Research Article

Obstacle Aware Connectivity Restoration for Disjoint Wireless Sensor Networks Using a Mix of Stationary and Mobile Nodes

Guizhen Ma, 1,2 Yang Yang, 1 Xuesong Qiu, 1 and Zhipeng Gao 1

1 State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications, Beijing 100876, China
2 Tourism College of Beijing Union University, Beijing 100106, China

Correspondence should be addressed to Guizhen Ma; mgz@bupt.edu.cn

Received 10 October 2015; Revised 11 March 2016; Accepted 24 April 2016

Academic Editor: Stefano Avallone

Copyright © 2016 Guizhen Ma et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

In many applications, wireless sensor networks (WSNs) are deployed in inhospitable environments and therefore the nodes are at increased risk of failure. Large scale damage may partition a network into disjoint segments, which has very negative effect on the application. Mobile nodes have been exploited to act as mobile data collectors (MDCs) among segments to restore the connectivity of WSNs in the previous works. However, almost all of the works assumed that mobile nodes travel via a direct path, which may not be the case in real-world applications. In order to improve the applicability of the recovery strategy, obstacles should be taken into account. In this paper, we present an obstacle aware connectivity restoration strategy (OACRS) to fit these requirements. Our algorithm is designed for the scenarios that the number of available relays is less than the number of relays required to construct stable links and more than the number of segments. At first we construct and optimize the obstacle-avoiding minimum spanning tree of the segments and then determine the relays which keep static and the ones which act as MDCs. Finally we optimize the tour path of MDCs. The effectiveness of the proposed algorithm is validated through simulation experiments.

1. Introduction

In recent years, wireless sensor networks (WSNs) have attracted many research attentions in numerous applications [1–4]. In many applications, WSNs are deployed into hazard environment, such as battle field, border protection, and security surveillance, where the networks may suffer large scale damage and be partitioned into disjoint segments. For example, in a battle field, multiple nodes would be destroyed by explosives and the surviving nodes are split into disjoint partitions, which prevent data exchange and hinder coordination among some nodes. Thus, connectivity restoration of intersegments would be essential for the networks to resume full operation.

A number of reactive approaches are proposed to establish connectivity among WSN segments in the literature. Some of the approaches form intersegment paths by repositioning some nodes from the individual segments [5–9]. However, this strategy is feasible only when the network is composed of movable nodes. The other strategy is to deploy additional relay nodes to relink segments. The approaches proposed in [10–12] involve the deployment of static relay nodes to establish stable communication links among segments, which is only suitable for the scenario that relay nodes are sufficient. In many emergent situations, it is difficult to obtain sufficient relay nodes. When the number of relay nodes is not enough to construct stable links among the partitions, some approaches deploy mobile data collectors (MDCs) traveling among the segments to provide intermittent communication links [9, 12–15]. MDCs can be designated to serve on the individual links and carry data from nodes in one segment to recipients in another.

When involved in the mobile node, almost all of the works assume that destinations can be reached via direct path movement. The only two recent works considering obstacle avoidance in connectivity restoration can be found in [5, 8].
However, neither of the two works considers the scenario of how to place stationary and mobile nodes among disjoint segments to restore the connectivity of the WSNs.

In this paper, we present an obstacle aware connectivity restoration strategy (OACRS), which considers the scenario that available relays are not sufficient to construct stable links but the number is more than that of segments. OACRS exploits a hybrid solution that employs a mix of stationary and mobile relays to restore the connectivity of the network. Rather than following a direct path, obstacles are taken into account when the MDCs are moving. The main objective of OACRS is to minimize the total tour length and maximum tour length. The strategy is divided into two phases. In the first phase, we propose an algorithm to construct and optimize the obstacle-avoiding minimum spanning tree (oamst) of all the segments. And in the second phase, an algorithm is present to determine which edges of the oamst will be served by MDCs and optimize the tours path to minimize the tour length. In order to avoid interference of signal caused by obstacles, we employed MDCs on the routes along the obstacle boundary. The simulation results confirm the effectiveness of OACRS compared to recent work in the literature.

This paper is organized as follows. Related work is summarized in Section 2. The system model and assumptions are discussed in Section 3. In Section 4, the proposed OACRS approach is described in detail. Section 5 presents the simulation results and finally Section 6 concludes the paper.

2. Related Work

The published relays placement schemes about connectivity restoration of disjoint WSNs segments fall into three categories. The first category is to establish a stable intersegment topology under the scenario that all relay nodes are stable. The second category is that all the relays act as mobile data collectors. And the third is to restore connectivity using a mix of mobile and stationary relays.

When relay nodes are sufficient to reestablish connectivity, the connectivity restoration problem becomes the problem of restoring connectivity with the fewest number of relays. Ma et al. [10] model the two-tiered WSNs relay nodes placement problem as a GDC (Geometric Disc Covering) problem and improve the network robustness against connection failure by solving the relay nodes double cover problem. Sitanyah et al. [11] propose GRASP-AR, which uses counting paths to minimize the number of relay nodes. Han et al. [12] address the minimum number of relay nodes’ placement in the context of H-WSNs. These approaches only suit for the scenario that there are sufficient relay nodes to establish a stable intersegment topology. However, in real-world applications, especially in an emergency, such as battlefield, it is hard to get enough relays.

When the number of additional nodes is less than the number of segments, all the relay nodes need to travel around to provide intermittent communication links. Senel and Younis [9] propose IDM-kMDC, in which disjoint segments are interconnected through k MDCs. IDM-kMDC merges two collocated MDCs to reduce the number of MDCs by one in each round. And it strives to minimize the merged tour length. Senturk et al. [14] exploit Game Theory to orchestrate the self-spreading process of mobile nodes.

Some other published approaches use a mix of stationary and mobile relays to restore connectivity of intersegments, such as MiMsi [13], FeSMoR [16], and ToCS [17]. These algorithms work under the situation that available relays are not sufficient to form stable data path among segments but the number of the available relays is still more than the count of segments. Thus, some of the relay nodes will travel among segments to establish intermittent links. MiMsi exploits the availability of relays to shorten the travel path of MDCs and lower the data delivery latency. FeSMoR assumes the relays are sufficient and the RN placement problem is modeled as Steiner Minimum Tree. And it incrementally replaces groups of relays on the Steiner Tree with mobile nodes. FeSMoR meets the resource availability constraints and achieves the minimal delay objective by exploiting relay mobility at the periphery of the federated intersegment topology. ToCS uses the available MRNs to form a simple star topology and it aims at minimizing both the maximum and average data delivery latency between any two segments.

OACRS also uses a mix of stationary and mobile relays to restore connectivity. However, unlike FeSMoR, OACRS aims at minimizing the maximum tour length and the total tour length. OACRS is also different from MiMsi. There is no limitation to the number of static nodes and mobile nodes in OACRS, while MiMsi assumes both numbers of stationary and mobile relays are fixed.

The only two works in which obstacles are under consideration [5, 8] are also different from OACRS both in application scenarios and in algorithms themselves. In OCRS [5] the failure of a single cut-vertex sensor of MRSNs is addressed and the connectivity is restored by cascaded nodes movement. Obstacle-avoiding problem is addressed when the nodes are moving. Senturk et al. [8] consider the terrain characteristics when mobile nodes are moving and pick the path with the least risk rate for the mobile nodes. This approach repositions nodes from various segments to restore network connectivity, which is much difficult to keep connectivity within segments, while OACRS addresses the connectivity restoration problem of intersegments under the scenario that available relays are insufficient but the number is more than the number of segments, which is not considered in [5, 8].

3. System Model

We consider the wireless sensor network, which suffers large scale damage and is divided into several disjoint segments, and there are obstacles among the partitions. The system model can be shown in Figure 1. For simplicity’s sake, each segment is represented by a node (collection point) in our work. And the node corresponds to the common communication range R (R is the maximum Euclidean distance that its radio can reach).

We also have the following assumptions and definitions.

**Assumption 1 (relay nodes).** All the relay nodes can move on demand.
Assumption 2 (obstacles). There is at least one edge in the minimum spanning tree which intersects the obstacles. In OACRS, an obstacle is defined as a convex polygon of any size, and the count of the vertex of one obstacle is not less than 4. In addition, any obstacles cannot overlap with segments.

Definition 3. Given an undirected graph $G = (S, E)$, where $S = \{s_1, s_2, s_3, \ldots, s_n\}$ is the set of segments and $E = \{(s_i, s_j) | s_i, s_j \in S\}$ is the set of edges, let $T = (S, E_t)$ denote the obstacles-avoiding minimum spanning tree (oamst), where $E_t$ represents the set of edges of oamst. Let $E_{\text{remove}} = E - E_t$, $\forall(s_i, s_j) \in E_t$, and $\neg\exists(s_m, s_n) \in E_{\text{remove}} \land \text{dis}(s_i, s_j) < \text{dis}(s_m, s_n)$, where $\text{dis}(s_i, s_j)$ is the Euclidean distance of segments $s_i$ and $s_j$.

Definition 4. $\forall(s_i, s_j) \in E_t$, $\text{RN}(s_i, s_j)$ denotes the number of relay nodes between segments $s_i$ and $s_j$. Then $\text{RN}(s_i, s_j)$ can be computed through the following formula:

$$\text{RN}(s_i, s_j) = \left\lfloor \frac{\text{dis}(s_i, s_j)}{R} \right\rfloor - 1. \quad (1)$$

Definition 5. Let $\text{RN}_{\text{total}}$ denote the total number of relay nodes required to form a stable topology. $\text{RN}_{\text{total}}$ can be computed through the following formula:

$$\text{RN}_{\text{total}} = \sum_{(s_i, s_j) \in E_t} \text{RN}(s_i, s_j). \quad (2)$$

Assumption 6. Let $M$ denote the count of segments. The number of available relay nodes $\text{RN}_{\text{available}}$ satisfies the following criteria: $\text{RN}_{\text{available}} > M$ and $\text{RN}_{\text{available}} < \text{RN}_{\text{total}}$.

System Model. The mathematical model of OACRS can be simplified this way: let $S = \{s_1, s_2, s_3, \ldots, s_n\}$ be the set of segments and $P = \{p_1, p_2, p_3, \ldots, p_m\}$ ($m \geq 4$) the set of the vertexes of the obstacle, where $p_i$ cannot overlap the segment $s_i$. Let $\text{RN}_{\text{total}}$ represent the number of relays required to form a stable topology. Let $\text{RN}_{\text{available}} = \{n_1, n_2, n_3, \ldots, n_k\}$ denote the set of available relays, where $k$ satisfies the following condition: $k > M \land k < \text{RN}_{\text{total}}$.

4. Obstacle-Avoiding Connectivity

Restoration Strategy

In this section, OARCS approach is described in detail. At first we construct and optimize the oamst. Then the delays deployment problem is handled. Finally the tour path of MDCs is optimized.

4.1. Obstacle-Avoiding Minimum Spanning Tree Construction.

OACRS aims to shorten the tour length of MDCs. Thus we model the relays placement problem as an obstacle-avoiding minimum spanning tree. The algorithm will be handled as follows.

Construction of Minimum Spanning Tree without Considering Obstacles. At first we construct the minimum spanning tree (mst) of the segments using Prim algorithm without considering obstacles, where each segment is represented by a collection point. As shown in Figure 2(a), the wireless sensor network is divided into ten disjoint segments which are presented by $[s_1, s_2, s_3, \ldots, s_{10}]$. $P = \{p_1, p_2, p_3, p_4\}$ denotes the set of vertexes of the obstacle, and edge $s_2\text{-}s_7$ intersects the obstacle $P$.

Construction and Optimization of Obstacle-Avoiding Minimum Spanning Tree. When one or more edges intersect obstacles, we change the routes to avoid obstacles. There are two scenarios.

1. Only one edge of the oamst intersects the obstacle. Suppose edge $(s_i, s_j)$ divides the obstacle into two partitions and $r_1$ and $r_2$ are two routes linking $s_i$ to $s_j$ in the oamst. OACRS takes tour length as optimization objective. Thus, we should choose the shorter route as the tour path of MDCs. As shown in Figure 2(b), the route $r_1$ is denoted by $s_2\text{-}p_1\text{-}p_2\text{-}s_7$ and $r_2$ is $s_2\text{-}p_2\text{-}p_3\text{-}s_7$. The lengths of $r_1$ and $r_2$ satisfy the following condition: length($r_1$) < length($r_2$). Thus we choose $r_1$ as the tour path linking up segments $s_i$ and $s_j$.

2. More than one edge intersects the obstacle. If there is more than one edge intersecting the obstacle, each edge should be replaced by a new route using the method described in (1). As shown in Figure 2(c), there are two edges that intersect the obstacle simultaneously, and the better replacements are $s_2\text{-}p_1\text{-}p_4\text{-}s_7$ and $s_3\text{-}p_2\text{-}p_4\text{-}s_8$, respectively. However, rings may be formed while searching the replacement, which is forbidden in oamst. Once a ring is formed, we destroy the ring by removing the longest edge in the ring. This case is illustrated in Figure 2(d). Edges $s_2\text{-}s_7$ and $s_3\text{-}s_8$ should be replaced by $s_2\text{-}p_1\text{-}s_7$ and $s_3\text{-}p_1\text{-}p_4\text{-}s_8$, respectively. However, ring $s_3\text{-}s_2\text{-}p_1\text{-}s_3$ is formed, so we remove the longest edge $s_3\text{-}s_2$.

Improvement of Obstacle-Avoiding Minimum Spanning Tree.

The obstacle-avoiding spanning tree we have constructed...
may be not the minimum spanning tree because we replace some edges by routes, and the length of routes may be longer than that of a qualified link in $E_{\text{remove}}$. OACRS tries to find edge $e_i$ in $E_{\text{remove}}$, which meets the following conditions: (i) when $e_i$ substitutes for a route in the spanning tree, the network keeps intersegment connected and there is no ring formed. (ii) $e_i$ is the shortest link in $E_{\text{remove}}$ which satisfies condition (i). As shown in Figure 2(b), the edge meeting conditions (i) and (ii) simultaneously is $s_1-s_6$, which satisfies length($s_1-s_6$) < length($s_2-p_1-p_4-s_8$). Now we eliminate route $s_2-p_1-p_4-s_8$ and link $s_1$ to $s_6$. If there is no ring formed, edge $s_1-s_6$ will substitute for $s_2-p_1-p_4-s_8$, which is shown in Figure 2(e).

Figure 2(d) shows a more complicated case.

(i) The shortest link in $E_{\text{remove}}$ is $s_7-s_8$; (ii) length($s_7-s_8$) > length($s_2-p_1-s_7$) and length($s_7-s_8$) < length($s_3-p_1-p_4-s_8$); (iii) there is no ring formed when edge $s_7-s_8$ substitutes for route $s_3-p_1-p_4-s_8$. Thus we remove $s_3-p_1-p_4-s_8$ from Figure 2(d) and link $s_7$ to $s_8$. However, Figure 2(d) will be divided into two partitions $\{s_1, s_2, p_1, s_7, s_8, s_9, s_10\}$ and $\{s_7, s_8, s_9\}$. This is because when $s_3-p_1-p_4-s_8$ substitutes for $s_3-s_8$, edge $s_2-s_3$ is eliminated to destroy the ring. We should link $s_2$ to $s_3$ again to keep the connectivity of the segments. The improved obstacle-avoiding minimum spanning tree is shown in Figure 2(f).

### 4.2. Federating Disjoint Segments

We assume relays are sufficient firstly and employ the relay nodes on each edge or route of the oamst. As is shown in Figure 3(a), each edge or route in oamst is served by one or more relays to form a stable topology among these segments. However, in this paper, we consider the case that available relays are not sufficient; that is to say, $R_{\text{available}} < R_{\text{total}}$. Thus OACRS tries to serve an edge, part of an edge, or several edges with a single MDC to meet the relays availability constraints. The question is which edge(s) should be served by static relays and which should be served by MDCs to minimize the total tour length and maximum tour length. This process is done as follows.

**Handling the Employment on the Route.** We should emphasize that the obstacles may interfere with the signal of the stable relays. Thus we place MDCs on the routes. Let $L_{\text{route}}(s_i, s_j)$ denote the length of a route linking $s_i$ to $s_j$ and let $\text{Avg } L_{\text{edge}}$ represent the average length of the edges in oamst. $L_{\text{route}}(s_i, s_j)$ and $\text{Avg } L_{\text{edge}}$ can be computed from formulas (3) and (4), respectively:

$$L_{\text{route}}(s_i, s_j) = \text{length } (s_i, p_i) + \sum_{p_m, p_n, E_{\text{route}}} \text{length } (p_m, p_n) + \text{length } (p_j, s_j)$$

(3)

$$\text{Avg } L_{\text{edge}} = \frac{\sum_{(s_i,s_j) \in E_{\text{route}}} \text{length } (s_i, s_j)}{N_{\text{edge}}}$$

(4)

where $N_{\text{edge}}$ is the count of edges in oamst.

In order to shorten the maximum tour length, the tour path of MDCs must follow the rule below: the tour length of each MDC cannot be longer than $\text{Avg } L_{\text{edge}}$.

If $L_{\text{route}}(s_i, s_j) < \text{Avg } L_{\text{edge}}$, the route needs one MDC.
If \( L_{\text{route}}(s_i, s_j) > n \text{Avg} L_{\text{edge}} \) and \( L_{\text{route}}(s_i, s_j) < (n + 1) \text{Avg} L_{\text{edge}} \), the route needs \((2n + 1)\) relays, where \( n \) relay nodes keep stable and the other \((n + 1)\) relays act as MDCs to form an intermittent link.

After the placement of relays on route \( s_i \cdot p_1 \cdot \ldots \cdot p_m \cdot p_j \cdot s_j \), we assume the released number of relays is \( \text{RN}_{\text{release}} \). If \( \text{RN}_{\text{total}} - \text{RN}_{\text{release}} = \text{RN}_{\text{available}} \), the algorithm ends. Otherwise OACRS continues to place MDCs on edges of the oamst.

**Employment on the Edges.** Let \( \text{length}(s_i, s_j) \) denote the Euclidean distance between segments \( s_i \) and \( s_j \), where \( s_i \cdot s_j \) is an edge of oamst. We try to find out all the adjacent edges of the tree. As shown in Figure 3(b), each triangle has one edge expressed by a dotted line. Then we compute the perimeter of the triangles which are constructed by the terminals of the two adjacent edges. Let \( \text{min} L \) denote the length of the shortest edge in \( E_i \) and \( \text{min} P \) shortest perimeter. At each iteration, OACRS firstly compares \( \text{min} L \) and \( \text{min} P \). If \( \text{min} L < \text{min} P \), the shortest edge is chosen. There are three cases:

1. If \( \text{RN}_{\text{total}} - \text{RN}_{\text{release}} > \text{RN}_{\text{available}} \), OACRS will place a MDC on the shortest edge and eliminate the edge from \( E_i \). The iteration continues.

2. If \( \text{RN}_{\text{total}} - \text{RN}_{\text{release}} = \text{RN}_{\text{available}} \), the algorithm ends.

3. If \( \text{RN}_{\text{total}} - \text{RN}_{\text{release}} < \text{RN}_{\text{available}} \), the MDCs on the chosen edge will reduce tour length; that is to say, OACRS will place some stationary relays and a MDC on the edge. And the algorithm ends.

If \( \text{min} P < \text{min} L \), OARS will place a MDC on the triangle. Then the following treatment is similar to (1), (2), and (3).

**Illustrative Example.** Figure 3 illustrates the operation of OACRS on federating 10 segments \( s_1, s_2, \ldots, s_{10} \) with 15 relays. Based on the communication range, the total number of relays \( \text{RN}_{\text{total}} \) is 23. Firstly, we assume 23 stable relays are placed on edges or routes, as shown in Figure 3(a). Then we will determine the number of relays placed on route \( s_1 \cdot p_4 \cdot p_5 \cdot s_7 \). The length of the route \( s_2 \cdot p_1 \cdot p_4 \cdot s_7 \) satisfies the following conditions: (i) \( L_{\text{route}}(s_2, s_7) > \text{Avg} L_{\text{edge}} \), and (ii) \( L_{\text{route}}(s_2, s_7) < 2 \text{Avg} L_{\text{edge}} \). We have emphasized that the tour length of each MDC cannot be longer than \( \text{Avg} L_{\text{edge}} \). Each route needs \((2n + 1)\) relays to link up two segments, where \( n \) follows

\[
 n = \left\lfloor \frac{L_{\text{route}}(s_i, s_j)}{\text{Avg} L_{\text{edge}}} \right\rfloor.
\]

According to conditions (i) and (ii), we are aware that \( n = 1 \). So route \( s_2 \cdot p_1 \cdot p_4 \cdot s_7 \) needs 3 relay nodes, where one keeps stable and the other two nodes act as MDCs. As shown in Figure 3(a), if we only place stable nodes, the route \( s_2 \cdot p_1 \cdot p_4 \cdot s_7 \) needs 5 nodes. Thus 2 nodes are released. The placement of relays on edges is an iterative process, where 5 MDCs institute the remaining 11 RNs. Firstly, OACRS compares \( \text{min} L \) and \( \text{min} P \). The length of edges \( s_2 \cdot s_4, s_4 \cdot s_5, \) and \( s_1 \cdot s_5 \) follows \( L_{\text{edge}}(s_i, s_j) > R \wedge L_{\text{edge}}(s_i, s_j) < 2R \), so here no relay needs to be released. Then OACRS places one MDC.
5.1. Experiment Setup and Performance Metrics. In the simulation experiments, segments are randomly placed in an area of 1500 m × 1500 m with one obstacle. The number of segments “n” is varied from 3 to 13. The communication range “R” for the collection points and MDCs has been set to 60, 80, 100, 120, 140, and 160. When changing the number of segments, the communication range R is fixed at 100 m; and the segments count is set to 8 while varying the communication range. In our simulation we set the number of available relays using the following formula:

\[ RN_{\text{available}} = RN_{\text{total}} \times p, \]

where \( 0 < p < 1 \). We repeat the experiment for different \( p \) values (0.6, 0.7, and 0.8) for a fair comparison, and \( RN_{\text{available}} \) is kept greater than or equal to the number of segments. We use the total tour length and maximum tour length as the performance metrics:

(i) Total tour length is the sum of distances traveled by all employed MDCs. Our goal is to minimize this metric in order to limit the overhead imposed on the MDCs.

(ii) Maximum tour length is the longest distance among all MDCs tours. This metric indicates the load that one of the MDCs will experience, and a large value indicates increased energy consumption, which would affect the rate of energy depletion and the node lifetime. The goal is to minimize this distance.

In our experiment, each individual simulation experiment involves 30 different topologies and the average result is reported.

5.2. Experiment Results

5.2.1. Total Tour Length. Figure 5 shows the effects of different parameters on total tour length. As shown in Figures 5(a) and 5(b), we can see that the total tour length decreases as the value of \( p \) increases. This is because the number of nodes which act as MDCs can be reduced when there are more available relays. Thus, the sum of tour length of all MDC will be reduced. From Figure 5(a), we observer that the changing of total tour length is not too large in whole while the number of segments increases, which indicates that the number of segments is not strict in OACRS. Figure 5(b) shows that the total tour length decreases as the values of communication range increase. This is because OACRS optimized MDCs tours, and the tour length from one segment to another is \( d - 2R \), where \( d \) is the Euclidean distance between the two segments.
5.2.2. Maximum Tour Length. Figure 6 shows the trend of maximum tour length with varying number of segments and communication range. As shown in Figures 6(a) and 6(b), we can see that the maximum tour length decreases as the value of \( p \) increases in most cases. This is because the smaller \( p \) is, the smaller the number of available relays is; then the distance of each MDC is increased. Thus, the maximum tour length increases as \( p \) decreases. From Figure 6(a), we can observe that the maximum tour length decreases as the number of segments increases, which is because the long tour decreases in dense networks. And we can observe in Figure 6(b) that the maximum tour length decreases as communication range increases, which is also because OACRS optimized MDCs tours, and when communication range increases, the value of \( d - 2R \) is decreased.

5.2.3. Comparison with FeSMoR. We compare the performance of OACRS with that of FeSMoR, which also uses mix of mobile and stationary relays to restore connectivity of WSNs. With the same setup, the results for the total tour length and the maximum tour length are shown in Figures 7(a) and 7(b), respectively. As can be seen from Figures 7(a) and 7(b), when the number of segments is more than 8, OACRS outperforms FeSMoR in terms of both total tour length and maximum tour length, which confirms that OACRS is more suitable for the case that the distance between segments is not too large.
6. Conclusion

Restoring network connectivity is crucial to ensure the WSNs functions effectively. In this paper, we studied the connectivity restoration problem of WSNs, where obstacles are considered when MDCs are traveling.

We have proposed OACRS, an obstacle aware connectivity restoration strategy. OACRS firstly constructs and optimizes the obstacle-avoiding minimum spanning tree among the segments and then determines the placement of relay nodes. We further optimize the tour path of MDCs to minimize the touring length. The simulation results have confirmed the effectiveness of OACRS. Our future work is to evaluate the performance of the proposed approaches when there is more than one obstacle and considering the signal interference.

Competing Interests

The authors declare that they have no competing interests.

Acknowledgments

This work was partly supported by NSFC (61401033, 61372108, and 61272515) and Beijing Higher Education Young Elite Teacher Project (YETP0474).

References

[1] N. Bahrami, N. H. H. Khamis, and A. B. Baharom, "Study of underwater channel estimation based on different node placement in shallow water," IEEE Sensors Journal, vol. 16, no. 4, pp. 1095–1102, 2016.

[2] Z. Liao, J. Wang, S. Zhang, J. Cao, and G. Min, "Minimizing movement for target coverage and network connectivity in mobile sensor networks," IEEE Transactions on Parallel and Distributed Systems, vol. 26, no. 7, pp. 1971–1983, 2015.

[3] H. F. Rashvand, A. Abedi, J. M. Alcaraz-Calero, P. D. Mitchell, and S. C. Mukhopadhyay, "Wireless sensor systems for space and extreme environments: a review," IEEE Sensors Journal, vol. 14, no. 11, pp. 3955–3970, 2014.

[4] Q. Liang, X. Cheng, S. C. Huang, and D. Chen, "Opportunistic sensing in wireless sensor networks: theory and application," IEEE Transactions on Computers, vol. 63, no. 8, pp. 2002–2010, 2014.

[5] Z. Mi, Y. Yang, and J. Y. Yang, "Restoring connectivity of mobile robotic sensor networks while avoiding obstacles," IEEE Sensors Journal, vol. 15, no. 8, pp. 4640–4650, 2015.

[6] K. Akkaya, F. Senel, A. Thimmapuram, and S. Uludag, "Distributed recovery from network partitioning in movable sensor/actor networks via controlled mobility," IEEE Transactions on Computers, vol. 59, no. 2, pp. 258–271, 2010.

[7] Z. Mi and Y. Yang, "Connectivity restorability of mobile ad hoc sensor network based on K-hop neighbor information," in Proceedings of the IEEE International Conference on Communications (ICC’11), pp. I–5, IEEE, Kyoto, Japan, June 2011.

[8] I. F. Senturk, K. Akkaya, and S. Janansefat, "Towards realistic connectivity restoration in partitioned mobile sensor networks," International Journal of Communication Systems, 2014.

[9] F. Senel and M. Younis, "Optimized interconnection of disjoint wireless sensor network segments using K mobile data collectors," in Proceedings of the IEEE International Conference on Communications (ICC ’12), pp. 492–496, Ottawa, Canada, June 2012.

[10] C. Ma, W. Liang, M. Zheng, and H. Sharif, "A connectivity-aware approximation algorithm for relay node placement in wireless sensor networks," IEEE Sensors Journal, vol. 16, no. 2, pp. 515–528, 2016.

[11] L. Sitanayah, K. N. Brown, and C. J. Sreenan, "A fault-tolerant relay placement algorithm for ensuring k vertex-disjoint shortest paths in wireless sensor networks," Ad Hoc Networks, vol. 23, pp. 145–162, 2014.

[12] X. Han, X. Cao, E. L. Lloyd, and C.-C. Shen, "Fault-tolerant relay node placement in heterogeneous wireless sensor networks,"
IEEE Transactions on Mobile Computing, vol. 9, no. 5, pp. 643–656, 2010.

[13] A. Abbas and M. Younis, “Interconnecting disjoint network segments using a mix of stationary and mobile nodes,” in Proceedings of the 37th Annual IEEE Conference on Local Computer Networks (LCN ’12), pp. 28–35, Clearwater, Fla, USA, October 2012.

[14] I. Senturk, K. Akkaya, and S. Yilmaz, “Distributed relay node positioning for connectivity restoration in partitioned wireless sensor networks,” in Proceedings of the IEEE International Symposium and Computers and Communications (ISCC ’12), Capadocia, Turkey, July 2012.

[15] I. Chatzigiannakis, A. Kinalis, and S. Nikoletseas, “Sink mobility protocols for data collection in wireless sensor networks,” in Proceedings of the 4th ACM International Workshop on Mobility Management and Wireless Access (MOBIWAC ’06), Terrromolinos, Spain, October 2006.

[16] J. L. V. M. Stanislaus and M. Younis, “Delay-conscious federation of Multiple wireless sensor network segments using mobile relays,” in Proceedings of the 76th IEEE Vehicular Technology Conference (VTC ’12-Fall), pp. 1–5, IEEE, Quebec City, Canada, September 2012.

[17] J. L. V. M. Stanislaus, Mobile relays based federation of multiple wireless sensor network segments with reduced-latency [M.S. thesis], Department of Computer Science and Electrical Engineering, University of Maryland, Baltimore, Md, USA, 2012.