On Algorithm for the Minimum Spanning Trees Problem with Diameter Bounded Below

Edward Kh. Gimadi[1,2], Alexey M. Istomin[1], and Ekaterina Yu. Shin[2]

[1] Sobolev Institute of Mathematics, 4 Acad. Koptyug avenue,
630090 Novosibirsk, Russia
[2] Novosibirsk State University, 2 Pirogova Str. 630090, Novosibirsk, Russia
gimadi@math.nsc.ru, alexeyistomin@gmail.com, Shinus90@yandex.ru

Abstract. The minimum spanning trees problem is to find k edge-disjoint spanning trees in a given undirected weighted graph. It can be solved in polynomial time. In the k minimum spanning trees problem with diameter bounded below (k-MSTBB) there is an additional requirement: a diameter of every spanning tree must be not less than some predefined value d. The k-MSTBB is NP-hard. We propose an asymptotically optimal polynomial time algorithm to solve this problem.

Keywords: minimum spanning tree, minimum spanning trees problem, asymptotically optimal algorithm, probabilistic analysis, bounded below, performance guarantees, random inputs

Introduction

The Minimum Spanning Tree (MST) problem is one of the classic discrete optimization problems. Given $G = (V, E)$ undirected weighted graph, the MST is to find a spanning tree of a minimal weight. The MST is polynomially solvable, there are classic algorithms by Boruvka (1926), Kruskal (1956) and Prim (1957). These algorithms have complexity $\mathcal{O}(n^2)$ and $\mathcal{O}(M \log n)$ where $M = |E|$ and $n = |V|$.

The Minimum Spanning Tree with diameter Bounded Below (MSTBB) problem is a natural generalization of the MST problem. In the MSTBB aim is to find a spanning tree of a minimal weight such that its diameter is not less than some predefined value d. Diameter of a tree is defined as the number of edges on the longest path between two leaves in the tree. The MSTBB is studied in [6]. It is NP-hard since with $d_n = n - 1$ it is equivalent to the problem of finding Hamiltonian Path of a minimal weight in a given graph. Moreover if d_n is constant then the MSTBB can be solved in polynomial time by iterating through all possible simple paths consisting exactly of d_n edges. Thus MSTBB is actual with d_n growing.

By $F_A(I)$ and $OPT(I)$ we denote respectively the approximate (obtained by some approximation algorithm A) and the optimum value of the objective function of the

In: A. Kononov et al. (eds.): DOOR 2016, Vladivostok, Russia, published at http://ceur-ws.org

problem on the input I. According to [5] an algorithm A is said to have *performance guarantees* $\big(\varepsilon_A(n), \delta_A(n)\big)$ on the set of random inputs of the problem of the size n, if

$$\mathbf{Pr}\Big\{F_A(I) > \big(1+\varepsilon_A(n)\big)OPT(I)\Big\} \leq \delta_A(n), \quad (1)$$

where $\varepsilon_A(n)$ is an assessment of *the relative error* of the solution obtained by algorithm A, $\delta_A(n)$ is an estimation of *the failure probability* of the algorithm, which is equal to the proportion of cases when the algorithm does not hold the relative error $\varepsilon_A(n)$ or does not produce any answer at all.

An algorithm A is called *asymptotically optimal* on the class of instances of the problem, if there are such performance guarantees that $\varepsilon_A(n) \to 0$ and $\delta_A(n) \to 0$ as $n \to \infty$.

Let us denote by UNI(a_n, b_n) a class of complete graphs with n vertices where weights of edges are independent identically distributed random variables with uniform distribution on a segment $[a_n, b_n]$. By EXP(a_n, α_n) we will denote a class of complete graphs with n vertices where weights of edges are independent identically distributed random variables with exponential distribution with a probability function

$$p(x) = \begin{cases} \frac{1}{\alpha_n} \exp\left(-\frac{x-a_n}{\alpha_n}\right), \text{ if } a_n \leq x < \infty, \\ \quad 0, \text{ otherwise.} \end{cases}$$

And by N(a_n, σ_n) we will denote a class of complete graphs with n vertices where weights of edges are independent identically distributed random variables with cutted-normal distribution with a probability function

$$p(x) = \begin{cases} \frac{2}{\sqrt{2\pi\sigma_n^2}} \exp\left(-\frac{(x-a_n)^2}{2\sigma_n^2}\right), \text{ if } a_n \leq x < \infty, \\ \quad 0, \text{ otherwise.} \end{cases}$$

In [6] presented an asymptotically optimal algorithm $\widetilde{A}$ with complexity $\mathcal{O}(n^2)$ for graphs which belong to UNI(a_n, b_n) class. On the first stage algorithm $\widetilde{A}$ build a path $P = \{(i_0, i_1), (i_1, i_2), \dots, (i_{d-1}, i_d)\}$ of d edges starting from an arbitrary vertex i_0 and enlarging it by adding successive edges of a minimal weight. On the second stage a spanning tree of a minimal weight $\widetilde{D_n}$ which contains path P is built using Prim algorithm. Built tree is taken as a solution.

The algorithm $\widetilde{A}$ on graphs which belong to UNI(a_n, b_n) class has the following performance guarantees:

$$\varepsilon_n = \mathcal{O}\Big(\frac{b_n/a_n}{n/\ln\frac{n-1}{n-d}}\Big), \delta_n = e^{-0.25(n-d)}.$$

So, $\widetilde{A}$ is asymptotically optimal if

$$b_n/a_n \leq \frac{n}{\ln n}, d = o(n).$$

The same result but for graphs with unlimited edges weights was presented in [7], where presented analysis of algorithm $\widetilde{A}$ for graphs from classes EXP(a_n, α_n) and

$N(a_n, \sigma_n)$. It is shown that $\widetilde{A}$ finds a solution with performance guarantees:

$$\varepsilon_n = O\Big(\frac{\beta_n/a_n}{n/\ln n}\Big), \delta_n = \mathcal{O}(1/n).$$

Thus the algorithm is asymptotically optimal in the case then

$$\frac{\beta_n}{a_n} = o\Big(\frac{n}{\ln n}\Big),$$

where $\beta_n = \alpha_n$ for the graphs from $\mathrm{EXP}(a_n, \alpha_n)$ and σ_n in the case of $N(a_n, \sigma_n)$.

Another generalization of the MST problem is the problem of finding k edge-disjoint spanning trees of a minimal total weight (k-MSTs). The k-MSTs is also polynomially solvable, in [4] by Roskind and Tarjan was presented an algorithm with complexity $\mathcal{O}(n^2 \log(n) + n^2 k^2)$.

In current work a modification of the k-MSTs is considered where the aim is to find k edge-disjoint spanning trees of a minimal total weight such that every tree has a diameter not less than some predefined value d. We will call this problem as k minimum spanning trees problem with diameter bounded below (k-MSTBB). It is NP-hard since with $d_n = n - 1$ and $k = 1$ it is equivalent to a problem of finding Hamiltonian Path of a minimal weight in a given graph.

We propose a polynomial time algorithm to solve the k-MSTBB problem. The algorithm builds k edge-disjoint Hamiltonian chains with d edges each on the preselected set of $d + 1$ vertices. Then by the algorithm by Roskind and Tarjan [4] edge-disjoint spanning trees in the number of k containing these chains are found. Also we present performance guarantees of the algorithm and sufficient conditions of asymptotic optimality for the graphs from $\mathrm{UNI}(a_n, b_n)$.

1 New algorithm A_k

Let us consider $G = (V, E)$, an undirected complete n-vertex graph belonging to $\mathrm{UNI}(a_n, b_n)$.

Our new algorithm is based on ideas of from [6], uses the algorithm by Roskind and Tarjan [4] and algorithm A_{AV} [1] which builds a Hamiltonian path in a given arbitrary graph G_p with probability $1 - \delta_{AV}$.

Algorithm A_k:

1. Build k paths with a number of edges of d on the set of $d + 1$ vertices.
   (a) Randomly remove all but $d + 1$ vertices from graph G, get $G[d]$ inducted by the set of vertices left.
   (b) Build subgraphs $G^1[d]$, ... , $G^k[d]$ by a random procedure: every edge from $G[d]$ randomly gets one color between 1 and k.
   (c) Remove all edges from $G^1[d]$, ... , $G^k[d]$ which weight more than some threshold w, so we get graphs $G^1_w[d]$, ... , $G^k_w[d]$.
   (d) In $G^1_w[d]$, ... , $G^k_w[d]$ find Hamiltonian paths by the algorithm A_{AV}.

2. By the Roskind-Tarjan algorithm find k edges-disjoint spanning trees containing paths built on the previous step.

Threshold value w must be chosen in a way that $G_w^i[d]$ contains a Hamiltonian path. In [2] it was established that on a $d+1$ vertex graph where edge exists with a probability p if the value of p is so that graph contains more than $c(d+1)\log(d+1)$ edges, the graph will have a Hamiltonian path. In the paper [1] presented an algorithm A_{AV} to find such Hamiltonian path with high probability (whp) in $\mathcal{O}(d\ln^2 d)$ time.

Theorem 1. *[1] For all $\alpha > 0$ there exists a $K(\alpha)$ such that if the number of edges in a random graph $N > (\alpha + K(\alpha))(d+1)\ln(d+1)$, where $K(\alpha)$ is sufficiently large, the probability that algorithm A_{AV} finds a Hamiltonian path is $1 - \mathcal{O}(d^{-\alpha})$.*

We will denote a threshold value of algorithm A_{AV} as $c_{AV} = (\alpha + K(\alpha))$. Now let us formulate a Lemma to provide an estimation of probability that $G_w^i[d]$ contains enough edges for A_{AV} to succeed.

Lemma 1. *The probability that $G_w^i[d]$ contains less than N edges is less than e^{-d} if $p \geq \frac{4(N+d)}{n(n-1)}$.*

Proof.

$$\delta_N = Pr\{G_w^i[d] \text{ contains less than } N \text{ edges }\} = \sum_{k=0}^{N-1} \binom{\frac{d(d-1)}{2}}{k} p^k (1-p)^{\frac{d(d-1)}{2} - k}.$$

Now by the inequality

$$\sum_{k=0}^{\lfloor (1-\beta)np \rfloor} \binom{n}{k} p^k (1-p)^{n-k} \leq e^{\frac{-\beta^2 np}{2}}.$$

which is correct for all n, p, β, where n is integer, $p \in [0,1]$ and $\beta \in [0,1]$, we get

$$\delta_N \leq e^{-\frac{d(d-1)}{4}p + N} \leq e^{-d},$$

if $p \geq \frac{4(N+d)}{d(d-1)}$.

Let's denote δ_{AV} a probability of a case when the algorithm A_{AV} was unable to give us a solution: $G_w^i[d]$ had not enough edges or algorithm A_{AV} returned failure. By Theorem 1 and Lemma 1

$$\delta_{AV} = \mathcal{O}(d^{-\alpha} + e^{-d}).$$

Let $\tilde{D_n}$ be a solution of the k-MSTBB found by algorithm A_k on a n-vertex graph. By w_0 we will denote a weight of solution of the k-MSTs. Now, let us formulate the following Lemma.

Lemma 2. *If all $C_i, i \in \{1, 2, \dots k\}$, were successfully built by the algorithm A_k, then $w(\tilde{D_n}) \leq w_0 + w(C_1) + \dots + w(C_k) - kda_n$.*

Proof. It is obvious that $\tilde{D_n}$ is a solution of the k-MSTs problem with a modified weight function, if w_{ij} is an original weight of a edge (i,j) then modified one will be

$$w'_{ij} = \begin{cases} w_{ij}, \text{ if } (i,j) \notin \bigcup_{t=1}^{k} C_t, \\ a_n, \text{ if } \in \bigcup_{t=1}^{k} C_t. \end{cases}$$

$\tilde{D_n}$ is a group of k edge-disjoint spanning trees with a minimum total weight among a set $\mathbb{D}_n$ of all possible groups of k edge-disjoint spanning trees, where edges weights are w_{ij}.

On the one hand we have:

$$\min\left\{\sum_{(i,j)\in D_n} w'_{ij} | D_n \in \mathbb{D}_n\right\} = \sum_{(i,j)\in \tilde{D_n}} w'_{ij} =$$

$$\sum_{(i,j)\in C_1} w'_{ij} + \ldots + \sum_{(i,j)\in C_k} w'_{ij} + w(\tilde{D_n}) - w(C_1) - \ldots - w(C_k) =$$

$$kda_n + w(\tilde{D_n}) - w(C_1) - \ldots - w(C_k).$$

On the other hand, since $w'_{ij} \leq w_{ij}$:

$$\min\left\{\sum_{(i,j)\in D_n} w'_{ij} | D_n \in \mathbb{D}_n\right\} \leq \min\left\{\sum_{(i,j)\in D_n} w_{ij} | D_n \in \mathbb{D}_n\right\} =$$

$$\min\left\{w(D_n) | D_n \in \mathbb{D}_n\right\} = w_0.$$

Combination of these two inequalities gives us the result of the Lemma.

Lemma 3. *If C_i was successfully built by the algorithm A_k, then*

$$w(C_i) \leq da_n + 5kc_{AV}(b_n - a_n)\log d, i = 1, \ldots, k,$$

where c_{AV} is a threshold value defined for algorithm A_{AV} by Theorem 1 [1].

Proof. Let $i \in \{1, \ldots, k\}$. Let w be $(b_n - a_n)p + a_n$, since edges weights have uniform distribution on a segment $[a_n, b_n]$ (we are considering graphs from class UNI(a_n, b_n)), the probability that an edge have weight less than w is equal to p. To grant to algorithm A_{AV} a possibility to find a Hamiltonian path graph $G^i_w[d]$ must have more than $c_{AV}(d+1)\log(d+1)$ edges. Together with Lemma 1 it means

$$p \leq 5kc_{AV}\frac{\log d}{d}.$$

It give us that every edge in $G^i_w[d]$ has weight less than

$$5kc_{AV}(b_n - a_n)\frac{\log d}{d} + a_n.$$

C_i contains d edges, so

$$w(C_i) \leq 5kc_{AV}(b_n - a_n)\log d + da_n.$$

Theorem 2. *Algorithm A_k has the following performance guarantees:*

$$\varepsilon_n = 5kc_{AV} \log d \frac{b_n - a_n}{a_n n},$$

$$\delta_n = \mathcal{O}\left(k\left(\frac{1}{d^\alpha} + \frac{1}{e^d}\right)\right),$$

where $c_{AV} = (\alpha + K(\alpha))$ is a constant, $\alpha > 0$.

Proof. Let us first assume that we are under the condition that every C_i, $i = 1, \ldots, k$, was successfully found by the algorithm A_{AV}. Using inequality $kna_n \le w_0 \le w(\tilde{D_n})$ and Lemma 2 we have:

$$Pr\left\{w(\tilde{D_n}) > (1 + \varepsilon_n)w^*\right\} \le$$

$$Pr\left\{w_0 + w(C_1) + ... + w(C_k) - kda_n > (1 + \varepsilon_n)w^*\right\} \le$$

$$Pr\left\{w(C_1) + ... + w(C_k) - kda_n > \varepsilon_n w^*\right\} \le$$

$$Pr\left\{w(C_1) + ... + w(C_k) - kda_n > \varepsilon_n kna_n\right\}.$$

The last probability equals to zero since the inequality inside is always false due to the definition of ε_n and Lemma 3 under the assumption that every C_i, $i = 1, \ldots, k$, was successfully found by the algorithm A_{AV}.

A probability that C_i was successfully found is $1 - \delta_{AV}$, it give us the following expression for the value of the failure probability of the algorithm A_k:

$$\delta_n = 1 - \prod_{i=1}^{k}(1 - \delta_{AV}) \le \sum_{i=1}^{k} \delta_{AV} = k\delta_{AV} = \mathcal{O}\left(k\left(\frac{1}{d^\alpha} + \frac{1}{d}\right)\right).$$

Theorem 3. *The algorithm A_k is asymptotically optimal if $d \to \infty$ as $n \to \infty$, $d = o(n)$ and*

$$b_n/a_n = o(n/\log n).$$

Proof. It follows from the Theorem 2.

2 Conclusion

The problem of finding k minimum spanning trees with diameter bounded below was studied. The polynomial algorithm was presented, its performance guarantees and sufficient conditions of being asymptotically optimal were found for the case where edges weights have independent uniform distribution on a segment $[a_n, b_n]$, $b_n > a_n > 0$. Further analysis of this problem on graphs with other distributions of edges weights, for example a case when $a_n = 0$, is of a big interest.

Acknowledgments. This research was supported by the Russian Foundation for Basic Research (grants 15-01-00976 and 16-31-00389).

References

1. Angluin D.,Valiant L.G.: Fast probabilistic algorithms for Hamiltonian circuits and matchings. Journal of Computer and System Sciences, Vol 18(2), 155–193 (1979)
2. Erdos P., Renyi A.: On random graphs I, Publ. Math. Debrecen 6, 290–297, (1959)
3. Petrov V.V.: Limit Theorems of Probability Theory. Sequences of Independent Random Variables. Clarendon Press, Oxford, 304 p., (1995)
4. Roskind J. and Tarjan R. E.: note on finding minimum-cost edge-disjoint spanning trees. Math. Oper. Res., 10 (1985), 4, 701-708
5. Gimadi E.Kh., Glebov N.I., and Perepelitsa V.A.: Algorithms with Estimates for Discrete Optimization Problems. Probl. Kibern., (1975), no. 31, pp. 35-42 (in Russian)
6. Gimadi E. Kh., Serdyukov A. I.: An algorithm for finding the minimum spanning tree with a diameter bounded from below. Diskretn. Anal. Issled. Oper., Ser. 1, 7(2) (2000), 3-11 (in Russian)
7. Gimadi E. Kh., Shin E. Yu.: Probabilistic analysis of an algorithm for the minimum spanning tree problem with diameter bounded below. Journal of Applied and Industrial Mathematics (October 2015), Volume 9, Issue 4, 480–488.