Thus, for $\xi_k$, it follows that for all $t \in [0, 1/2]$

$$\mathbf{E}\exp\big(t(\xi_k - \mathbf{E}\xi_k)\big) = \mathbf{E}\exp\left(\frac{t}{\lambda_n}(\eta_k - \mathbf{E}\eta_k)\right) \le \exp\left(\frac{(3/k^2)t^2}{2}\right).$$

□

Below we give a complete and accurate proof for a similar statement in the case of $\mathcal{D} = \text{d-U}(a_n, b_n)$.

**Lemma 6.** *Let* $\mathcal{D} = \text{d-U}(a_n, b_n)$, $\widetilde{\xi}_k = \xi_k - \mathbf{E}\xi_k$, *and* $h_k = 3/(k+1)^2$, $k \in \mathbb{N}$. *Then for each* $t \in [0,1]$ *and each* $k \in \mathbb{N}$, *it holds that* $\mathbf{E}\exp(t\widetilde{\xi}_k) \le \exp(h_k t^2/2)$.

P r o o f. We estimate $\exp(t\widetilde{\xi}_k)$, taking into account that $\widetilde{\xi}_k \le 1$ and $t \le 1$:

$$\exp(t\widetilde{\xi}_k) = \sum_{j=0}^{\infty} \frac{(t\widetilde{\xi}_k)^j}{j!} = 1 + t\widetilde{\xi}_k + \frac{(t\widetilde{\xi}_k)^2}{2}\left(1 + \frac{t\widetilde{\xi}_k}{3} + \frac{(t\widetilde{\xi}_k)^2}{3\cdot 4} + \frac{(t\widetilde{\xi}_k)^3}{3\cdot 4\cdot 5} + \cdots\right)$$
$$\le 1 + t\widetilde{\xi}_k + \frac{(t\widetilde{\xi}_k)^2}{2}\sum_{j=0}^{\infty}\left(\frac{t\widetilde{\xi}_k}{3}\right)^j = 1 + t\widetilde{\xi}_k + \frac{(t\widetilde{\xi}_k)^2}{2}\frac{1}{1 - t\widetilde{\xi}_k/3} \le 1 + t\widetilde{\xi}_k + \frac{3(t\widetilde{\xi}_k)^2}{4}.$$

By definition of $\widetilde{\xi}_k$, $\mathbf{E}\widetilde{\xi}_k = 0$ and $\mathbf{E}\widetilde{\xi}_k^2 = \mathbf{Var}\,\xi_k \le \mathbf{E}(\xi_k)^2$. Below we will show that $\mathbf{E}(\xi_k)^2 \le 2/(k+1)^2$, and now, using this fact, we get the required inequality:

$$\mathbf{E}\exp(t\widetilde{\xi}_k) \le 1 + t\mathbf{E}\widetilde{\xi}_k + \frac{3t^2}{4}\mathbf{E}\widetilde{\xi}_k^2 \le 1 + \frac{3t^2}{2(k+1)^2} = 1 + \frac{h_k t^2}{2} \le \exp(h_k t^2/2).$$

It remains to show that $\mathbf{E}(\xi_k)^2 \le 2/(k+1)^2$. To that end, recall that $\mathbf{P}\{\xi \ge (x+1)/\beta_n\} = (\beta_n - x - 1)/\beta_n$ for $x \in \{0, 1, \ldots, \beta_n - 1\}$ and $\mathbf{P}\{\xi \ge x/\beta_n\} = 0$ for $x \ge \beta_n$. Then we have

$$\begin{aligned}
\mathbf{E}(\xi_k)^2 &= \sum_{x=0}^{\beta_n-1}\frac{x^2}{\beta_n^2}\mathbf{P}\left\{\xi_k = \frac{x}{\beta_n}\right\} = \sum_{x=0}^{\beta_n-1}\frac{x^2}{\beta_n^2}\left(\mathbf{P}\Big\{\xi_k \ge \frac{x}{\beta_n}\Big\} - \mathbf{P}\Big\{\xi_k \ge \frac{x+1}{\beta_n}\Big\}\right)\\
&= \frac{1}{\beta_n^2}\left(\sum_{x=0}^{\beta_n-2}(x+1)^2\mathbf{P}\Big\{\xi_k \ge \frac{x+1}{\beta_n}\Big\} - \sum_{x=0}^{\beta_n-1}x^2\mathbf{P}\Big\{\xi_k \ge \frac{x+1}{\beta_n}\Big\}\right)\\
&= \frac{1}{\beta_n^2}\sum_{x=0}^{\beta_n-1}\big((x+1)^2 - x^2\big)\mathbf{P}\Big\{\xi_k \ge \frac{x+1}{\beta_n}\Big\} = \frac{1}{\beta_n^2}\sum_{x=0}^{\beta_n-1}(2x+1)\left(\frac{\beta_n - x - 1}{\beta_n}\right)^k\\
&= \frac{1}{\beta_n^{k+2}}\sum_{i=0}^{\beta_n-1}(2\beta_n - 2i - 1)i^k = \frac{1}{\beta_n^{k+2}}\left(2\sum_{i=0}^{\beta_n}(\beta_n - i)i^k - \sum_{i=0}^{\beta_n-1}i^k\right).
\end{aligned} \tag{4.13}$$

Consider the function $f(x) = (\beta_n - x)x^k$ on $[0, \beta_n]$. It is easy to see that $f'(x) = 0$ at $x^* = \beta_n k/(k+1)$, $f''(x) = 0$ at $x^{**} = \beta_n(k-1)/(k+1) < x^*$, and $f(x)$ is increasing on $[0, x^*]$ and is decreasing and concave on $[x^*, \beta_n]$. Thus, we have

$$\begin{aligned}
\sum_{i=0}^{\beta_n} f(i) &= \sum_{i=0}^{\lfloor x^*\rfloor - 1} f(i) + f(\lfloor x^*\rfloor) + \sum_{i=\lfloor x^*\rfloor+1}^{\beta_n} f(i)\\
&\le \int_0^{\lfloor x^*\rfloor} f(x)\,dx + f(\lfloor x^*\rfloor) + \int_{\lfloor x^*\rfloor+1}^{\beta_n} f(x)\,dx + \frac{1}{2}f(\lfloor x^*\rfloor + 1)\\
&= \int_0^{\beta_n} f(x)\,dx - \int_{\lfloor x^*\rfloor}^{\lfloor x^*\rfloor+1} f(x)\,dx + f(\lfloor x^*\rfloor) + \frac{1}{2}f(\lfloor x^*\rfloor + 1)
\end{aligned}$$

$$\begin{aligned}
&\le \int_0^{\beta_n} f(x)\,dx - \int_{x^*}^{\lfloor x^*\rfloor+1} f(x)\,dx + f(\lfloor x^*\rfloor)(\lfloor x^*\rfloor+1-x^*) + \frac{1}{2}f(\lfloor x^*\rfloor+1)\\
&\le \int_0^{\beta_n} f(x)\,dx - \frac{f(x^*)+f(\lfloor x^*\rfloor+1)}{2}(\lfloor x^*\rfloor+1-x^*) + f(\lfloor x^*\rfloor)(\lfloor x^*\rfloor+1-x^*) + \frac{1}{2}f(\lfloor x^*\rfloor+1)\\
&\le \int_0^{\beta_n} f(x)\,dx + \frac{f(\lfloor x^*\rfloor)}{2}(\lfloor x^*\rfloor+1-x^*) + \frac{f(\lfloor x^*\rfloor+1)}{2}(x^*-\lfloor x^*\rfloor) \qquad (4.14)\\
&\le \frac{\beta_n^{k+2}}{k+1} - \frac{\beta_n^{k+2}}{k+2} + \frac{1}{2}\max\{f(\lfloor x^*\rfloor), f(\lfloor x^*\rfloor+1)\}\\
&= \frac{\beta_n^{k+2}}{(k+1)(k+2)} + \frac{1}{2}\max\{f(\lfloor x^*\rfloor), f(\lfloor x^*\rfloor+1)\},
\end{aligned}$$

where the first inequality is due to (4.10) and (4.12), the second inequality is due to (4.10) and the fact that $f(x)$ is increasing on $[\lfloor x^*\rfloor, x^*]$, and the third inequality is due to (4.11) and the fact that $f(x)$ is concave on $[x^*, \lfloor x^*\rfloor+1]$. Next, note that for $x \in \{0,1,\ldots,\beta_n-1\}$

$$f(x) = (\beta_n - x)x^k = \sum_{i=x}^{\beta_n-1} x^k \le \sum_{i=x}^{\beta_n-1} i^k. \qquad (4.15)$$

Finally, plugging (4.14) and (4.15) into (4.13), we obtain

$$\mathbf{E}(\xi_k)^2 \le \frac{1}{\beta_n^{k+2}} \cdot \frac{2\beta_n^{k+2}}{(k+1)(k+2)} \le \frac{2}{(k+1)^2}.$$

□

## 4.3. Asymptotic optimality

In this section, we put everything together and show the conditions under which Algorithm 1 is asymptotically optimal for distributions $\mathcal{D} \in \{\mathrm{U}(a_n,b_n), \mathrm{d\text{-}U}(a_n,b_n), \mathrm{Exp}(a_n,\lambda_n)\}$. As mentioned above, to obtain an explicit upper bound on the relative error from (4.7), we need to estimate $\mathbf{E}(W')$, whereas to obtain an explicit upper bound for the failure probability from (4.8) we will apply Petrov's theorem (Proposition 1), which requires estimating the sum $H$ of certain constants $h_k$. These estimates are provided in the following two lemmas.

**Lemma 7.** *For* $\mathcal{D} \in \{\mathrm{U}(a_n,b_n), \mathrm{d\text{-}U}(a_n,b_n), \mathrm{Exp}(a_n,\lambda_n)\}$, $\mathbf{E}(W') \le \mathcal{E}$, *where*

$$\mathcal{E} = 2\ln(n/2) + 2\ell m + \frac{nm}{\lfloor (D-1)\ell/2\rfloor}.$$

P r o o f. By Lemma 4, $\mathbf{E}\xi_k \le 1/k$ for all $\mathcal{D} \in \{\mathrm{U}(a_n,b_n), \mathrm{d\text{-}U}(a_n,b_n), \mathrm{Exp}(a_n,\lambda_n)\}$, $k \in \mathbb{N}$. Consider separately the expectations for $W_1'$, $W_2'$, and $W_3'$. From equations (4.3)–(4.5), we obtain

$$\begin{aligned}
\mathbf{E}W_1' = \sum_{i=1}^{m}\sum_{j=1}^{2\ell}\mathbf{E}\xi_1 + \sum_{i=1}^{m}\sum_{r=1}^{\ell}\sum_{k=1}^{D-1}\mathbf{E}\xi_{\lfloor n/2\rfloor - D\ell(i-1) - \lfloor i/2\rfloor - (r-1)D - k}\\
+\mathbf{E}\xi_{\lceil n/2\rceil - D\ell(i-1) - \lceil i/2\rceil - (r-1)D - k} \le 2m\ell + \sum_{j=1}^{m(D-1)\ell}\mathbf{E}\xi_{\lfloor n/2\rfloor - j} + \mathbf{E}\xi_{\lceil n/2\rceil - j}
\end{aligned}$$

$$\leq 2m\ell + \sum_{j=1}^{m(D-1)\ell} \frac{2}{\lfloor n/2 \rfloor - j} \leq 2m\ell + \sum_{j=1}^{\lfloor n/2 \rfloor -1} \frac{2}{j} < 2m\ell + 2\ln(n/2);$$

$$\mathbf{E}W_2' = \sum_{i=1}^{m} \sum_{j=1, j\neq i}^{m} \mathbf{E}\xi_{\ell(D-1)} + \sum_{i=1}^{m} \sum_{j=1, j\neq i}^{m} \sum_{r=1}^{2} \sum_{k=1, k\%2=1}^{D\ell} \mathbf{E}\xi_{\lceil (D-1)\ell/2 \rceil}$$
$$+ \sum_{i=1}^{m} \sum_{j=1, j\neq i}^{m} \sum_{r=1}^{2} \sum_{k=1, k\%2=0}^{D\ell} \mathbf{E}\xi_{\lfloor (D-1)\ell/2 \rfloor}$$
$$= m(m-1)\big(D\ell \mathbf{E}\xi_{\lceil (D-1)\ell/2 \rceil} + D\ell \mathbf{E}\xi_{\lfloor (D-1)\ell/2 \rfloor} + \mathbf{E}\xi_{\ell(D-1)}\big) \leq \frac{D\ell m^2}{\lfloor (D-1)\ell/2 \rfloor} + \frac{(2D\ell+1)m^2}{(D-1)\ell};$$

$$\mathbf{E}W_3' = \sum_{i=1}^{m} \sum_{j=1}^{\lfloor n/2 \rfloor - mD\ell - \lfloor m/2 \rfloor} \mathbf{E}\xi_{(D-1)\ell} + \sum_{i=1}^{m} \sum_{j=1}^{\lceil n/2 \rceil - mD\ell - \lceil m/2 \rceil} \mathbf{E}\xi_{(D-1)\ell} \leq m\frac{n-m-2mD\ell}{(D-1)\ell}.$$

Finally, taking into account that $2D\ell m < n$, we get

$$\mathbf{E}(W') = \mathbf{E}W_1' + \mathbf{E}W_2' + \mathbf{E}W_3' \leq 2m\ell + 2\ln(n/2) + \frac{D\ell m^2}{\lfloor (D-1)\ell/2 \rfloor} + \frac{nm}{(D-1)\ell}$$
$$< 2m\ell + 2\ln(n/2) + \frac{nm}{2\lfloor (D-1)\ell/2 \rfloor} + \frac{nm}{(D-1)\ell} \leq 2\ln(n/2) + 2\ell m + \frac{nm}{\lfloor (D-1)\ell/2 \rfloor}.$$

□

**Lemma 8.** *Let $h_k \leq 3/k^2$, $k \in \mathbb{N}$. For each $i = 1, 2, 3$, consider the sum $H_i$ corresponding to $W_i'$, where each summand $\xi_k^j$ in $W_i'$ for every superscript $j$ is substituted with $h_k$ in $H_i$. Then*

$$H = H_1 + H_2 + H_3 < 6\Big(\ell m + 2 + \frac{nm}{(\lfloor (D-1)\ell/2 \rfloor)^2}\Big).$$

P r o o f. Taking into account equations (4.3)–(4.5) and that $2mD\ell < n$, we estimate $H_1$, $H_2$, and $H_3$ as follows:

$$H_1 < 2\ell m h_1 + 2\sum_{i=1}^{n/2-1} h_{n/2-i} \leq 6\ell m + \sum_{i=1}^{n/2-1} \frac{6}{(n/2-i)^2} < 6\ell m + \sum_{j=1}^{\infty} \frac{6}{j^2} = 6\ell m + \pi^2;$$

$$H_2 = m(m-1)D\ell\big(h_{\lfloor (D-1)\ell/2 \rfloor} + h_{\lceil (D-1)\ell/2 \rceil}\big) + m(m-1)h_{\ell(D-1)}$$
$$\leq 3m^2\Big(\frac{D\ell}{(\lfloor (D-1)\ell/2 \rfloor)^2} + \frac{4D\ell+1}{((D-1)\ell)^2}\Big);$$

$$H_3 = m(\lfloor n/2 \rfloor - mD\ell - \lfloor m/2 \rfloor + \lceil n/2 \rceil - mD\ell - \lceil m/2 \rceil)h_{(D-1)\ell}$$
$$= m(n - 2mD\ell - m)h_{(D-1)\ell} \leq \frac{3(nm - (2D\ell+1)m^2)}{((D-1)\ell)^2}.$$

Finally, taking into account that $2D\ell m < n$, we obtain

$$H = H_1 + H_2 + H_3 < 3\Big(2\ell m + 4 + \frac{D\ell m^2}{(\lfloor (D-1)\ell/2 \rfloor)^2} + \frac{2D\ell m^2 + nm}{((D-1)\ell)^2}\Big)$$
$$< 3\Big(2\ell m + 4 + \frac{nm}{2(\lfloor (D-1)\ell/2 \rfloor)^2} + \frac{2nm}{4(\lfloor (D-1)\ell/2 \rfloor)^2}\Big) \leq 6\Big(\ell m + 2 + \frac{nm}{(\lfloor (D-1)\ell/2 \rfloor)^2}\Big).$$

□

Now we are ready to prove the main theorem of the paper.

**Theorem 1.** *Algorithm 1 is asymptotically optimal on random instances with distribution*

$$\mathcal{D} \in \{\mathrm{U}(a_n, a_n + \beta_n),\ \text{d-U}(a_n, a_n + \beta_n - 1),\ \mathrm{Exp}(a_n, \beta_n)\},$$

*if*

$$\beta_n/a_n = o(\sqrt{n}). \tag{4.16}$$

P r o o f. Recall that

$$D \geq 2, \quad \ell = \lceil \sqrt{n}/(2D) \rceil, \quad D\ell \geq \sqrt{n}/2, \quad m(1 + 2D\ell) \leq n,$$

and thus, $m < \sqrt{n}$. By Lemma 7 and expression (4.7), the relative error $\varepsilon_n$ of the algorithm satisfies:

$$\varepsilon_n = \frac{2\beta_n \mathcal{E}}{m(n-1)a_n} = \frac{4\beta_n}{a_n(n-1)}\left(\frac{\ln(n/2)}{m} + \ell + \frac{n}{2\lfloor (D-1)\ell/2 \rfloor}\right)$$
$$< \frac{4\beta_n}{a_n(n-1)}\left(\frac{\ln(n/2)}{m} + \frac{\sqrt{n}}{4} + 1 + \frac{n}{\sqrt{n}/2 - \sqrt{n}/4 - 2}\right) = O\left(\frac{\beta_n}{a_n} \cdot \frac{1}{\sqrt{n}}\right) = o(1),$$

when (4.16) holds.

To estimate the failure probability $\delta_n$, we extend (4.8) using Petrov's theorem (Proposition 1). To that end, first note that, according to (4.3)–(4.5), the sum of random variables $W'$ in (4.8) is a sum of random variables only of the type $\xi_k$, defined by (4.6). By Lemmas 5–6, these variables satisfy the conditions of Petrov's theorem (Proposition 1) with constants $T = 1/6$ and $h_k = 3/k^2$ for distribution $\mathcal{D} \in \{\mathrm{U}(a_n, b_n), \text{d-U}(a_n, b_n), \mathrm{Exp}(a_n, \lambda_n)\}$. Using $\mathcal{E}$ from Lemma 7 and an upper bound for $H$ from Lemma 8, we show that $\mathcal{E} \geq TH$ for $n > 16$:

$$\mathcal{E} - TH \geq 2\ln(n/2) + 2\ell m + \frac{nm}{\lfloor (D-1)\ell/2 \rfloor} - \left(2 + \ell m + \frac{nm}{(\lfloor (D-1)\ell/2 \rfloor)^2}\right)$$
$$= (2\ln(n/2) - 2) + \ell m + \frac{nm}{\lfloor (D-1)\ell/2 \rfloor}\left(1 - \frac{1}{\lfloor (D-1)\ell/2 \rfloor}\right) \geq 0,$$

since $\lfloor (D-1)\ell/2 \rfloor \geq 1$ for $n > 16$.

Now, applying Petrov's theorem (Proposition 1) to (4.8), we get

$$\delta_n = \mathbf{P}\left\{\widetilde{W'} \geq \mathcal{E}\right\} \leq \exp\left(-T\mathcal{E}/2\right) \leq \exp\left(-4m\sqrt{n}/12\right) = \exp\left(-m\sqrt{n}/3\right),$$

since from the definition of $\mathcal{E}$ in Lemma 7, it follows that

$$\mathcal{E} \geq \frac{nm}{\lfloor (D-1)\ell/2 \rfloor} \geq \frac{nm}{\sqrt{n}/4} = 4m\sqrt{n}.$$

Summing up, as $n \to \infty$, the failure probability $\delta_n \to 0$ and the relative error $\varepsilon_n \to 0$, provided that $\beta_n/a_n = o(\sqrt{n})$, which completes the proof of the theorem. □

Note that constraint (4.16) on the distribution parameters in Theorem 1 is not particularly restrictive: the parameters are not only allowed to be constant, but their ratio may grow slowly with $n$. Also note that Theorem 1 can be extended to the case of dominating distributions (see Definition 1).

*Remark 1.* Let $\beta_n/a_n = o(\sqrt{n})$. For the $m$-$d$-MST on random inputs with distribution $\widehat{\mathcal{D}}$, which has support $[a_n, y]$, $y \in \mathbb{R} \cup \{\infty\}$, and dominates the distribution

$$\mathcal{D} \in \{\mathrm{U}(a_n, a_n + \beta_n),\ \text{d-U}(a_n, a_n + \beta_n - 1),\ \mathrm{Exp}(a_n, \beta_n)\},$$

Algorithm 1 provides asymptotically optimal solutions.

The proof follows from Propositions 2 and 3 presented in [18], which imply that if distribution $\widehat{\mathcal{D}}$ dominates $\mathcal{D}$, then for all $x \in [a_n, y]$,

$$\mathbf{P}\Big\{\sum_{k\in J} \widehat{\rho}_k \le x\Big\} \ge \mathbf{P}\Big\{\sum_{k\in J} \rho_k \le x\Big\},$$

where each summand $\rho_k$ ($\widehat{\rho}_k$) is an independent random variable equal to the minimum of $k$ i. i. d. random variables with distribution $\mathcal{D}$ ($\hat{\mathcal{D}}$). Then, since (4.3)–(4.5) hold,

$$\mathbf{P}\Big\{\widehat{W} > (1+\varepsilon_n)OPT\Big\} \le \mathbf{P}\Big\{\widehat{W} > (1+\varepsilon_n)am(n-1)\Big\} \le \mathbf{P}\Big\{W > (1+\varepsilon_n)am(n-1)\Big\} \le \delta_n,$$

where $\widehat{W}$ and $W$ are the weights of the solutions constructed by Algorithm 1 in the case of distributions $\widehat{\mathcal{D}}$ and $\mathcal{D}$, while the performance guarantees $\varepsilon_n$ and $\delta_n$ are as in Theorem 1.

## 5. Conclusion

In this paper, we studied the $m$-$d$-MST problem on random inputs with continuous and discrete uniform distributions, shifted exponential distribution, and distributions dominating the above. We proposed a new polynomial-time approximation algorithm and showed the conditions under which the algorithm is asymptotically optimal for the considered classes of random inputs. The result holds for any $d \ge 4$, whereas the algorithms from previous papers, e.g., [9, 14], required $d = \omega(1)$ to obtain asymptotically optimal solutions for the same classes of random inputs.

It is easy to see that the proposed algorithm can be used to solve the Given-Diameter $m$-$d$-MST problem for $4 \le d \le \sqrt{n}$. For larger $d$, one can either modify the current algorithm along the lines of [15] or use algorithms from [9, 14]. However, in both cases, since the algorithms start by constructing $m$ vertex-disjoint base trees of diameter $d$, for very large $d$ they are only suitable for solving instances with small $m$, in some cases only with $m = 1$. To a certain extent, this limitation can be alleviated by a different approach proposed in [16] for the $m$-Peripatetic Salesmen Problem: first, uniformly at random, divide the edges of a given graph into $m$ parts $E_1, \dots, E_m$, and then, with high probability, construct a suitable single MST in each $G[E_i]$, $i = 1, \dots, m$.

## Acknowledgements

The study was carried out within the framework of the state contract of the Sobolev Institute of Mathematics (project FWNF-2022-0019).

## REFERENCES

1. Anderson R., Stenger B., Cipolla R. Using bounded diameter minimum spanning trees to build dense active appearance models. *Int. J. Comput. Vis.*, 2014. Vol. 110. P. 48–57. DOI: 10.1007/s11263-013-0661-9
2. Angel O., Flaxman A. D., Wilson D. B. A sharp threshold for minimum bounded-depth and bounded-diameter spanning trees and Steiner trees in random networks. *Combinatorica*, 2012. Vol. 32, P. 1–33. DOI: 10.1007/s00493-012-2552-z

3. Bala K., Petropoulos K., Stern T. E. Multicasting in a linear lightwave network. In: *Proc. IEEE INFOCOM'93 The Conf. on Computer Communications, San Francisco.* CA, USA, 1993. Vol. 3. P. 1350–1358. DOI: 10.1109/INFCOM.1993.253399
4. Bar-Ilan J., Kortsarz G., Peleg D. Generalized submodular cover problems and applications. *Theoret. Comput. Sci.*, 2001. Vol. 250, No. 1–2. P. 179–200. DOI: 10.1016/S0304-3975(99)00130-9
5. Boruvka O. O jistém problému minimálním. *Prace Mor. Prirodoved. Spol. V Brne III*, 1926. Vol. 3, P. 37–58. (in Czech)
6. Frieze A. On the value of a random minimum spanning tree problem. *Discrete Appl. Math.*, 1985. Vol. 10, No. 1. P. 47–56. DOI: 10.1016/0166-218X(85)90058-7
7. Garey M. R., Johnson D. S. *Computers and Intractability; A Guide to the Theory of NP-Completeness.* USA: W.H. Freeman & Co.: 1990. 338 p.
8. Gimadi E. Kh. On some probability inequalities for some discrete optimization problems. *Oper. Res. Proc., vol. 2005. HD. Haasis, H. Kopfer, J. Schonberger (eds.).* Berlin, Heidelberg: Springer, 2006. P. 283–289. DOI: 10.1007/3-540-32539-5_45
9. Gimadi E. Kh. Several edge-disjoint spanning trees with given diameter in a graph with random discrete edge weights. In: *Commun. Comput. Inf. Sci., vol. 1913: Advances in Optimization and Applications (OPTIMA 2023), N. Olenev et al. (eds.).* Cham: Springer, 2024. P. 281–292. DOI: 10.1007/978-3-031-48751-4_21
10. Gimadi E. Kh., Istomin A. M., Shin E. Yu. On algorithm for the minimum spanning tree problem bounded below. In: *CEUR Workshop Proc., vol. 1623: Suppl. Proc. 9th Int. Conf. on Discrete Optimization and Operations Research and Scientific School (DOOR 2016), Vladivostok, Russia, September 19–23, 2016.* CEUR-WS, 2016. P. 11–17. URL: https://ceur-ws.org/Vol-1623/paperco4.pdf
11. Gimadi E. Kh., Istomin A. M., Shin E. Yu. On bounded diameter MST problem on random instances. In: *CEUR Workshop Proc., vol. 2098: Proc. School-Seminar on Optimization Problems and their Applications (OPTA-SCL 2018), Omsk, Russia, July 8–14, 2018.* CEUR WP, 2018. P. 159–168. URL: https://ceur-ws.org/Vol-2098/paper14.pdf
12. Gimadi E. Kh., Shevyakov A. S., Shtepa A. A. A given diameter MST on a random graph. In: *Lecture Notes in Comput. Sci., vol. 12422: Optimization and Applications (OPTIMA 2020), N. Olenev et al. (eds.).* Cham: Springer, 2020. P. 110–121. DOI: 10.1007/978-3-030-62867-3_9
13. Gimadi E. Kh., Shin E. Yu. Probabilistic analysis of an algorithm for the minimum spanning tree problem with diameter bounded below. *J. Appl. Ind. Math.*, 2015. Vol. 9, No. 4. P. 480–488. DOI: 10.1134/S1990478915040043
14. Gimadi E. Kh., Shtepa A. A. On asymptotically optimal approach for finding of the minimum total weight of edge-disjoint spanning trees with a given diameter. *Autom. Remote Control*, 2023. Vol. 84. P. 772–787. DOI: 10.1134/S0005117923070068
15. Gimadi E. K., Tsidulko O. Y. An asymptotically optimal algorithm for the minimum weight spanning tree with arbitrarily bounded diameter on random inputs. In: *Lecture Notes in Comput. Sci., vol. 15419: Analysis of Images, Social Networks and Texts (AIST 2024), A. Panchenko et al. (eds.).* Cham: Springer, 2025. P. 249–261. DOI: 10.1007/978-3-031-88036-0_15
16. Gimadi E. K., Istomin A. M., Tsidulko O. Y. On asymptotically optimal approach to the $m$-peripatetic salesman problem on random inputs. In: *Lecture Notes in Comput. Sci., vol. 9869: Discrete Optimization and Operations Research (DOOR 2016), Y. Kochetov et al. (eds.).* Cham: Springer, 2016. P. 136–147. DOI: 10.1007/978-3-319-44914-2_11
17. Gimadi E. K., Le Gallou A., Shakhshneyder A. V. Probabilistic analysis of an approximation algorithm for the traveling salesman problem on unbounded above instances. *J. Appl. Ind. Math.*, 2009. Vol. 3. P. 207–221. DOI: 10.1134/S1990478909020070
18. Gimadi E. Kh., Glazkov Yu. V. An asymptotically exact algorithm for one modification of planar three-index assignment problem. *J. Appl. Indust. Math.*, 2007. Vol. 1, No. 4. P. 442–452. DOI: 10.1134/S1990478907040072
19. Gouveia L., Magnanti T. L. Network flow models for designing diameter-constrained minimum-spanning and Steiner trees. *Networks*, 2003. Vol. 41, No. 3. P. 159–173. DOI: 10.1002/net.10069
20. Gruber M. *Exact and Heuristic Approaches for Solving the Bounded Diameter Minimum Spanning Tree Problem.* PhD Thesis. Vienna University of Technology, 2008. 156 p.
21. Kruskal J. B. On the shortest spanning subtree of a graph and the traveling salesman problem. *Proc. Amer. Math. Soc.*, 1956. Vol. 7, No. 1. P. 48–50. DOI: 10.1090/S0002-9939-1956-0078686-7

22. Lemke P. *The Bounded Diameter Two Edge-Disjoint Spanning Trees Problem is NP-Complete*. Inst. Math. Appl., IMA Preprints Series, 1988. https://hdl.handle.net/11299/4860
23. Petrov V. V. *Limit Theorems of Probability Theory. Sequences of Independent Random Variables*. Oxford: Clarendon Press, 1995. 304 p. DOI: 10.1093/oso/9780198534990.001.0001
24. Prim R. C. Shortest connection networks and some generalizations. *Bell Syst. Tech. J.*, 1957. Vol. 36, No. 6. P. 1389–1401. DOI: 10.1002/j.1538-7305.1957.tb01515.x
25. Roskind J., Tarjan R. E. A note on finding minimum-cost edge-disjoint spanning trees. *Math. Oper. Res.*, 1985. Vol. 10, No. 4. P. 701–708. DOI: 10.1287/moor.10.4.701
26. Segal M., Tzfaty O. Finding bounded diameter minimum spanning tree in general graphs. *Comput. Oper. Res.*, 2022. Vol. 144. Art. no. 105822. DOI: 10.1016/j.cor.2022.105822
27. Torkestani J. A. A stable virtual backbone for wireless MANETS. *Telecommun. Syst.*, 2014. Vol. 55. No. 1. P. 137–148. DOI: 10.1007/s11235-013-9760-8