Nearly Optimal Solution for Restricted Euclidean Bottleneck Steiner Tree Problem

Zimao Li* and Wenying Xiao
College of Computer Science, South-Central University for Nationalities, Wuhan City, Hubei Province, P. R. China
*Corresponding author, Email: lizm@mail.scuec.edu.cn, 1047464075@qq.com

***Abstract*—A variation of the traditional Steiner tree problem, the bottleneck Steiner tree problem is considered in this paper, which asks to find a Steiner tree for *n* terminals with at most *k* Steiner points such that the length of the longest edge in the tree is minimized. The problem has applications in the design of WDM optical networks, design of wireless communication networks and reconstruction of phylogenetic tree in biology. We study a restricted version of the bottleneck Steiner tree problem in the Euclidean plane which requires that only degree-2 Steiner points are possibly adjacent in the optimal solution. The problem is known to be MAX-SNP hard and cannot be approximated within $\sqrt{2}$ unless P=NP, we propose a nearly optimal randomized polynomial time approximation algorithm with performance ratio $\sqrt{2}+\varepsilon$, where ε is a positive number.**

***Index Terms*—Bottleneck Steiner Tree; Approximation Algorithm; Performance Ratio; Wireless Networks**

I. Introduction

Given a weighted graph $G=(V,E;W)$ and a subset $S\subset V$ of required vertices (also called terminals), the traditional *Steiner tree* problem [1] asks a shortest acyclic network connecting *S*. In fact, the acyclic network is a tree and it may use additional points (also called Steiner points) in *V-S*. We call such a tree a Steiner tree. In the past 20 years, the traditional Steiner tree problem attracts considerable attention and interests from both theoretical point of view and its applicability and once occupied a central place in the emerging theory of approximation algorithms.

The problem is MAX-SNP hard even when the edge weights are only 1 or 2 [2]. For the Steiner tree problem in Euclidean plane, it is still NP-hard and there is a polynomial-time approximation scheme (PTAS) [3] for Euclidean Steiner trees, i.e., a near-optimal solution can be found in polynomial time [4].

New applications of Steiner tree problem in VLSI routing [5], wireless communications [6] and phylogenetic tree reconstruction in biology [7] have been found and studied deeply. These applications generally need to do some modification to the traditional Steiner tree problem. Therefore, the study of variations of traditional Steiner tree problem become a hot issue.

For example, recent advances in affordable and efficient electronics have had a dramatic impact on the availability and performance of radio-frequency wireless communication equipment. A number of defense and civil applications involve deployment of computing devices or sensors able to communicate digital information through wireless connections. In most cases the sensors are battery powered and therefore operate for a limited time before they consume all power and stop working. In order to prolong the network lifetime in general, it is desirable to minimize the distance between nodes [8].

Another example, in the design of wavelength division multiplexing (WDM) optical network, suppose we need to connect the *n* nodes located at p_1, p_2,..., p_n by WDM optical network, due to transmit power limit, signal can only transmit a limited distance to ensure correct transmission. If the distance between some nodes in the connection tree is large, signal amplifiers are required to place at proper positions to shorten the connection distance.

Two examples leads us to consider minimizing the maximum edge length problem and minimizing the number of Steiner points problem, implying the two variants of the classic Steiner tree problem: the bottleneck Steiner tree problem [9] and the Steiner tree problem with minimum number of Steiner points and bounded edge-length [8, 10, 11, 13].

In this paper, we consider one related variation of the traditional Steiner tree problem, the bottleneck Steiner tree problem, which is defined as follows: given a set $P=\{p_1, p_2, ..., p_n\}$ of *n* terminals and a positive integer *k*, we want to find a Steiner tree with at most *k* Steiner points such that the length of the longest edges in the tree is minimized.

The problem can be applied to extend the lifetime of a wireless network when *n* nodes have fixed locations and a number of up to *k* additional nodes can be placed at arbitrary positions. The objective is to build a spanning tree that connects the *n* fixed points and up to *k* additional nodes in the Euclidean plane, so that the length of the longest tree edge is minimized. Hence, the power required to transmit on the longest link is minimized also, and the network lifetime, in terms of connectivity, is extended.

Other applications such as design of multifacility location, VLSI routing, network routing, optical switching networks and phylogenetic tree reconstruction indicates the broad applicability of the bottleneck Steiner tree problem.

© 2014 ACADEMY PUBLISHER
doi:10.4304/jnw.9.4.1000-1004

The problem is showed to be NP-hard. In [9], D.-Z Du and L. Wang proved that unless P=NP, the problem cannot be approximated in polynomial time within performance ratios 2 and $\sqrt{2}$ in the rectilinear plane and the Euclidean plane, respectively. Moreover, they gave an approximation algorithm with performance ratio 2 for both the rectilinear plane and the Euclidean plane. For the rectilinear plane, the performance ratio is best possible, that is, the performance ratio is tight. For the Euclidean plane, however, the gap between the lower bound $\sqrt{2}$ and upper bound 2 is still big. Based on the existence of a 3-restricted Steiner tree, we presented a randomized polynomial approximation algorithm with performance ratio 1.866 + ε, for any positive number ε for the Euclidean plane [12]. Later I. Cardei, M. Cardei, L. Wang, B. Xu, and D.-Z Du improved the performance ratio to $\sqrt{3}$+ε, for any positive number ε [8, 13]. This is so far the best results possible.

In 2004, a restricted version of the problem in the Euclidean plane which requires that no edge connects any two Steiner points in the optimal solution was considered. We proved that the problem is NP-hard and cannot be approximated in polynomial time within performance ratio $\sqrt{2}$ and proposed a randomized polynomial approximation algorithm with performance ratio $\sqrt{2}+$ ε, for any positive number ε [14]. S. Bae, C. Lee, and S. Choi studied the Euclidean bottleneck Steiner tree problem when k is restricted to 1 or 2, they gave exact solutions to this problem [15]. M. Li, B. Ma and L. Wang studied the bottleneck Steiner tree problem in String space when k = 1 (also called the closest string problem). They proved the problem to be NP-hard and present a PTAS for it, and hence solved it perfectly in theory [16].

In this paper, we study the bottleneck Steiner tree problem in the Euclidean plane by allowing only degree-2 Steiner points are possibly adjacent in the optimal bottleneck Steiner tree. The case we consider is more general than the restricted version in [14]. We denote the problem *restricted-BST* for short. We have shown that the problem is MAX-SNP hard an cannot be approximated within performance ratio $\sqrt{2}$ and provide an O($n$log$n$+$k$log$n$), approximation algorithm with performance ratio $\sqrt{3}$ [17]. But there still exist a gap between the lower bound $\sqrt{2}$ and upper bound $\sqrt{3}$. In this paper, by introducing the notion of 3-restricted Steiner tree, we prove the existence of ratio $\sqrt{2}$ and propose a randomized polynomial time approximation algorithm with performance ratio $\sqrt{2}$+ε, for any positive number ε, which is nearly optimal and almost close the problem.

In Section II, we show that the existence of 3-restricted Steiner tree with the length of the longest edge not exceeding $\sqrt{2}$ of the optimal solution. Section III provide a method to construct a weighted 3-hyoergraph and a polynomial time randomized approximation algorithm with performance ratio $\sqrt{2}$ +ε. By introducing the binary search strategy, we also give a fast implementation. The concluding remarks appear in Section IV.

II. The Existence of Performance Ratio

In this section, by introducing the notion of 3-restricted Steiner tree, we will show the existence of performance ratio $\sqrt{2}$ for the restricted-BST problem. First, the following theorem in [17] shows the hardness of the problem.

Theorem 1: Unless P=NP, the restricted-BST problem in the Euclidean plane cannot be approximated within performance ratio $\sqrt{2}$ in polynomial time.

For describing convenience, we need some related notions. Usually, every leaf in a Steiner tree is a terminal. However, a terminal may not be a leaf. A Steiner tree is *full* if all terminals are leaves. Thus, if a Steiner tree is not full, there must exist a terminal which is not a leaf, we can decompose the tree at this terminal into several small trees, and these small trees share a common terminal. In this way we can always decompose any Steiner tree into the union of several small trees, in each of them a vertex is a leaf if and only if it is a terminal. These small trees are called *full Steiner components*, or formally,

Definition 1: A full Steiner component of a Steiner tree is a subtree in which each terminal is a leaf and each internal node is a Steiner point.

Consequently, We can define the notion of *k-restricted Steiner tree* and in this paper we focus on the case when k = 3.

Definition 2: A Steiner tree for n terminals is a k-restricted Steiner tree if each of its full component spans at most k terminals.

Below notation is adopted in the proof of our main theorem-Theorem 2. Let a and b be two points in the plane, we denote ab an edge and $|ab|$ the length of ab. Without loss of generality, we assume the length of the longest edges in the optimal Steiner tree is 1.

Theorem 2: Given a set of n terminals P in the Euclidean plane, let T be an optimal bottleneck Steiner tree for the restricted-BST problem. Then, there exists a 3-restricted Steiner tree T' for P with the same number of Steiner points as T such that the length of the longest edges in T' is at most $\sqrt{2}$.

Proof: Because only degree-2 Steiner points are possibly adjacent in the optimal solution, any optimal bottleneck Steiner tree T can be decomposed into the union of its full components, each of which is either a star with a Steiner point as center (see Figure 1) or just a line-segment path connecting two terminals with $l \geq 0$ intermediate degree-2 Steiner points (See Figure 2).

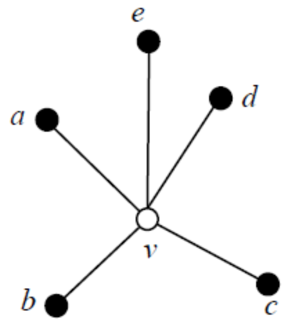

Figure 1. A star with a Steiner point

© 2014 ACADEMY PUBLISHER

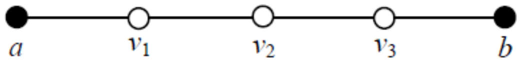

Figure 2. A line-segment path with 3 degree-2 Steiner points

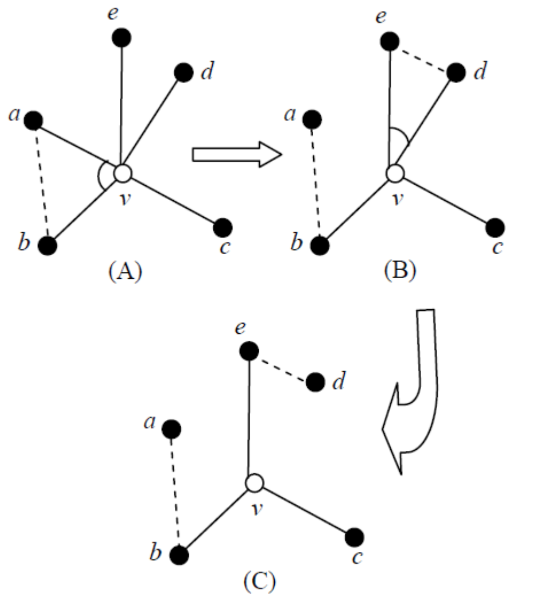

Figure 3. Transformation of a star to 3-restricted Steiner subtree

For a star T_s with at least 4 terminals, we can always decrease the degree of the Steiner point step by step to 3 and guarantee the length of the longest edges in the modified tree is at most $\sqrt{2}$. The procedure is as below:

Suppose the Steiner point is labeled as v, there must exist two terminals a and b satisfying $\angle avb \le 90°$, by directly connecting a and b and removing the longer edge of va and vb, the degree of v is decreased by 1, and it is easily seen that

$$|ab| = \sqrt{|va|^2 + |vb|^2 - 2|va| \cdot |vb| \cos\angle avb} \le \sqrt{2}$$

(remember the assumption that the length of the longest edges in the optimal Steiner tree is 1). Repeat the procedure until the degree of v becomes 3. Figure 3 gives an example to illustrate the procedure.

Thus we transform the star T_s into a Steiner subtree in which the length of the longest edges is at most $\sqrt{2}$ and the number of Steiner points in the Steiner subtree does not increase. For the line-segment path like full Steiner component, no transformation work is need because the length of its edges is at most 1. Finally we union all the Steiner subtrees to form a steinerized spanning tree T' with the same number of Steiner points as the optimal bottleneck Steiner tree T, apparently T' is a 3-restricted Steiner tree and the length of the longest edges in T' is at most $\sqrt{2}$.

A *hypergraph* $H=(V, F)$ is a generalization of a graph where the edge set F is an arbitrary family of subsets of vertex set V. A 3-*hypergraph* $H_3=(V, F)$ is a hypergraph, each of whose edges has cardinality at most 3. A *weighted* 3-*hypergraph* $H_3=(V, F; W)$ is a 3-hyperpgrah with each edge associated with a weight. A minimum spanning tree for a weighted 3-hypergraph $H_3=(V, F; W)$ is a subgraph T of H_3 that is a tree containing every node in V with the least weight.

The following theorem proves the existence of a randomized algorithm for computing a minimum spanning tree for a weighted 3-hypergraph [18].

Theorem 2: There exists a randomized algorithm for the minimum spanning tree problem for weighted 3-hypergraphs, with probability at least 0.5, running in poly(n,w_{max}) time, where n is the number of nodes in the hypergraph and w_{max} is the largest weight of edges in the hypergraph.

III. The Approximation Algorithm

In this section, we transform the computation of an optimal 3-restricted Steiner tree into the minimum spanning tree problem for weighted 3-hypergraphs.

To construct a weighted 3-hypergraph, we need to know B, the length of the longest edges in an optimal solution. It is hard to find the exact value of B in an efficient way because of the hardness of the restricted-BST problem. However, we can guess the length of the longest edges in an optimal solution. The following procedure finds a value B' that is at most $(1+\varepsilon)B$ for any $\varepsilon>0$:

Run the polynomial time approximation algorithm with performance ratio $\sqrt{3}$ in [17] to get an upper bound X of B.

Try to use one of $\frac{X}{\sqrt{3}}, \frac{X}{\sqrt{3}}(1+\varepsilon), \frac{X}{\sqrt{3}}(1+2\varepsilon), \ldots$, X as B', where ε is a positive number.

Thus, we can assume that $B'=(1+\varepsilon)B$ is the approximation of the longest edges in an optimal solution. Now we can construct a weighted 3-hypergraph $H_3=(V, F; W)$ from the set P of terminals. The construction process is shown in Figure 4.

3-hypergraph $H_3=(V, F; W)$

- Vertex set $V = P$;
- Edge set $F = \{(a, b) | a \in P \text{ and } b \in P\} \cup \{(a, b, c) | a \in P, b \in P \text{ and } c \in P\}$;
- The weight for an edge $f=(a, b) \in F$ is the smallest number of Steiner points that should be added to the edge such that the length of each resulting edge is at most $\sqrt{2}B'$. If an edge length is at most $\sqrt{2}B'$, then its weight is 0. Generally, $w(f) = \left\lceil \frac{|ab|}{\sqrt{2}B'} \right\rceil - 1$;
- The weight for an edge $f=(a, b, c) \in F$ is defined as $w(f) = 1$ if there is a Steiner point v connecting 3 a, b and c such that $|va| \le B'$, $|vb| \le B'$ and $|vc| \le B'$, otherwise $w(f) = \infty$ and such an edge is omit when we compute the minimum spanning tree for H_3;
  - ➢ The position of v can be considered as the circumcentre of triangle Δabc if abc forms a triangle, or the center of the longest segment among ab, ac and bc if a, b and c are collinear.

Figure 4. Construction process of a 3-hypergraph

It is easily seen that for a given bound B', the above construction can be done in $O(n^3)$ time because the number of edges is $O(n^3)$ and the computation of the weight of an edge uses only constant time. After obtaining the weighted 3-hypergraph $H_3(V, F; W)$, we can

© 2014 ACADEMY PUBLISHER

use the algorithm in [18] to computer a minimum spanning for H_3. Now it is ready for us to present our approximation algorithm with performance ratio $\sqrt{2}+\varepsilon$, where ε is a positive number.

Algorithm restricted-BST(P, n, k, ε)

Input: A set P of n terminals in the Euclidean plane, an integer k and a positive number ε.

Output: A 3-restricted Steiner tree T for P with at most k Steiner points.

Call the O($n$log$n$+$k$log$n$) approximation algorithm with performance ratio $\sqrt{3}$ for restricted Euclidean bottleneck Steiner tree problem in [17] to get the length of the longest edge X.

For $B' \leftarrow \frac{X}{\sqrt{3}}, \frac{X}{\sqrt{3}}(1+\varepsilon), \frac{X}{\sqrt{3}}(1+2\varepsilon), \ldots, \frac{X}{\sqrt{3}}(1+\left\lceil\frac{\sqrt{3}-1}{\varepsilon}\right\rceil \times \varepsilon)$ **do**

Construct a weighted 3-hypergraph $H_3(V,F;\ W)$ according to B' and Figure 4.

Call the polynomial randomized algorithm in [18] to compute a minimum spanning tree T' for $H_3(V, F;\ W)$.

if $w(T') \le k$ **then** exit the for loop.

Replace every edge f of the minimum spanning tree T' on $H_3(V, F;\ W)$ with a Steiner subtree as below descriptions.

If $f = (a, b)$, replace f with a path connecting a and b by adding $w(f)$ intermediate Steiner points with a even partition of f.

If $f = (a, b, c)$, replace f with a star centered at the circumcenter triangle Δabc if abc forms a triangle, or at the center of the longest segment among ab, ac and bc if a, b and c are collinear.

Output the resulting 3-restricted Steiner tree.

Theorem 1 and Theorem 2 indicates the existence and performance of a randomized approximation algorithm for the restricted Euclidean bottleneck Steiner tree problem. Combined with algorithm restricted-BST, we have the following Theorem.

Theorem 3: For any given ε, there exists a randomized algorithm that computes a Steiner tree with n terminals and k Steiner points with probability at least 0.5 such that the longest edge in the tree is at most $\sqrt{2}+\varepsilon$ times of the optimum, and the algorithm's running is $\frac{1}{\varepsilon} \times poly(n,k)$.

In fact, by using a binary search strategy, we can decrease the number of loops in Step 2 from $\frac{1}{\varepsilon}$ to $\log(\frac{1}{\varepsilon})$ and hence improve Algorithm restricted-BST(P, n, k, ε). Algorithm faster-restricted-BSP(P, n, k, ε) is an improvement of Algorithm restricted-BST(P, n, k, ε).

Algorithm faster-restricted-BST(P, n, k, ε)

Input: A set P of n terminals in the Euclidean plane, an integer k and a positive number ε.

Output: A 3-restricted Steiner tree T for P with at most k Steiner points.

Call the O($n$log$n$+$k$log$n$) approximation algorithm with performance ratio $\sqrt{3}$ for restricted Euclidean bottleneck Steiner tree problem in [17] to get the length of the longest edge X.

Initialize $low \leftarrow 0$ and $high \leftarrow \left\lceil\frac{\sqrt{3}-1}{\varepsilon}\right\rceil$

while ($low < high$) **do**

$mid \leftarrow (low+high)/2$ and $B' \leftarrow \frac{X}{\sqrt{3}}(1+mid\times\varepsilon)$

Construct a weighted 3-hypergraph $H_3(V,F;\ W)$ according to B' and Figure 4.

Call the polynomial randomized algorithm in [18] to compute a minimum spanning tree T for $H_3(V, F;\ W)$.

Consider the solution T', **if** $w(T') > k$, **then** $low \leftarrow mid+1$; **else** $high \leftarrow mid$.

Replace every edge f of the minimum spanning tree T' on $H_3(V, F;\ W)$ with a Steiner subtree as below descriptions.

If $f = (a, b)$, replace f with a path connecting a and b by adding $w(f)$ intermediate Steiner points with a even partition of f.

If $f = (a, b, c)$, replace f with a star centered at the circumcenter triangle Δabc if abc forms a triangle, or at the center of the longest segment among ab, ac and bc if a, b and c are collinear.

Output the resulting 3-restricted Steiner tree.

IV. Conclusion

We mainly considered a restricted version of the bottleneck Steiner tree problem in the Euclidean plane. The problem is MSX-SNP hard and cannot be approximated with ratio $\sqrt{2}$ unless P=NP. In this paper we presented a polynomial time randomized approximation algorithm with performance ratio $\sqrt{2}+\varepsilon$. The algorithm is near optimal and almost close the gap between lower bound $\sqrt{2}$ and upper bound $\sqrt{2}+\varepsilon$.

Further study include the derandomization of the randomized algorithm efficiently.

As an application, the algorithm can be used to improve the lifetime of wireless networks by minimizing the length of the longest edge in the interconnecting tree by deploying additional relay nodes at specific locations.

Acknowledgment

The author wish to thank Rongbo Zhu for his double-check and suggestions on this paper. This work was fully supported by National Natural Science Foundation of China (Project 61103248 and 61379059).

References

[1] M. R. Garey, R. L. Graham and D. S. Johnson, "The Complexity of Computing Steiner Minimal Trees," *SIAM Journal on Applied Mathematics*, vol. 32, pp. 835-859, 1977.

[2] M. Bern and P. Plassmann, "The Steiner Problem with Edge Lengths 1 and 2," *Information Processing Letters*, vol. 32, pp. 171-176, 1989.

[3] M. R. Garey and D. S. Johnson, "Computers and Intractability, A Guide to the Theory of NP-Completeness," W. H. Freeman and Company, New York, 1979.

[4] S. Arora, "Polynomial Time Approximation Scheme for Euclidean TSP and Other Geometric Problems," *Proceedings of the 37th Annual Symposium on Foundations of Computer Science*, Burlington VT, pp. 2-11, Oct. 1996.

[5] A. Kahng and G. Robins, "On Optimal Interconnections for VLSI," Kluwer Publishers, 1995.

[6] A. Caldwell, A. Kahng, S. Mantik, I. Markov and A. Zelikovsky, "On Wirelength Estimations for Row-based Placement," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol 18, pp. 1265-1278, 1999.

[7] F. K. Hwang, D. S. Richards and P. Winter, "The Steiner Tree Problem," North-Holland, 1992.

[8] I. Cardei, M. Cardei, L. Wang, B. Xu, D. -Z. Du, "Optimal relay location for resource-limited energy-efficient wireless communication," *Journal of Global Optimization*, vol. 36, pp. 391–399, 2006.

[9] L. Wang and D. -Z Du, "Approximations for a Bottleneck Steiner Tree Problem," *Algorithmica*, vol. 32, pp. 554-561, 2002.

[10] M. Sarrafzadeh and C. Wong, "Bottleneck Steiner Trees in the Plane," *IEEE Transactions on Computers*, vol. 41, pp. 370-374, 1992.

[11] G. Lin and G. Xue, "Steiner Tree Problem with Minimal Number of Steiner Points and Bounded Edge-length," *Information Processing Letters*, vol. 69, pp. 53-57, 1999.

© 2014 ACADEMY PUBLISHER

[12] L. Wang and Z. Li, "An approximation algorithm for a bottleneck *k*-Steiner tree problem in the Euclidean plane", *Information Processing Letters*, vol. 81, pp. 151-156, 2002.
[13] D. -Z Du, L. Wang and B. Xu, "The Euclidean Bottleneck Steiner Tree and Steiner Tree with Minimum Number of Steiner Points," in *Proceedings of the 7th Annual International Conference on Computing and Combinatorics*, Guilin, China, LNCS vol. 2108, pp. 509-518, August 2001.
[14] Z. Li, D. Zhu and S. Ma, "Approximation algorithm for bottleneck Steiner tree problem in the Euclidean plane," *Journal of Computer Science and Technology*, vol. 19, pp. 791-794, 2004.
[15] S. Bae, C. Lee, and S. Choi, "On Exact Solutions to the Euclidean Bottleneck Steiner Tree Problem," in *Proceedings of the Third Annual Workshop on Algorithms and Computation*, Kolkata, India, LNCS vol. 5431, pp. 105-116, Feb. 2009.
[16] M. Li, B. Ma and L. Wang, "On the Closest String and Substring Problems," *Journal of the ACM*, vol. 49, pp. 157-171, 2002.
[17] Z. Li and W. Xiao, "Fast Approximation Algorithm for Restricted Euclidean Bottleneck Steiner Tree Problem," unpublished.
[18] H. J. Prömel and A Steger, "A new approximation algorithm for the steiner tree problem with performance ratio5/3," *Journal of Algorithms*. vol. 36, pp. 89–101, 2000.

Zimao Li, Ph.D., associate professor and vice dean, born in Linqing, P.R, China, 1974, received his Bachelor's degree in Mathematics, Master's degree in Computer Science from Shandong University in 1996 and 1999, respectively, and his Ph.D. degree in Computer Science from City University of Hong Kong in 2002. His research interests include computational complexity, approximation algorithms and design and analysis of algorithms.

From 2002 to 2005, he was a lecturer of College of Computer Science and Technology, Shandong University, P.R. China; from 2005, he is an associate professor of College of Computer Science, South-Central University for Nationalities, P.R. China; from 2012, he is the vice dean of College of Computer Science, South-Central University for Nationalities. He has been supported by 1 national fund and 3 provincial level funds and published more than 10 research papers in journals such as SIAM Journal on Computing, IEEE Transactions on SMC, Journal of Computer Science and Technology, Information Processing Letters, etc.

Wenying, Xiao, born in Anguo, P.R. China, 1988, received his Bachelor's degree in Computer Science and Technology from South-Central University for Nationalities in 2012, currently he is a Master degree candidate. His research interest is design and analysis of algorithms.

© 2014 ACADEMY PUBLISHER