

## Short Paper

---

# An Efficient Relay Sensor Placing Algorithm for Connectivity in Wireless Sensor Networks

JYH-HUEI CHANG

*Department of Computer Science  
National Chiao Tung University  
Hsinchu, 300 Taiwan*

Randomly deployed sensor networks often make initial communication gaps inside the deployed area, even in an extremely high-density network. How to add relay sensors such that the underlying graph is connected and the number of relay sensors added is minimized is an important problem in wireless sensor networks. This paper presents an Efficient Relay Sensor Placing Algorithm (ERSPA) for solving such a problem. Compared with the minimum spanning tree algorithm and the greedy algorithm, ERSPA achieves better performance in terms of the number of relay sensors added. Simulation results show that the average number of relay sensors added by the minimal spanning tree algorithm is approximately two times that of the ERSPA algorithm.

**Keywords:** relay sensor, connectivity, algorithm, sensor networks, Delaunay

## 1. INTRODUCTION

Randomly deployed networks often make initial communication gaps inside the deployed area, even in an extremely high-density network. In the random sensor network topology, the sensors may be sparsely located and connectivity is not guaranteed. Therefore, it is an important topic for researchers to find an efficient algorithm for improving connectivity in wireless sensor networks, while minimizing the number of relay sensors added.

In a finite domain, the connectivity of a random network depends only on the probability distribution of the critical transmission range. Many studies have tried to find efficient algorithms for determining the critical transmission range for connectivity [1-3]. The asymptotic distribution of the critical transmission radius for  $k$ -connectivity is derived in [1]. This study proved that the critical transmission range in the unit-area square is  $r_n = \sqrt{\frac{\log n + (2k-1)\log \log n + \xi}{\pi n}}$  where  $n$  is the number of network nodes and  $\xi$  is a constant. The critical transmitting range for connectivity in mobile ad hoc networks is proved in [2]. The author showed the critical transmission range (CTR) for a mobility model  $M$  is  $r_M = c\sqrt{\frac{\ln n}{\pi n}}$  for some constants  $c \geq 1$  where  $n$  is the number of nodes in the network. The mobility model  $M$  is assumed to be obstacle free, and nodes are allowed to move only

---

Received December 25, 2008; revised April 7, June 23 & August 13, 2009; accepted August 20, 2009.  
Communicated by Tsan-sheng Hsu.

within a certain bounded area. In addition, many studies have focused on maintaining sensing coverage and connectivity in wireless sensor networks [4-7]. The transmission range  $R_t$  being at least twice that of the sensing range  $R_s$  is the sufficient condition to ensure that complete coverage preservation implies connectivity among active nodes [4]. Another study [5] enhanced the work in [4] to prove that the sufficient condition for complete coverage implies that connectivity is  $R_t = 2R_s$ . In [7], a coverage configuration protocol is proposed to achieve guaranteed degrees of coverage and connectivity. This work provided different degrees of coverage requested by applications. To measure the coverage, the work divides the sensing area into  $1m \times 1m$  patches. The coverage degree of a patch is approximated by measuring the number of active nodes that cover the center of the patch.

Note that the above studies [1-7] do not discuss how to place relay sensors to improve connectivity for a disconnected ad hoc network. Thus, there are many papers proposed for finding the optimal location to place the additional nodes to achieve network connectivity [8-12]. This can be reduced to a minimal Steiner tree problem. In [8], a relay sensor placement algorithm to maintain connectivity is proposed. The authors formulated this problem into a network optimization problem, named the Steiner Minimum Tree with Minimum Number of Steiner Points (SMT-MSP). This study restricts the transmission power of each sensor to a small value and adds relay sensors to guarantee connectivity. Simulation results show that their method can achieve better performance in terms of total consumed power and maximum degree, especially for sparse network topology. However, their algorithm has a time complexity in  $O(N^3)$ . Some heuristic algorithms for the bounded edge-length Steiner tree problem with a good approximate ratio are proposed in [9-12]. Nevertheless, these heuristic algorithms do not consider the heterogeneous transmission ranges of terminal nodes and relay nodes.

Many studies have focused on finding efficient heuristic algorithms to solve the minimal additional nodes placing problem and to prolong network lifetime [13-15]. A heuristic algorithm for the energy preserving problem is proposed in [13]. This algorithm transforms the mixed-integer nonlinear problem into a linear programming problem. This study provides additional energy on the existing nodes and deploys relay nodes into the network to prolong network lifetime. In [14], three heuristic algorithms are proposed for achieving the connectivity of a randomly deployed ad hoc wireless network. This work connects the network with a minimum number of additional nodes and maximizes utility from a given number of additional nodes for the disconnected network. The time complexity of the greedy algorithms is  $O(N^2)$  in a two dimensional space, where  $N$  is the number of terminal nodes.

Our motivation is to find an efficient relay sensor placing algorithm to construct a connected communication graph for connectivity, and to minimize the total number of relay sensors. We assume that all terminal and relay sensors have the same transmission range and all sensors are location aware. The simulation results show that the ERSPA algorithm gives better performance in terms of the average number of relay sensors with respect to the minimal spanning tree algorithm and the greedy algorithm [15]. The average number of relay sensors in the minimum spanning tree algorithm is approximately two times that of the ERSPA algorithm. This is because the ERSPA places relay sensors in optimal locations to connect the maximal number of initial connected sub-graphs.

The remainder of this paper is organized as follows. In section 2, we describe the

problem formulation and the network model. Section 3 illustrates the details of the ERSPA algorithm. The simulation results and performance analysis are shown in section 4. Finally, the conclusions are given in section 5.

## 2. PROBLEM FORMULATION AND NETWORK MODEL

Consider a wireless sensor network. We assume that the sensing area is in a two dimensional space which is a bounded convex subset  $R$  of the Euclidean space. In this sensing area, the initially deployed sensors, referred to as terminal nodes, have been placed and a set of relay sensors is available to be added for connectivity. All terminal nodes and relay nodes are location aware so that the location information can be collected. The set of the terminal nodes is denoted as  $N_t = \{N_{t1}, N_{t2}, \dots, N_{tm}\}$ . The transmission range of each terminal node is adjustable. Initially, the terminal nodes can adjust their transmission range to convey their location information to the base station, then limit their transmission range in a bounded value  $R_t$  for energy efficiency. The set of locations of  $n$  terminal nodes is denoted as  $L_t = \{p_i \in R \mid i = 1, \dots, n\}$ . The set of the initial network topology is in the form of an undirected graph denoted as  $G(N_t, E(L_t, R_t))$  where  $E(L_t, R_t) = \{(l_i, l_j) \mid l_i, l_j \in L_t, i \neq j, \|l_i - l_j\| \leq R_t\}$ . In order to construct the connected communication graph, we can add the relay sensors to connect the initial separated sub-graphs. A solution is a set of locations to place relay sensors,  $L_r = \{q_i \in R \mid i = 1, \dots, r\}$ . The set of relay sensors is denoted as  $N_r = \{N_{r1}, N_{r2}, \dots, N_{rm}\}$ . We formulate our problem as follows: A randomly deployed sensor network with an  $nR_t \times nR_t$  sensing area in two dimensional space. Given  $N_t$  and  $R_t$ , find the  $L_r$  for minimum relay sensor set  $N_r$  to make the graph  $G(N_t \cup N_r, E(L_t \cup L_r, R_t))$  connected.

## 3. DETAILS OF THE ERSQA ALGORITHM

The ERSQA algorithm includes the following two phases: (1) Construct Delaunay and find the initial graph  $G(N_t, E(L_t, R_t))$ ; (2) Add relay nodes. The details of the algorithm are given as follows.

**Phase 1:** Construct Delaunay and find the initial graph  $G(N_t, E(L_t, R_t))$

Construct the Delaunay by using terminal nodes [16]. The construction of Delaunay is illustrated as follows. Let  $S$  be a set of points in a two dimensional space. The Voronoi diagram of  $S$ , denoted as  $Vol(S)$  which is decomposed into Voronoi cells  $\{V_a: a \in S\}$  is defined as Eq. (1).

$$V_a = \{x \in R^2: |x - a| \leq |x - b| \forall b \in S\} \quad (1)$$

The dual of the Voronoi diagram is the Delaunay triangulation  $Del(S)$ .  $Del(S)$  is geometrically realized as a triangulation of the convex hull of  $S$  (see Fig. 1). After constructing Delaunay, we calculate the length of the three edges of each triangle. If the length of the edge is no more than transmission range  $R_t$ , then connect it. As shown in Fig. 1, the initial connected sub-graphs in Phase 1 are indicated by the thick solid line.

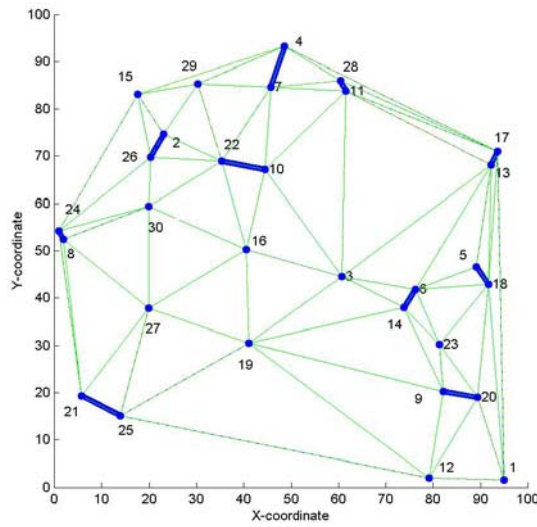


Fig. 1. An example of constructing Delaunay using 30 terminal nodes. The thin solid lines represent the edges of the Delaunay triangulation that are not connected. The initial disconnected terminal nodes are indicated by the ‘•’-sign. The initial connected sub-graphs are indicated by the thick solid lines.

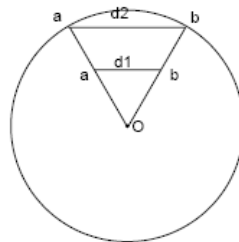


Fig. 2. An example of Lemma 1.

**Phase 2:** Add relay nodes

We first illustrate the following lemmas about adding relay nodes.

**Lemma 1** One relay node can connect at most five nodes that belong to different sub-graphs [11].

*Proof:* We prove this lemma by contradiction. As shown in Fig. 2, if six nodes can be covered by a minimum circle of radius  $R$  where  $R$  is no more than the transmission range  $R_t$ , we draw a circle of radius  $R_t$  to cover the six nodes centered at  $o$ , then draw six lines from each node to the center of the circle. Assume that the shortest length of the adjacent nodes pair  $(a, b)$  is  $d_1$ . We extend the length of line  $(a, b)$  from  $d_1$  to  $d_2$ . That is, nodes  $a$  and  $b$  are on the circle. If  $d_2$  is equal to  $R_t$ , each edge of triangle  $(a, o, b)$  has the same length. Angle  $(a, o, b)$  is equal to  $60^\circ$ . If  $d_2$  is larger than  $R_t$ , then the angle  $(a, o, b)$  is larger than  $60^\circ$ . The sum of all six angles is larger than  $360^\circ$ . It is a contradiction. Therefore, one relay node can connect at most five nodes that belong to different sub-graphs.

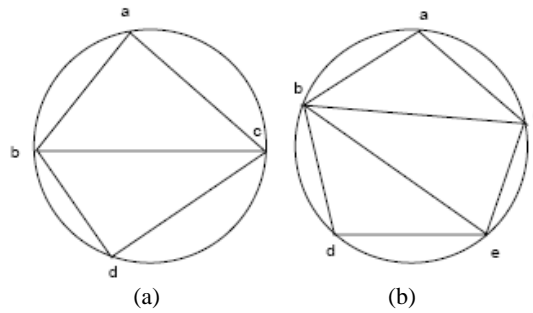


Fig. 3. (a) An example of four nodes covered by a minimum circle of radius  $R$ ; (b) An example of five nodes covered by a minimum circle of radius  $R$ .

**Lemma 2** Four nodes  $a, b, d, c$  can be covered by a minimum circle of radius  $R$  if and only if all the four triangles  $(a, b, d)$ ,  $(b, d, c)$ ,  $(d, c, a)$ ,  $(c, a, b)$  can be covered by a minimum circle of radius  $R$ .

**Proof:** As shown in Fig. 3 (a), triangles  $(a, b, c)$  and  $(b, d, c)$  form a quadrilateral  $(a, b, d, c)$ . From Lemma 8 in [11], four nodes  $a, b, d, c$  can be covered by a disc of radius  $R$  if all the four triangles  $(a, b, d)$ ,  $(b, d, c)$ ,  $(d, c, a)$  and  $(c, a, b)$  can be covered by a minimum circle of radius  $R$ .

**Lemma 3** Five nodes  $a, b, d, e, c$  can be covered by a minimum circle of radius  $R$  if and only if all the four quadrilaterals  $(a, b, d, e)$ ,  $(b, d, e, c)$ ,  $(d, e, c, a)$ ,  $(e, c, a, b)$  can be covered by a minimum circle of radius  $R$ .

**Proof:** As shown in Fig. 3 (b), each of the three adjacent triangles  $(c, a, b)$ ,  $(c, b, e)$  and  $(b, e, d)$  can be covered by a minimum circle of radius  $R$ . We prove this lemma by contradiction. If one of the four quadrilaterals  $(a, b, d, e)$ ,  $(b, d, e, c)$ ,  $(d, e, c, a)$  and  $(e, c, a, b)$  can not be covered by a minimum circle of radius  $R$ , say  $(e, c, a, b)$ . From Lemma 2, there exists at least one triangle of the four triangles  $(c, a, b)$ ,  $(a, b, e)$ ,  $(b, e, c)$ ,  $(e, c, a)$  that cannot be covered by a minimum circle of radius  $R$ . Assume that the triangle is  $(c, a, b)$ . This contradicts our assumption that triangle  $(c, a, b)$  can be covered by a minimum circle of radius  $R$ . Therefore, five nodes  $a, b, d, e,$  and  $c$  can be covered by a minimum circle of radius  $R$  if and only if all four quadrilaterals  $(a, b, d, e)$ ,  $(b, d, e, c)$ ,  $(d, e, c, a)$ ,  $(e, c, a, b)$  can be covered by a minimum circle of radius  $R$ .

After Phase 1, we divide the Delaunay triangles into three types. In Type 1, the length of all edges of the triangle is larger than  $R_i$  and smaller than  $2R_i$ . In Type 2, the longest edge of the triangle is at most  $4R_i$ , while the shortest edge is larger than  $R_i$  and at most  $2R_i$ . The properties of triangles different from Types 1 and 2 are defined as Type 3. For the triangles of Type 1, we place the relay nodes as per the following steps. First, we place one relay node to connect five nodes that are formed by three adjacent triangles. Second, we place one relay node to connect four nodes that are formed by two adjacent triangles. Third, we place one relay node to connect three nodes of one triangle. For the triangles of Type 2, we try to place two relay nodes to connect three nodes of one triangle. For the

triangles of Type 3, we add relay nodes to connect the nearest disconnected nodes pair along the edge of the triangle.

The details of adding relay nodes in the ERSPA algorithm are illustrated as follows.

- (1) Find the initial sub-graphs  $G(N_i, E(L_i, R_i))$ , the maximal connected sub-graph, and the Types 1 and 2 triangles.
- (2) Find the point where adding one relay node to connect the nodes in Type 1 triangles yields the maximal increase in the performance metric. The performance metric is the number of relay sensor. Add the relay node to  $N_r$ . Maintain the new sub-graphs, and the maximal connected sub-graph. Repeat this as long as new triangles can be connected.
- (3) Find the point to add one relay node to connect two different sub-graphs and add the relay node to  $N_r$ .
- (4) Find the point to add two relay nodes to connect three different sub-graphs and add the relay nodes to  $N_r$ .
- (5) If the graph  $G(N_i \cup N_r, E(L_i \cup L_r, R_i))$  is not yet connected, add relay nodes to connect the nearest disconnected nodes pair along the edge of the triangle.

In step 2 of Phase 2, two adjacent triangles can form a quadrilateral. For example, as shown in Fig. 1, triangle (2, 29, 15) and triangle (2, 29, 22) form a quadrilateral (2, 29, 15, 22). From Lemma 2, if the circumradius of each triangle is at most  $R_r$ , then we can place one relay node in the center of the minimal enclosing circle that covers the four points of the quadrilateral. We select the triangles that belong to the three different sub-graphs. The triangles inside the convex hull of the initial connected sub-graphs are not required to be searched. If the circumradius of the selected triangle is at most  $R_r$ , then add a node on the circumcenter of the triangle. The circumcenter is the intersection of the perpendicular bisectors of the sides of the triangle. The circumradius of a triangle is the radius of the circumscribed circle. The circumscribed circle of a triangle is the circle that passes through

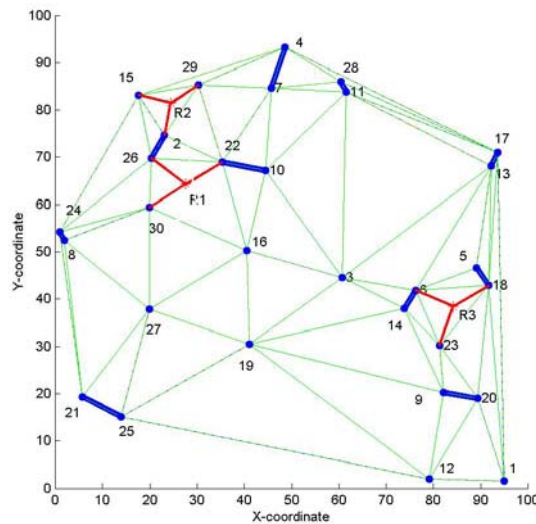


Fig. 4. An example of adding relay sensors after step 2 of Phase 2 for the ERSPA algorithm. Placing one relay node to connect three nodes is indicated by the circle centered at  $R_1$ ,  $R_2$  and  $R_3$ . The relay sensors are indicated by the ‘\*’-sign.

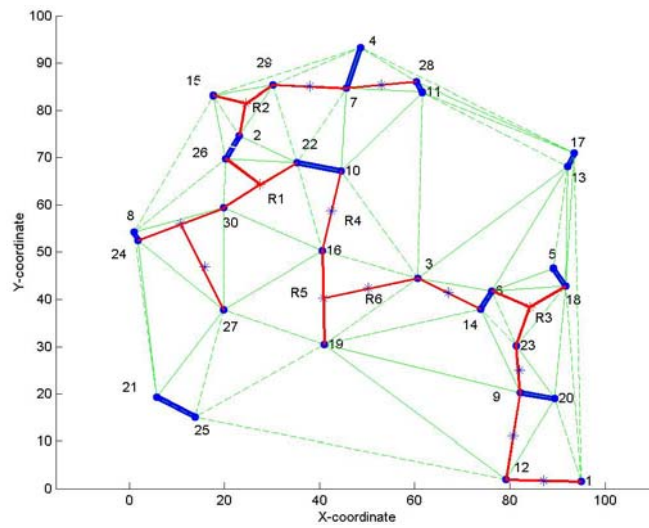


Fig. 5. The connected sub-graphs after steps 3 and 4 of Phase 2 in the ERSPA algorithm.

the three vertices of the triangle. For example, as shown in Fig. 4,  $R_1$  is the circumcenter of the triangle (30, 26, 22). If the distance from  $R_1$  to each apex of the triangle is not larger than transmission range  $R_t$ , then add a relay node  $R_1$  on the circumcenter. The relay sensors are indicated by the ‘\*’-sign. The ‘•’-sign represents the initial disconnected terminal nodes, and the initial connected sub-graphs are represented by the thick solid line. After adding a relay node, the edge becomes connected, which is indicated by a thin solid line. The performance metric of step 2 of ERSPA is the number of relay sensor. After step 2, we place one relay node to connect two sub-graphs. For example, as shown in Fig. 5, we place one relay node  $R_4$  to connect nodes 10 and 16. After step 3, we place two relay nodes to connect three different sub-graphs. For example, as shown in Fig. 5, we place one relay node  $R_6$  to connect nodes  $R_5$  and node 3. After step 4, if the graph  $G(N_t \cup N_r, E(L_t \cup L_r, R_t))$  is not yet connected, then place  $K$  relay nodes to connect two sub-graphs. Step 5 exists here to handle the case of Type 3 triangle.

The number of required relay sensors to add into the edge of the nearest pair of nodes is illustrated as Eq. (2).

$$K \times R_t < d(u, v) \leq (K + 1) \times R_t, K = 3, \dots, n \quad (2)$$

where  $K$  is the number of relay nodes.  $d(u, v)$  is the distance between node  $u$  and node  $v$ . For example, as shown in Fig. 4, we place two relay nodes to connect nodes 21 and 27. Repeat Phase 2 until the complete communication graph is connected (see Fig. 6).

In Phase 1, constructing Delaunay takes  $O(N \log N)$  times. Steps 1-2 of Phase 2 cost  $O(N)$  in plane and  $O(N^2)$  in three-dimensional sensing space. Steps 3-5 of Phase 2 cost  $O(N)$ . Therefore, Phase 2 requires  $O(N)$  times to add relay sensors in two-dimensional sensing space and  $O(N^2)$  in three-dimensional sensing space. The total time complexity of the ERSPA algorithm is  $O(N^2)$ . It is feasible in a two dimensional space.

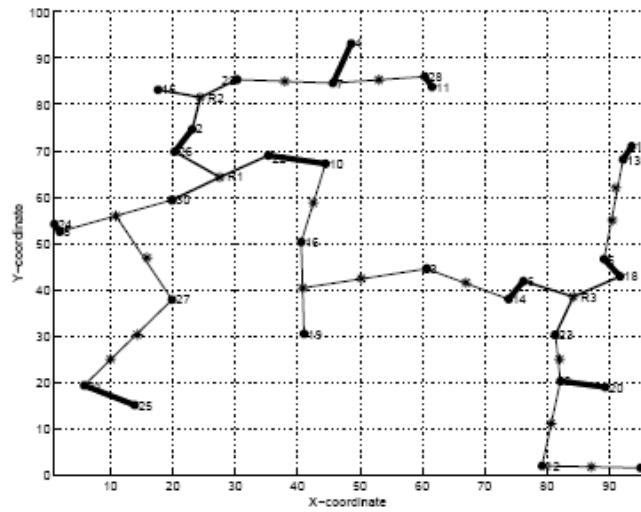


Fig. 6. The final connected communication graph. The transmission range is 10 percent of the side of the square sensing field.

## 4. SIMULATION RESULTS

### 4.1 Performance Metrics and Environment Setup

This section presents the performance analysis of the ERSPA algorithm. The metrics for performance are given as follows, (1) Average number of relay sensors: The average number of relay sensors is defined as the required minimal average number of additional sensors to make the network connected; (2) Time complexity: Time complexity is defined as the time to run the algorithm.

The environment setup of the simulation is described as follows. There are different numbers of terminal sensors randomly deployed in a  $100 \times 100$  two-dimensional sensing area. The maximum transmission range is 10 percent of the side of the square sensing field. The network can convey location information of terminal nodes to the base station, then limit the transmission range to a bounded value  $R_r$  for energy efficiency.

### 4.2 Numerical Results

Comparisons of the two performance metrics were made for three schemes: the ERSPA algorithm, the Minimum Spanning Tree (MST) algorithm and the Greedy algorithm [13]. The MST algorithm has two steps. First, it generates a minimum spanning tree to connect the terminal nodes. Then, it places the relay nodes on the edges of the minimal spanning tree that are longer than the transmission range  $R_r$ . The MST algorithm takes  $O(N \log N)$  times.

The performance metrics include the total number of relay sensors and the time complexity. The details are illustrated as follows,



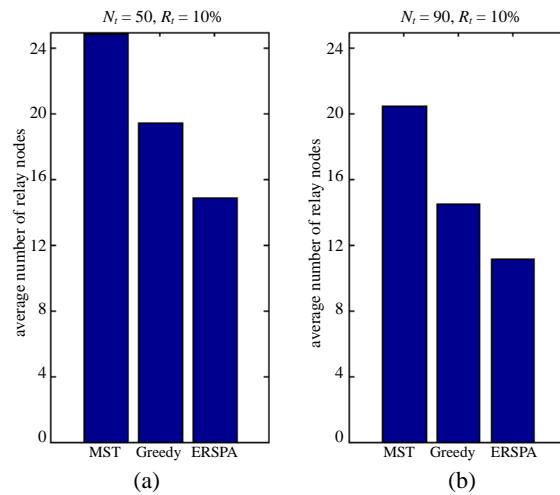


Fig. 7. (a) The average number of relay nodes for connectivity with 50 terminal nodes; (b) The average number of relay nodes for connectivity with 90 terminal nodes. The transmission range is 10 percent of the side of the square sensing field.

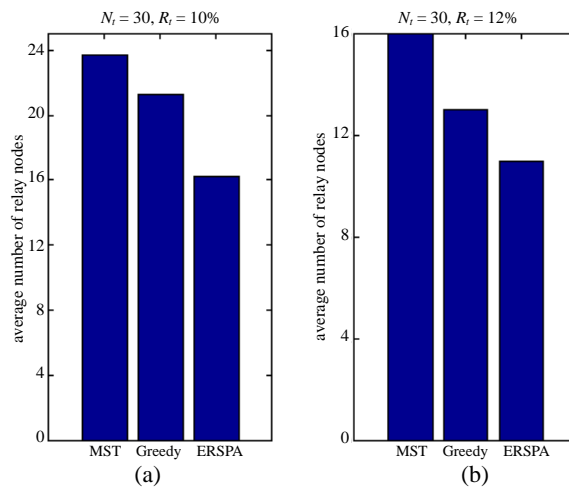


Fig. 8. (a) The average number of relay nodes for connectivity with 30 terminal nodes. The transmission range is 10 percent of the side of the square sensing field; (b) The average number of relay nodes for connectivity with 30 terminal nodes. The transmission range is 12 percent of the side of the square sensing field.

- (1) Average number of relay sensors: Fig. 7 shows that the average number of relay nodes of ERSPA is smaller than that of the minimum spanning tree algorithm and the greedy algorithm with different number of terminal nodes. Fig. 8 shows that the average number of relay nodes of ERSPA is smaller than the minimum spanning tree algorithm and the greedy algorithm when  $N_t$  is 30 and  $R_t$  is different. Fig. 9 shows that the average number of ERSPA relay nodes is smaller than that of the minimum spanning tree and the greedy algorithms under different terminal nodes when  $R_t$  is 10 percent of the side of the square sensing field. Fig. 10 shows that the average number of

relay nodes of ERSPA is also smaller than that of the minimum spanning tree and the greedy algorithms when  $R_r$  is 12 percent of the side of the square sensing field. The average number of relay sensors in MST is approximately two times that of the ERSPA algorithm. This is because ERSPA finds the optimal locations to place relay sensors and connects the maximal number of disconnected sub-graphs.

- (2) Time complexity: The time complexity of ERSPA algorithm is illustrated as follows. Phase 1 costs  $O(N \log N)$  times to construct Delaunay triangulations where  $N$  is the number of terminal nodes. Steps 1-2 of Phase 2 cost  $O(N)$  in plane and  $O(N^2)$  in three-dimensional sensing space. Steps 3-5 of Phase 2 cost  $O(N)$ . Therefore, Phase 2 requires  $O(N)$  times to add relay sensors in two-dimensional sensing space and  $O(N^2)$  in three-dimensional sensing space. The total time complexity of the ERSPA algorithm is  $O(N^2)$ . The greedy algorithm takes  $O(N^2)$  times. The minimum spanning tree takes  $O(N \log N)$  times. The scheme in [12] takes  $O(N^3)$  times. The time complexity of ERSPA is feasible in a two dimensional space.

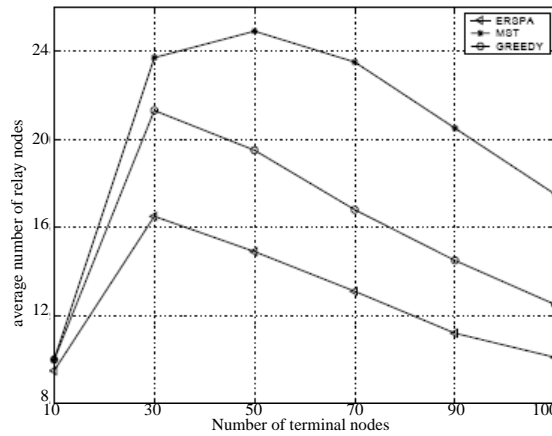


Fig. 9. The average number of relay nodes for connectivity under different terminal nodes. The transmission range is 10 percent of the side of the square sensing field.

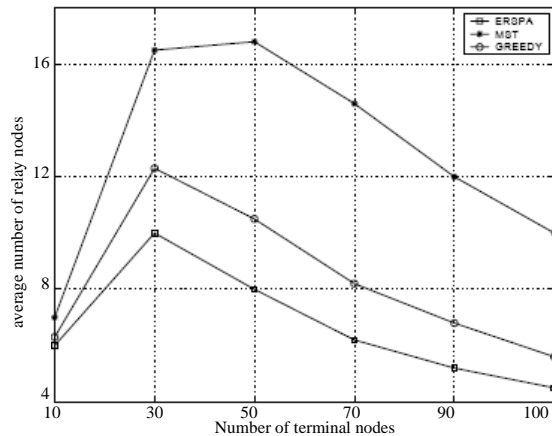


Fig. 10. The average number of relay nodes for connectivity under different terminal nodes. The transmission range is 12 percent of the side of the square sensing field.

**Table 1. Time complexity of different schemes.**

Schemes	Time complexity
MST	$O(N \log N)$
Greedy	$O(N^2)$
The scheme in [12]	$O(N^3)$
ERSPA	$O(N^2)$

## 5. CONCLUSIONS

This paper presents an efficient relay sensor placing algorithm for connectivity in wireless sensor networks. Compared with the minimal spanning tree and the greedy algorithms, our ERSPA algorithm gives better performance in terms of the average number of relay sensors. This is because ERSPA places the relay sensors in optimal places to connect the maximum number of initial connected sub-graphs such that the average number of relay sensors can be minimized. We are confident that ERSPA is an efficient and useful algorithm for further wireless ad hoc sensor networks.

## REFERENCES

1. R. J. Wan and C. W. Yi, "Asymptotic critical transmission radius and critical neighbor number for  $K$ -connectivity in wireless ad hoc networks," in *Proceedings of the 5th ACM International Symposium on Mobile Ad Hoc Networking and Computing*, 2004, pp. 1-8.
2. P. Santi, "The critical transmitting range for connectivity in mobile ad hoc networks," *IEEE Transactions on Mobile Computing*, Vol. 4, 2005, pp. 310-317.
3. H. Koskinen, "A simulation-based method for predicting connectivity in wireless multihop networks," *Telecommunication Systems*, Vol. 26, 2004, pp. 321-338.
4. H. Zhang and J. C. Hou, "Maintain sensing coverage and connectivity in large sensor networks," *Ad Hoc and Sensor Networks*, Vol. 1, 2005, pp. 89-124.
5. D. Tian and N. D. Georganas, "Connectivity maintenance and coverage reservation in wireless sensor networks," *Ad Hoc Networks*, Vol. 3, 2005, pp. 744-761.
6. H. Gupta, S. R. Das, and Q. Gu, "Connected sensor cover: Self-organization of sensor networks for efficient query execution," in *Proceedings of ACM Symposium on Mobile Ad Hoc Networking and Computing*, 2003, <http://www.cs.sunysb.edu/~hgupta/ps/coverage-mobihoc-2003.pdf>.
7. X. Wang, G. Xing, Y. Zhang, C. Lu, R. Pless, and C. Gill, "Integrated coverage and connectivity configuration in wireless sensor networks," in *Proceedings of the 1st International Conference on Embedded Networked Sensor Systems*, 2003, pp. 28-39.
8. X. Cheng, D. Z. Du, L. Wang, and B. Xu, "Relay sensor placement in wireless sensor networks," *IEEE Transactions on Computers*, Vol. 56, 2001, pp. 1-19.
9. E. L. Lloyd and G. Hue, "Relay node placement in wireless sensor networks," *IEEE Transactions on Computers*, Vol. 58, 2007, pp. 134-138.
10. I. Mandoiu and A. Zelikovsky, "A note on the MST heuristic for bounded edge-length Steiner trees with minimum number of Steiner points," *Information Process-*

- ing Letters*, Vol. 69, 1999, pp. 53-57.
11. D. Chen, D. Z. Du, X. Xu, G. H. Lin, L. Wan, and G. Xue, "Approximations for Steiner trees with minimum number of Steiner points," *Journal of Global Optimization*, Vol. 18, 2000, pp. 17-33.
  12. X. Cheng, D. Z. Du, L. Wang, and B. Xu, "Relay sensor placement in wireless sensor networks," *Wireless Networks*, Vol. 14, 2008, pp. 347-355.
  13. Y. T. Hou, Y. Shi, H. D. Sherali, and S. F. Midkiff, "On energy provisioning and relay node placement for wireless sensor networks," *IEEE Transactions on Wireless Communications*, Vol. 4, 2005, pp. 2579-2590.
  14. H. Koskinen, J. Karvo, and O. Apilo, "On improving connectivity of static ad-hoc networks by adding nodes," *Med-Hoc-Net*, 2005, pp. 169-178.
  15. D. Ganesan, R. Christescu, and B. B. Lozano, "Power-efficient sensor placement and transmission structure for data gathering under distortion constrain," *ACM Transactions on Sensor Networks*, Vol. 2, 2006, pp. 155-181.
  16. K. Mulmuley, *Computational Geometry*, Prentice Hall, Englewood Cliffs, NJ, 1994.

**Jyh-Huei Chang (張志輝)** received his M.S. degree in Computer Science from National Chiao Tung University, Taiwan in 2002. He is currently working toward the Ph.D. degree in Computer Science at National Chiao Tung University, Taiwan. He is a Ph.D. candidate now. His research interests include wireless networks, wireless sensor networks and mobile computing.