Journal of Universal Computer Science, vol. 21, no. 3 (2015), 384-405
submitted: 30/6/14, accepted: 28/2/15, appeared: 1/3/15 © *J.UCS*

A Resolving Set based Algorithm for Fault Identification in Wireless Mesh Networks

Xiaoding Wang, **Li Xu***[1], **Shuming Zhou**
(Fujian Normal University, Fuzhou, China
{wangdin1982, xuli, zhoushuming}@fjnu.edu.cn)

Joseph K. Liu
(Institute for Infocomm Research, Singapore
ksliu9@gmail.com)

Abstract: Wireless Mesh Networks (*WMN*s) have emerged as a key technology for next-generation wireless networking. By adding some Long-ranged Links, a wireless mesh network turns into a complex network with the characteristic of small worlds. As a communication backbone, the high fault tolerance is a significant property in communication of *WMN*s. In this paper, we design a novel malfunctioned router detection algorithm, denoted by *A-SRS*, on searching resolving set based on private neighbor of dominating set. The *A-SRS* not only offers a highly efficient solution to position malfunctioned routers against intermitted communication that guarantees the *availability* of network services, but also pursues the minimum number of detecting routers due to limited resource of wireless mesh routers. We also explore the cardinality of resolving set and complexity of *A-SRS* based on the parameters: the minimum degree, the size of underlying graph G and the number of iterations. The algorithm enjoys better simulation results that it employs less detecting routers than the other strategies in the size of resolving set.

Key Words: Wireless Mesh Network, Fault Tolerance, Dominating Set, Resolving Set
Category: C.2.0

1 Introduction

Wireless Mesh Networks (*WMN*s) have achieved significant development since they enjoy the properties of fast deployment, easy maintenance and low investment compared with traditional networks. With the increasing scale of *WMN*s, some stationary routers are deployed to establish a certain number of Long-ranged Links (*LLs*) that bring down the average path length of the network. In this case, a wireless mesh network can be considered as a complex network with the characteristic of small worlds [Verma et al., 2011].

Since *WMN*s are typically used as backbones of communication network, they have the nature that the communication breaks down sometimes due to the malfunction of routers. Once the sensitive information fails to deliver in time or the intermitted communication can not be restored, both of which will

[1] *Corresponding Author

cause huge loss to the entire network. Therefore, it is imperative to develop an efficient system to detect malfunctioned routers for network recovery. Moreover, in the view of network security, a highly efficient failure detecting system will help to achieve the *availability* [Bella, 2008] [Uzunov et al., 2012] [Mellado and Rosado, 2012] that guarantees network services operate properly and tolerate failures [Zhou and Haas, 1999].

Actually plenty of new theories and methods are proposed and improved for positioning problem [Gustafsson and Gunnarsson, 2005] [Patwari et al., 2005], one of which employs the technology of metric dimension [Khuller et al., 1996].

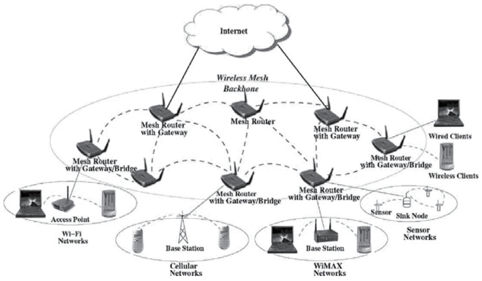

Figure 1: The topology of a wireless mesh network.

Applications of resolving sets arise in various areas including robot navigation [Khuller et al., 1996], coast guard sonar system [Slater, 1975], network discovery and verification [Beerliova et al., 2006],and fault detect and positioning for wireless network [Hoffmann and Wanke, 2013].

1.1 Motivations and Goals

This paper is motivated by the problem of uniquely determining the location of a router in wireless mesh networks due to the significance of a fault-tolerant communication in *WMN*s. It aims at designing a failure detecting algorithm to efficiently identify faulty routers for network recovery.

1.2 Our Contributions

1. In this paper, we employ the resolving set to uniquely identify mesh routers. By designating some mesh routers as resolving ones, every router is assigned a distinct coordinate such that malfunctioned routers can be detected efficiently.

2. An novel detecting algorithm based on resolving set, namely *A-SRS*, is proposed to establish a resolving set with the cardinality upper bounded by

$n \times \left(1 - \left(1 - \frac{1+\ln(\delta(G)+1)}{1+\delta(G)}\right)^k\right)$, where n denotes the number of vertices in G, $\delta(G)$ represents the minimum degree of graph G and k stands for the number of rounds the A-SRS takes to resolve G.

1.3 Network Models

In this paper, the first layer of *WMNs* that comprise mesh routers is mapped to a graph $G = (V, E)$, where V denotes the set of vertices while E stands for the set of edges in G. In general every vertex of G represents a router and every edge accounts for a wireless link between two routers. Deliberately placing some routers with fault detection algorithm to monitor the entire network is to designate some vertices of corresponding graph G as resolving vertices.

More specifically, the fault-detection algorithm bases on coordinate that it can uniquely identify every single router. For simplicity, we call the routers embedded with this algorithm detecting ones. Correspondingly in graph G, we choose a set of vertices as resolving vertices. Then every vertex is assigned a vector as its coordinate with respect to those resolving vertices. And as a coordinate, the vector is composed by lengths of the shortest paths from a specific vertex to all resolving vertices in hops such that any pair of vertices does not have the same coordinate.

In *WMN*s a router is designed to send a malfunction message once it breaks down. Therefore, after this fault-detection algorithm is applied to *WMN*s, the breaking down router will send a malfunction message toward all detecting routers. When all detecting routers gather the message, they have the knowledge of the distances from the malfunctioned one to themselves respectively. Then all distance information will be relayed to a computational station' by detecting ones. For simplicity we call this station the "center", which establishes connection to all detecting routers through satellite communication or optical fibers. Eventually the "center" simply calculates the coordinate of the malfunctioned router by combining together all distance information such that positioning and replacement on faulty router can be executed efficiently.

It is worth to mention that the positioning of resolving set is applied to distinguish different routers by giving them distinct coordinates (unique labels) over resolving set. It is different from seeking the geographical positions through GPS.

1.4 Related Work

Faulty node detection has been discussed in many research papers. Guo, et al. [Guo et al., 2009] provided a sequence-based detection approach FIND for detecting nodes with functionnal faults. Ding [Ding et al., 2005] proposes detecting faulty nodes by determining if the difference between a nodes reading and its neighbors is above a threshold. A zoning approach is proposed in [Taleb et al., 2009] that divides the network into disjoint zones while having a master for each zone. The zone masters are used to identify faulty nodes by virtually dividing the zone into quadrants until a suspect node is found.

Many researchers have invested plenty of times Metric Dimension problem. Beerliova, et al. [Beerliova et al., 2006] showed that the Metric Dimension problem (which they call the Network Verification Problem) cannot be approximated within a factor of $O(\log(n))$ unless P = NP. Halldrsson, et al. [Halldórsson et al., 2001] studied the Test Set Collection Problem with bounded test size, and they gave a $(3+3\ln(k))$-approximation algorithm for the Test Set Collection Problem with the test of the size at most k. Hauptmann [Hauptmann et al., 2012], et al. provided a constant-factor approximation algorithm, *Pre-ICH*, with approximation ratio $(2 + 2\ln(k) + \ln(log2(k-1)) + o(1))$ for k-super dense graph.

The remainder of this paper is organized as follows: Section 2 introduces some general definitions. The algorithm *A-SRS* is proposed in Section 3. All theorems and proofs are presented in Section 4. Performance evaluation is given in Section 5. Section 6 concludes this paper.

2 Preliminaries

2.1 Terminology and Background Knowledge

Throughout the paper, $G = (V, E)$ is a finite, simple, and connected graph of order n with vertex-set V and edge-set E. The distance between two vertices u and v over Graph G, denoted by $d(u, v)$, is the length of the shortest path in G from u to v. The set of paths from u to v is denoted by $sp(u, v)$. The set of all neighbors of a vertex v in G is denoted by $N(v)$ and the maximum degree of graph G is denoted by $\Delta(G)$.

Definition 1. For an ordered set $W = \{w_1, w_2, \ldots, w_k\} \subseteq V(G)$ and a vertex v of G, the k-vector $r(v|W) = (d(v, w_1), d(v, w_2), \ldots, d(v, w_k))$ is called the metric representation of v with respect to W, where $w_1, w_2, \ldots, w_k$ are called resolving vertices. And the set W is called a resolving set for G if distinct vertices have different representations or coordinates.

In addition, a vertex x resolves a pair of vertices v and w if $d(v, x) \neq d(w, x)$, where $v, w \in V(G)$. Furthermore, the vertex $w_1 \in W$ is resolved by W because of $r(w_1, W) = (0, d(w_1, w_2), \ldots, d(w_1, w_k)) \neq r(w_i, W)$ for $i \neq 1$. A resolving set for G with minimum cardinality is called a metric basis, and its cardinality is the metric dimension of G, denoted by $dim(G)$.

Definition 2. A dominating set in G is a set D of vertices such that every vertex in $V - D$ is adjacent to at least one vertex in D. And a dominating set with minimum cardinality is called the Minimum Dominating Set (MDS).

Let $MDS(G)$ represents the minimum domination set of graph G, the dominating vertices are denoted as dominators while the dominated vertices are denoted as dominatees. $\gamma(G)$ denotes the domination number of G.

Definition 3. Let u be a vertex of a graph G. The open neighborhood of u is the set of neighbors of u excluding vertex u itself, which is denoted by $N(u) = \{v \in V(G)|uv \in E(G)\}$. And the closed neighborhood of u is the set of neighbors of u including vertex u itself, which is denoted by $N[u] = N(u) \cup \{u\}$.

Twin. Two distinct vertices u and v are adjacent twins if $N[u] = N[v]$, and non-adjacent twins if $N(u) = N(v)$. We call u and v are twins, if u and v are either adjacent twins or non-adjacent twins.

Group. The vertices of graph G can be divided into groups, any two vertices of one group P_i are twins. Obviously, $P_i \bigcap P_j = \emptyset$, $i \neq j$.

Private neighbor. Let MDS be a dominating set in a connected graph G and $u \in MDS$. A vertex $v \in V(G) \backslash MDS$ is called a private neighbor of u if u is the unique neighbor of v in MDS, i.e., $N(v) \bigcap MDS = \{u\}$.

$\mathbf{G^k}$. The *A-SRS* works in rounds. In every single round, all resolved and resolving vertices are taken from G then leave unresolved vertices to the next iteration. The graph established in the $(k-1)th$ iteration is denoted by G^k, where $G^k = G^k - S$, and S is the set of vertices resolved by W in G^{k-1}. It is clear that $G^0 = G$.

UDS. If there are k vertices such that $r(v_1|W) = r(v_2|W) = \ldots = r(v_k|W)$ in G^k ,then all k vertices constitute an undistinguished set, abbreviated as U.

Definition 4. $S_w(u)$ denotes a set of relay vertices on the shortest paths from the vertex $u \in U_i$ to a resolving vertex $w \in W$, that is $S_w(u) = \{x|x \in sp(u,w), u \in U_i, w \in W\}$. $S_w(u)$ denotes all relay vertices on the shortest paths from $u \in U_i$ to every resolving vertex in W, that is $S_W(u) = \bigcup\limits_{w \in W} S_w(u)$. All relay vertices on the shortest paths from vertices in U_i to every vertex in W are denoted as $S(U_i)$, that is $S(U_i) = \bigcup\limits_{u \in U_i} S_W(u)$.

2.2 Problem Definition

This paper aims to solve the problem of positioning faulty routers in $WMNs$. To this end, a highly efficient algorithm $A - SRS$ is proposed to position mesh routers by designating every one of them a unique coordinate.

We assume $A - SRS$ considers the actions and procedures taken by the detecting routers and the computational "center" under the assumption that the topology of the network will not change. And the network is deployed with a computational "center" such that every router acquaints with the geographical position of the "center". The reason for disregarding adding or removing mesh routers is as follows. To the best of our knowledge, all positioning algorithm will set up beacons, as our resolving nodes, to identify routers. However, adding or removing new nodes will significantly increase the traffic load that will eventually jeopardise the primary function of mesh network. Thus we assume the topology of the network will remain unchanged.

An example of the *A-SRS* distinguishes all vertices by assigning all vertices different coordinates is presented in Figure 2. Note that throughout this paper all private neighbors are represented in triangles, all dominators are shown in squares and the other vertices are denoted as circles.

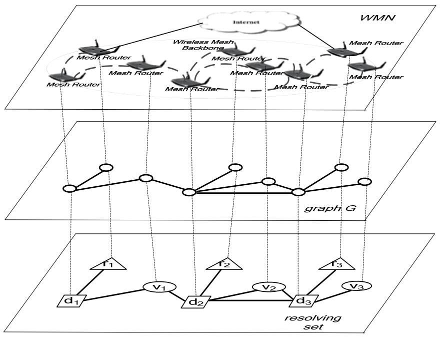

Figure 2: A *WMN* uniquely identified by the *A-SRS*.

In Figure 2, a wireless mesh network is mapped into a graph G. In graph G, nine vertices represent all nine mesh routers, while the edges of G stand for the wireless link between mesh routes. Then a resolving set is built by the *A-SRS*. It is clear that r_1, r_2 and r_3 are private neighbors of dominators d_1, d_2 and d_3 respectively, all of which are carefully chosen by the *A-SRS* as a resolving set W. Different coordinates of remaining vertices are listed as follows:

$$r(v_1|r_1, r_2, r_3) = (2, 2, 3), r(v_2|r_1, r_2, r_3) = (4, 2, 2),$$
$$r(v_3|r_1, r_2, r_3) = (5, 3, 2), r(d_1|r_1, r_2, r_3) = (1, 3, 4),$$
$$r(d_2|r_1, r_2, r_3) = (3, 1, 2), r(d_3|r_1, r_2, r_3) = (4, 2, 1).$$

Since there is no other resolving set with less cardinality, W is the optimal solution of resolving set problem for this graph.

3 *A-SRS*

In this session, we introduce the algorithm, denoted by *A-SRS*, on searching resolving set based on private neighbor of dominating set. Actually the *A-SRS* consists of two important components, A-$RSDS$ and A-$RSLR$, both of which collaborate to build a resolving set for graph G.

3.1 Algorithm Description

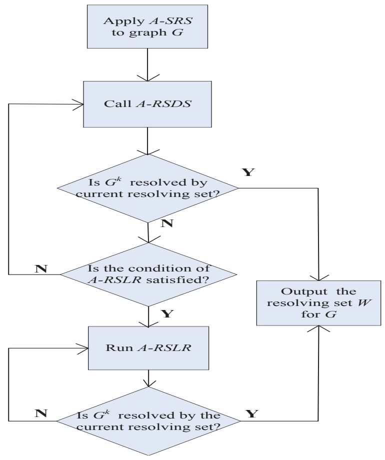

Figure 3: The framework of the A-SRS.

3.1.1 *A-RSDS* (The algorithm of searching resolving set based on dominating set)

The A-SRS works in rounds. In each round A-$RSDS$ is applied to locate private neighbors of dominating set as resolving vertices. To be more specific, A-$RSDS$ is designed to choose private neighbors of dominators that can resolve as many vertices as possible to join the resolving set in each round. Then all resolving vertices and resolved ones are taken off from the current graph G^k and leave the induced sub-graph G^{k+1} of unresolved vertices to the next round. Note that a set of vertices with the same coordinate form an undistinguished set. Obviously $V(G^{k+1})$ consists of all undistinguished sets. The distance between two vertices remains the same in every induced graph. It is clear that repeating these steps aforementioned can possibly resolve G. However redundant resolving vertices could be chosen. To pursue the minimal size of resolving set, a local route information based algorithm (A-$RSLR$) is devised.

3.1.2 *A-RSLR* (The algorithm of searching resolving set based on local route information)

A-RSLR is employed to deal with the unresolved vertices when its condition is satisfied. Based on the length of the shortest paths (which are from the unresolved vertices to all resolving vertices) and the ID of relay vertices, *A-RSLR* carefully chooses some relay vertices on those shortest paths as resolving vertices to resolve graph G.

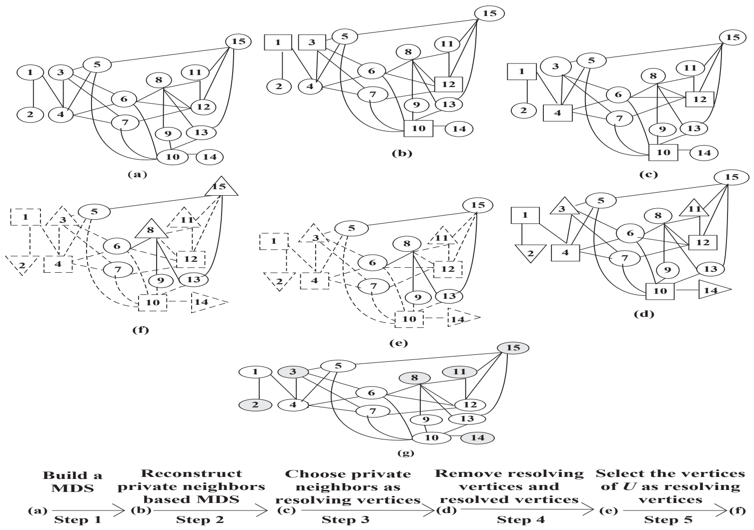

Figure 4: An example of how the *A-SRS* resolves a graph G.

In Figure 4, the algorithm *A-SRS* begins with the process described in Figure 4(a), which consists of 15 vertices, ends with a resolving set W composed by 6 grey vertices in Figure 4(g). Specifically, the processes of *A-RSDS* are presented in Figure 4(a), 4(c), 4(d) and 4(e) by taking the following steps: building an MDS, reconstructing a private neighbors-based MDS, choosing private neighbors as resolving vertices and putting all unresolved vertices into the next round of iteration. Then *A-RSLR* starts with Figure 4(e) and ends in the resolving set of G shown in Figure 4(f) and 4(g). Now we explain these five steps one by one in details.

Let the graph G consist of 15 vertices and the *A-SRS* start with Figure 4(a). The processes of *A-RSDS* include the following four steps presented from Figure 4(b) to 4(e). Note that all vertices taken off from graph G^k and the disconnected links resulted from the removal of corresponding vertices are presented in dash lines.

- Step 1:Build an MDS on G, and the vertices 1, 3, 10 and 12 are chosen as dominators. However, only dominator 3 does not have its own private neighbors because all neighbors of dominator 3 are dominated by other dominators 1, 10 and 12.
- Step 2: Reconstruct MDS by randomly choosing a dominated vertex. For example vertex 4 is chosen as a new dominator to replace vertex 3.
- Step 3: Private neighbors that can resolve more than 3 vertices have the priority to join the resolving set. The vertices 2, 3, 11, 14 and 15, all of which are private neighbors of dominators 1, 4, 10 and 12 respectively, are selected as resolving vertices. We have the resolving set $W = \{1, 4, 10, 12\}$.
- Step 4: Check coordinates of all vertices on current resolving set W. Then take vertices 1, 2, 3, 4, 10, 11, 12 and 14 off G due to their difference in coordinates. There are still 7 unresolved vertices left, all of which constitute three different undistinguished set. Let vertices 5, 6 and 7 comprise U_1, and U_2 is composed by vertices 8 and 15, while vertices 9 and 13 belong to U_3.
- Step 5: We can see, the vertices 15 and 8 are on the shortest path from vertices 5, 6 and 7 to resolving vertex 11. It is quite clear that $15 \in sp(13, 11)$, $15 \notin sp(9, 11)$, $15 \notin sp(7, 11)$, $8 \in sp(6, 11)$, $8 \notin sp(7, 11)$ and $8 \notin sp(5, 11)$. Therefore vertices 8 and 15 are chosen to join the resolving set W.

The resolving set W for graph G that consists of grey vertices is shown in Figure 4(g) with $W = \{1, 4, 8, 10, 12, 15\}$.

3.2 Algorithm and Pseudo-code

Algorithm 1 *A-SRS* (searching resolving set based on private neighbor of dominating set).

Input: G^k.
Output: resolving set W.

```
while (G^k is not resolved by W or the condition of A-RSLR is satisfied) do
    Run A-RSDS
end while
if G^k is resolved by W then then
    Output W
    else run A-RSLR
end if
```

Algorithm 2 *A-RSDS* (resolving set based on private neighbor of dominating set).

Input: G^k.
Output: G^{k+1}
1: Build a MDS on G^k and choose $|P_i|-1$ vertices from every group P_i to join the resolving set W
2: Run $RMDS$ to reconstruct the original MDS to a private neighbor based dominating set
3: Run SRV to select resolving vertices
4: **if** G^k is not resolved by W, then take all resolved and resolving vertices off G^k, update G^k to G^{k+1}
5: **return** G^{k+1}.

Algorithm 3 $RMDS$ (reconstruction of minimum dominating set).

Input: Minimum Dominating Set - MDS.
Output: Minimum Dominating Set based on private neighbors - MDS.
1: **for** all vertices in MDS **do**
2: **if** there is a dominator whose neighbors are dominated by other dominators **then**
3: randomly choose one of its neighbors as a new dominator to replace it.
4: **end if**
5: **end for**
6: **return** new MDS

In algorithm 1, the *A-SRS* employs *A-RSDS* to resolve general graph G. Then *A-RSLR* is applied to locate the minimum resolving set when the condition of *A-RSLR* is satisfied.

In algorithm 2, *A-RSDS* takes four steps to achieve the minimum resolving set.

- Step 1: A polynomial algorithm is used to locate a minimum dominating set for graph G. It is worth to mention that the vertex with maximum degree has the prior to be selected as a dominator. After the MDS is established, $(|P_i| - 1)$ elements are taken from each group P_i to join the resolving set. The reason for choosing $(|P_i| - 1)$ vertices from every group P_i to join the resolving set W is that every pair of vertices $u \in P_i$ and $v \in P_i$ are twins, then u and v share the same coordinate. Intuitively, for each P_i, at least $(|P_i| - 1)$ members should join the resolving set.

- Step 2: $RMDS$ (Reconstruction of Dominating Set), transforms the original dominating set into a new dominating set based on private neighbors.

Algorithm 4 SRV (selection of resolving vertices)

Input: Minimum Dominating Set based on private neighbors - MDS'.
Output: resolving set W.

for all dominators in MDS **do**
    if there is a private neighbor of the dominator which can resolve at least two vertices **then**
        join it to resolving set W
        else randomly choose a private neighbor as a resolving vertex.
    end if
end for
return W

Algorithm 5 *A-RSLR* (searching of resolving set based on local route information)

Input: G^k.
Output: resolving set W.

Let set $S_1 = \emptyset, S_2 = \emptyset, D = \emptyset$
for all vertices of G^k **do**
    if there doesn't exist a $v \in G^k, S_W(v) \neq \emptyset$ **then**
        randomly choose a vertex v to join the resolving set W
        else
        while (every vertex $v \in G^k$, $S_W(v) \neq \emptyset$) **do**
            $S_2 = S_2 \cup \{S_W(v)\}$ and $S_1 = S_1 \cup \{v\}$
            if there exist a pair of elements $X, Y \in S_2$ such that $X \cap Y \neq \emptyset$ **then**
                $S_2 = S_2 \cup \{X - Y\} \cup \{Y - X\} \cup \{X \cap Y\} \cup \{v\}/\{X, Y\}$
            end if
            if G^k is not resolved **then**
                go to 3.
                else record S_1 as a resolving set in $D, D = D \cup \{S_1\}$
            end if
            if there are no more new resolving set recorded in D **then**
                Output the resolving set with minimum cardinality
                else go to 3.
            end if
        end while
    end if
end for
return the minimum resolving set W from D

- Step 3: SRV is responsible for choosing eligible private neighbors as resolving vertices.
- Step 4: Put all unresolved vertices into the next round of iteration and builds undistinguished sets with the vertices that have the same coordinates.

In Algorithm 3, line 2 illustrates that how to deal with a dominator without private neighbors. For example there is a vertex v whose neighbors are dominated by vertices u and w, $N(v) \subseteq N(u) \bigcup N(w)$ and v is not adjacent to neither u nor w (otherwise it is a contradiction to the definition of MDS). Then choose a dominated vertex $x \in N(v)$ to replace v as a new dominator that has its own private neighbor v. And the size of the new MDS remains the same.

Algorithm 4 decides what kind of private neighbors can be selected as a member of resolving set. Furthermore, if a vertex can resolve more than three vertices, it has the priority over all candidates to join resolving set.

In Algorithm 5, it is worth to mention that S_2 is a set $S_W(v)$, where v is in G^k and W is the resolving set of G^{k-1}. In order to explain *A-RSLR*, we give the following example.

Suppose in Figure 4(g), vertices 5, 6 and 7 comprise U_1. U_2 is composed by vertices 8 and 15. And vertices 9 and 13 constitute U_3. From our observation, there are $S_{11}(8) = \{\{6\}, \{9, 13\}\}$ and $S_{11}(15) = \{\{13\}, \{5\}\}$. Then it is clear that $S_1 = \{8, 15\}$ and $S_2 = S_{11}(8) \cup S_{11}(15) = \{\{6\}, \{9, 13\}, \{13\}, \{5\}\}$. Therefore there exist a pair of elements $X = \{9, 13\}$ and $Y = \{13\}$ such that $X \cap Y \notin \emptyset$, we get $S_2 = S_2 \cup \{\{X - Y\}\} \cup \{\{Y - X\}\} \cup \{\{X \cap Y\}\}$ in set operations. Eventually the set $S_2 = \{\{6\}, \{9\}, \{13\}, \{5\}\}$ implies vertices 8 and 15 resolve vertices 8, 15, 6, 9 and 13.

Note: Every time we check the coordinates of all vertices in G^k, we only look for the shortest paths from all unsolved nodes to the resolving vertices newly chosen to join the W. Since the information of the shortest path (from unresolved vertices to the existing resolving ones) including the path length and the IDs of relay vertices is stored by every vertex, the communication cost can be significantly reduced.

4 Theoretical Analysis

In this section, we are in a position to prove that the *A-SRS* establishes a resolving set with the cardinality approximating to $n \times \left(1 - \left(1 - \frac{1+\ln(\delta(G)+1)}{1+\delta(G)}\right)^k\right)$ for a graph G.

Theorem 4.1 Let U be an undistinguished set with $3 \leq |U| \leq 4$ and $N(v_i) \cap N(v_{i+1}) \neq \emptyset$ for any pair of vertices v_i, $v_{i+1} \in U, i < |U|$. If there only exist a pair of vertices $x, y \in U$ such that $d(x, y) = 3$, then x can resolve U.

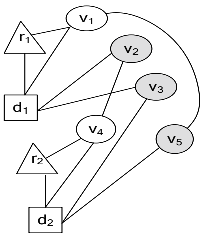

Figure 5: Three vertices resolved by one vertex.

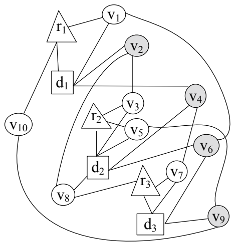

Figure 6: Four vertices resolved by one vertex.

Proof: According to the assumption, we are going to prove this theorem in two cases:

Case 1: U is of order 3. Set v_1, v_2 and v_3 constitute U. Take into account that $N(v_i) \cap N(v_{i+1}) \neq \emptyset$ for any pair of vertices v_i, $v_{i+1} \in U$, where $i \leq 2$, we have $N(v_1) \cap N(v_2) \neq \emptyset$, $N(v_2) \cap N(v_3) \neq \emptyset$ and $N(v_1) \cap N(v_3) = \emptyset$, so $d(v_1, v_2) = 2, d(v_2, v_3) = 2$ and $d(v_1, v_3) \geq 3$. If there exist only a pair of vertices $x \in N(v_1)$ and $y \in N(v_3)$ such that $xy \in E(G)$, then $d(v_1, v_3) = 3$. Hence v_1 can resolve v_1, v_2 and v_3.

Case 2: U is of order 4. Similar to case 1, we have $d(v_1, v_2) = 2$, $d(v_2, v_3) = 2$, $d(v_1, v_3) \geq 3$ and $d(v_1, v_4) \geq 3$. If there only exist a pair of vertices $x \in N(v_1)$ and $y \in N(v_3)$ such that $xy \in E(G)$, then $d(v_1, v_3) = 3$ and $d(v_1, v_4) = 4$. Hence v_1 can resolve v_1, v_2, v_3 and v_4.

Figure 5 and 6 give two examples of how a vertex resolves at least three vertices.

In Figure 5, we assume r_1 and r_2 are resolving vertices, d_1 and d_2 are dominators. It is clear that v_1 and v_4 are resolved by resolving vertices, v_2, v_3 and v_5 remain unresolved. Due to $d(v_2, v_3) = 2$, $d(v_2, v_5) = 3$ and $d(v_2, v_3) = 0$, the vertex v_2 can be chosen as a resolving vertex. The coordinates of v_2, v_3 and v_5 are listed as follows:

$$r(v_2|r_1, r_2, v_2) = (2, 2, 0),$$
$$r(v_3|r_1, r_2, v_2) = (2, 2, 2),$$
$$r(v_5|r_1, r_2, v_2) = (2, 2, 3).$$

Therefore v_2 resolves v_2, v_3 and v_5.

In Figure 6, we assume r_1, r_2 and r_3 are resolving vertices, d_1, d_2 and d_3 are dominators. It is clear that v_2, v_4, v_6 and v_9 are not resolved by resolving set. The vertex v_2 can be chosen as a resolving vertex because $d(v_2, v_4) = 2$, $d(v_2, v_6) = 3$ and $d(v_2, v_9) = 4$. The coordinates of v_2, v_4, v_6 and v_9 are listed as follows:

$$r(v_2|r_1, r_2, r_3, v_2) = (2, 2, 2, 0),\ r(v_4|r_1, r_2, r_3, v_2) = (2, 2, 2, 2),$$

$$r(v_6|r_1, r_2, r_3, v_2) = (2, 2, 2, 3),\ r(v_9|r_1, r_2, r_3, v_2) = (2, 2, 2, 4).$$

Without loss of generality, we need to consider the case that if an undistinguished set is comprised by a set of vertices all of which do not share common neighbors. Theorem 4.2 illustrates the case how many vertices are required to resolve the undistinguished set.

Theorem 4.2 Let U be an undistinguished set with $|U| \geq 3$ and $N(u) \cap N(v) = \emptyset$ for any pair of vertices u, $v \in U$. If there exist only one pair of vertices x,$y \in U$ such that $d(x, y) = 3$ then x can resolve at least two vertices.

Proof: We distinguish the following two cases to prove the theorem.

Case 1: U is of order 3. Suppose that vertices v_1, v_2 and v_3 constitute U. Take into account that $N(u) \cap N(v) = \emptyset$ for any pair of vertices $u, v \in U$, we have $N(v_1) \cap N(v_2) = \emptyset$, $N(v_2) \cap N(v_3) = \emptyset$ and $N(v_1) \cap N(v_3) = \emptyset$. Thus $d(v_1, v_2) \geq 3$, $d(v_2, v_3) \geq 3$ and $d(v_1, v_3) \geq 3$. If there exist only one pair of vertices $x \in N(v_1)$ and $y \in N(v_2)$ such that $xy \in E(G)$, then $d(v_1, v_2) = 3$ and $d(v_1, v_3) \geq 4$. Hence v_1 can resolve v_1, v_2 and v_3. In this case v_1 resolves U.

Case 2: U is of order at least 4. Similar to case 1, we have $d(v_1, v_2) \geq 3$, $d(v_2, v_3) \geq 3$, $d(v_1, v_3) \geq 3$ and $d(v_1, v_i) \geq 3$, where $i \geq 4$. If there exist only one pair of vertices $x \in N(v_1)$ and $y \in N(v_2)$ such that $xy \in E(G)$, then $d(v_1, v_2) = 3$ and $d(v_1, v_i) \geq 4$ for all i, $i \geq 4$. Therefore v_1 can resolve v_1 and v_2.

Figure 7 gives an example of Theorem 4.2 that an undistinguished set consists of three vertices all of which do not share common neighbors can be resolved by one vertex.

In Figure 7, suppose r_1, r_2, and r_3 are resolving vertices, d_1, d_2 and d_3 are dominators. It is easy to check that v_4, v_5 and v_6 are resolved by resolving vertices, but v_1, v_2 and v_3 remain unresolved. The vertex v_2 can be chosen as a resolving vertex because $d(v_2, v_1) = 3$ and $d(v_2, v_3) = 4$. The coordinates of v_1, v_2 and v_3 are listed as follows:

$$\begin{aligned} r(v_1|r_1, r_2, r_3, v_2) &= (2, 2, 2, 0), \\ r(v_2|r_1, r_2, r_3, v_2) &= (2, 2, 2, 3), \\ r(v_3|r_1, r_2, r_3, v_2) &= (2, 2, 2, 4). \end{aligned}$$

Actually by Theorems 4.1 and 4.2, if a private neighbor can resolve at least three vertices, then it can be chosen as a member of the resolving set.

Theorem 4.3: Let U_i be an undistinguished set in G^k, $1 \leq i \leq m$. A vertex $v \in U_i$ can resolve $(m-1)$ vertices if for every U_j, where $i \neq j$, there exist two vertices $u \in U_j$ and $w \in W$ such that v only appears on the shortest paths from u to w.

Proof: According to assumption, let U_i be an undistinguished set in G^k of order n, $1 \leq i \leq n$. By the definition of an undistinguished set, for any pair of vertices $u, v \in U_i$, there exists some $w \in W$ such that $d(u, w) = d(v, w)$.

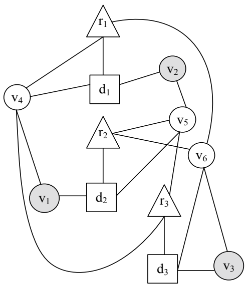

Figure 7: Three vertices resolved by one vertex.

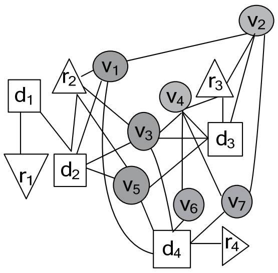

Figure 8: Three undistinguished sets resolved by one undistinguished set.

Let v be a vertex of U_1. If for every U_j, $j > 1$, there exists a vertex $u \in U_j$ such that v only appears on the shortest paths from u to a specific resolving vertex $w \in W$ only. That is $v \in sp(u, w)$, but $v \notin sp(x, w)$ for any vertex $x \in U_j$. Thus $d(u, w) = d(u, v) + d(v, w)$ and $d(x, w) < d(x, v) + d(v, w)$. Due to $d(u, w) = d(x, w)$, it is clear that $d(u, v) < d(x, v)$. Therefore $v \in U_i$ resolves $u \in U_j$. Without loss of generality, v can resolve one vertex in each U_j, $1 < j \leq m$, which implies that v can resolve $(m - 1)$ vertices.

Theorem 4.4: Let U_i be an undistinguished set in G^k, $1 \leq i \leq m$. U_1 can resolve G^k, if every vertex $v \in U_1$ can resolve a vertex $u \in U_j$ for all j, $1 < j \leq m$.

The proof is similar to that of Theorem 4.3.

It is worth to mention that Theorem 4.3 shows the case that a vertex of an undistinguished set can resolve only one vertex of the other undistinguished sets that at most $m - 1$ vertices can be resolved, where m denotes the number of undistinguished sets. Thus Theorem 4.3 is employed by *A-RSLR* to reduce the chance of redundant resolving vertices been chosen. Compared with Theorem 4.3, Theorem 4.4 clarifies a particular case that an undistinguished set can resolve all the other undistinguished sets. Figure 8 gives an example of the case mentioned above.

In Figure 8, the chosen resolving vertices are r_1, r_2, r_3 and r_4. And only the dominators d_1, d_2, d_3 and d_4 are resolved by these resolving vertices. Let unresolved vertices v_1, v_3 and v_5 consist of U_1. U_2 is composed by v_2 and v_4. The vertices v_6 and v_7 constitute U_3. It is clear that $d(v_1, v_4) = 3$, $d(v_3, v_4) = 1$, $d(v_5, v_4) = 2$, $d(v_6, v_4) = 1$, $d(v_7, v_4) = 1$, $d(v_5, v_2) = 2$, $d(v_1, v_2) = 1$, $d(v_3, v_2) = 2$, $d(v_6, v_2) = 3$ and $d(v_7, v_2) = 1$. Therefore v_2 and v_4 can resolve v_2, v_4, v_1, v_3, v_4, v_6 and v_7. The coordinates of which are listed as follows:

$$r(v_1|r_1, r_2, r_3, r_4, v_2, v_4) = (3, 1, 2, 2, 1, 3),$$
$$r(v_3|r_1, r_2, r_3, r_4, v_2, v_4) = (3, 1, 2, 2, 3, 1),$$
$$r(v_5|r_1, r_2, r_3, r_4, v_2, v_4) = (3, 1, 2, 2, 2, 2),$$

$$r(v_6|r_1, r_2, r_3, r_4, v_2, v_4) = (5, 3, 2, 2, 3, 1),$$
$$r(v_7|r_1, r_2, r_3, r_4, v_2, v_4) = (5, 3, 2, 2, 1, 1).$$

In other words, U_2 resolves U_1, U_2 and U_3.

Note that an undistinguished set U_1 is called "adjacent" to other two undistinguished sets U_2 and U_3, if there exist four vertices $u, v \in U_1$, $x \in U_2$ and $y \in U_3$ such that $ux, vy \in E(G)$. Actually Figure 8 not only gives an example that two distinct undistinguished sets U_2 and U_3 can be resolved by the other undistinguished set U_1, but also illustrates the existence of U_1 been "adjacent" to U_2 and U_3. And this is the premise of *A-RSLR*. The formal proof on the existence of this case is given by Theorem 4.4.

Definition 5 Let v be a dominator of G and a set of vertices dominated by v constitute an undistinguished set U. If a private neighbor $r \in N(v)$ is a resolving vertex, then we call r the U's private resolving vertex.

In Figure 8, r_2, r_3 and r_4 are private resolving vertices of U_1, U_2 and U_3 respectively.

Theorem 4.5: Let U_1, U_2 and U_3 be three undistinguished sets of G^k. And r_i is the private neighbor of U_i respectively, $1 \leq i \leq 3$. If there exist three vertices $v \in U_1$, $u \in U_2$ and $w \in U_3$ such that $d(v, r_2) = d(v, r_3) = 2$, $u \in sp(v, r_2)$ and $w \in sp(v, r_3)$, then U_1 is "adjacent " to U_2 and U_3.

Proof: According to the assumption, G^k have three undistinguished sets U_1, U_2 and U_3, the private neighbor of which are r_1, r_2 and r_3 respectively. And there exist three vertices $v \in U_1, u \in U_2$ and $w \in U_3$ such that $d(v, r_2) = d(v, r_3) = 2$, $u \in sp(v, r_2)$ and $w \in sp(v, r_3)$. Since $d(v, r_2) = d(v, u) + d(u, r_2) = 2$, $d(v, r_3) = d(v, w) + d(w, r_3) = 2$, $d(u, r_2) \geq 1$, and $d(w, r_3) \geq 1$, we have $d(v, u) = d(v, w) = 1$. It is clear that U_1 is "adjacent" to U_2 and U_3.

Before we proceed to prove the upper bound of the *A-SRS*, the two important theorems about the domination number and minimum degree of introduced subgraph will be presented.

Theorem 4.6: [Clark et al., 1998] $\gamma(G) \leq \frac{1+\ln(\delta(G)+1)}{1+\delta(G)} \times n$, where n denotes the size of $V(G)$

In each round, the vertex with maximum degree is selected as a dominator in MDS. For every dominator in $d \in MDS$, a private neighbor $p \in N(d)$ is chosen as a resolving vertex when p can resolve at least two vertices or p is of maximum degree among all private neighbors of the dominator d. Since all unresolved vertices enter into the next round of iteration, irrespective of the resolving and resolved vertices, it is obviously that $\delta(G^0) = \delta(G^1) = \ldots = \delta(G^k)$. Therefore the following theorem holds.

Theorem 4.7:
$$(\prod_{i=1}^{k-1}(1 - \frac{1+\ln\left(\delta(G^{i-1})+1\right)}{1+\delta(G^{i-1})})) \times \frac{1+\ln\left(\delta(G^{k-1})+1\right)}{1+\delta(G^{k-1})}$$

$= (1 - \frac{1+\ln(\delta(G)+1)}{1+\delta(G)})^{k-1} \times \frac{1+\ln(\delta(G)+1)}{1+\delta(G)}$, $\delta(G) = \delta(G^0)$

Theorem 4.8: The algorithm *A-SRS* establishes a resolving set W with the size approximating to $n \times \left(1 - \left(1 - \frac{1+\ln(\delta(G)+1)}{1+\delta(G)}\right)^k\right)$, where n denotes the number of vertices in G and k stands for the number of rounds the *A-SRS* takes to resolve G.

Proof: For simplicity, we let $f(G^k) = \frac{1+\ln\left(\delta(G^k)+1\right)}{1+\delta(G^k)}$ and $G = G^0$, where $k \geq 1$. Then the theoretical proof on the size of resolving set W is as follows.

When $k = 1$, either one private neighbor of every dominating vertex chosen by *A-RSDS* or one relay vertex located by *A-RSLR* will join the resolving set. By Theorem 4.6, the size of resoling set W is $|G^0| \times f(G^0) = |G^0| \times [1-(1-f(G^0))^1]$.

Suppose in the kth round every chosen private neighbor can not resolve any other vertices in G^k but itself, only $f(G^k) \times |G^k|$ vertices are chosen as resolving vertices and left $(1 - f(G^k)) \times |G^k|$ vertices for the next round.

When $k = 2$, the number of resolving vertices chosen in the first two rounds is approximating to $|G| \times [f(G^0) + (1-f(G^0)) \times f(G^1)]$. By Theorem 4.7, we have $(1 - f(G^0)) \times f(G^1) = (1 - f(G^0)) \times f(G^0)$. Hence, the size of resolving set W is approximate to $|G| \times [f(G^0) + (1 - f(G^0)) \times f(G^1)] = |G| \times [1 - (1 - f(G^0))^2]$.

When $k = 3$, the number of chosen resolving vertices approximates to $|G| \times [f(G^0) + (1 - f(G^0)) \times f(G^1) + [1 - (1 - f(G^0))f(G^1)] \times f(G^2)]$.

By Theorem 4.7, we have

$$(1 - f(G^0)) \times f(G^1) = (1 - f(G^0)) \times f(G^0)$$

and

$$[1 - (1 - f(G^0)) \times f(G^1)] \times f(G^2)] = (1 - f(G^0))^2 \times f(G^0).$$

Therefore in first three rounds, the total number of resolving vertices is approximating to $|G| \times [f(G^0) + (1-f(G^0)) \times f(G^1) + [1-(1-f(G^0)) \times f(G^1)] \times f(G^2)] = |G| \times [1 - (1 - f(G^0))^3]$.

By induction, we have the resolving set with size approximating to $n \times \left(1 - \left(1 - \frac{1+\ln(\delta(G)+1)}{1+\delta(G)}\right)^k\right)$.

Theorem 4.9:

The complicity of the *A-SRS* is $\left(\log_{1-\frac{1+\ln(\delta(G)+1)}{1+\delta(G)}} \frac{1+\delta(G)}{n \times (1+\ln(\delta(G)+1))}\right) + 1$, where n denotes the order of graph G.

Proof: In order to prove the theorem clearly, we let $\frac{1+\ln\left(\delta(G^k)+1\right)}{1+\delta(G^k)} = f(G^k)$. According to Theorem 4.8, if only $f(G^k) \times |G^k|$ vertices are chosen as resolving vertices and $(1 - f(G^k)) \times |G^k|$ vertices are left for the next round of iteration, then in the $(k + 1)$th rounds there are only two unresolved vertices left, both of which constitute an undistinguished set. Either one of these two vertices can be chosen as a resolving vertex. Therefore, there exists some k such that $\prod_{i=1}^{k-1}\left(1-f(G^{i-1})\right)\times f(G^{k-1})\times n=1$. By Theorem 4.7 the following equation $\prod_{i=1}^{k-1}(1-f(G^{i-1}))\times f(G^{k-1})=(1-f(G))^{k-1}\times f(G)$ holds for all k. Hence the complicity of algorithm *A-SRS* is k, where $k=\left(\log_{1-f(G)}\frac{1}{n\times f(G)}\right)+1=\left(\log_{1-\frac{1+\ln(\delta(G)+1)}{1+\delta(G)}}\frac{1+\delta(G)}{n\times(1+\ln(\delta(G)+1))}\right)+1$.

5 Simulation and Analysis

The performance evaluation of the *A-SRS* is analyzed in this section. We compare *A-SRS* with *Pre-ICH* [Hauptmann et al., 2012] in terms of the cardinality of the resolving set.

5.1 Simulation Setup

Our algorithms are simulated using Ns-2.33 for different number of nodes that are deployed randomly over $500m\times 500m$. The number of deployed nodes varies from 50 to 300. In our simulation, the algorithms *A-SRS* with/without *A-RSLR* are both considered and each simulation is run about 30 rounds to get the average of the data. The communication range varies from 100m to 140m and a homogenous network environment is considered.

5.2 Simulation Result

In order to observe the established resolving set, we considered the simulation environment before and after applying *A-SRS*, as shown in Figure 9. Initially, as shown in Figure 9 (a), we randomly deployed 50 nodes with communication radius $r=100m$ over the monitoring region. Then we run the *A-SRS* to create a resolving set.

In Figure 9 (b), the red marked nodes represent the set of resolving nodes. From the figure, it is observed that resolving nodes mostly locate on the margin of the monitoring region and only a few are in the center area.

As to compare the number of resolving nodes established by *A-SRS* and *Pre-ICH*, simulation is run for different numbers of nodes in a fixed communication radius $r=100m$ with the increasing number of deployed nodes (from 50 to 300). For *A-SRS*, the same simulation must be run twice. One is done with *A-RSLR* and the other one is executed without *A-RSLR*. As shown in Figure 10, it is obvious that the average number of resolving nodes increases if more nodes are

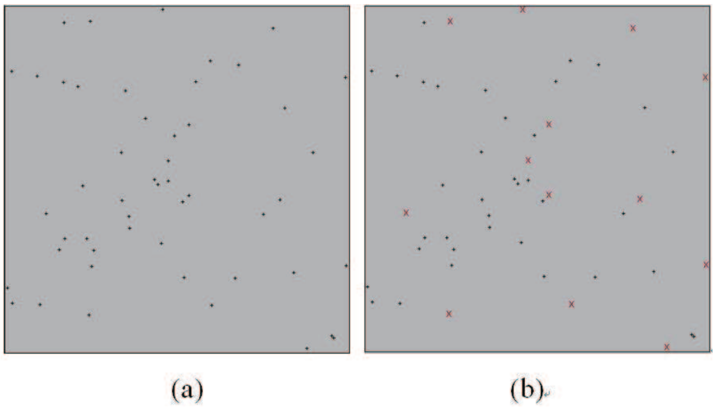

Figure 9: All resolving nodes located by the *A-SRS*.

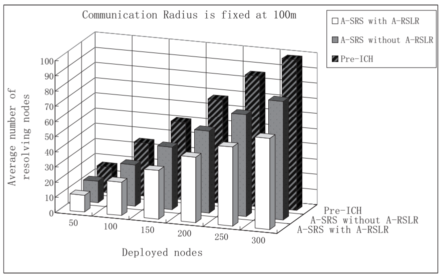

Figure 10: *A-SRS* vs. *Pre-ICH* in average number of resolving vertices with a fixed $r = 100m$.

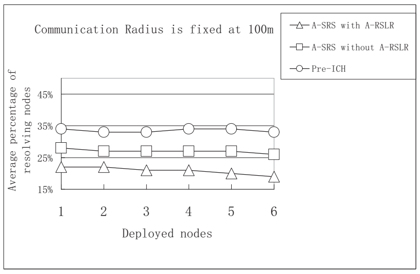

Figure 11: *A-SRS* vs. *Pre-ICH* in average number of resolving vertices with a fixed $r = 100m$.

deployed. For every number of deployed nodes, *A-SRS* with *A-RSLR* always locates the minimum number of resolving nodes while the maximum number of resolving nodes is discovered by *Pre-ICH*.

As shown in Figure 11, it is observed that for every number of deployed nodes nearly 20% of which are chosen as resolving nodes by *A-SRS* with *A-RSLR* compared with 30% by *A-SRS* and 35% by *Pre-ICH*.

When the number of deployed nodes is fixed at 100, the average number of resolving nodes is simulated in different communication radius from 100m to 140m.

As shown in Figure 12, it is observed that with the growth of communication radius, the number of resolving nodes increases gradually. However for every communication radius, *A-SRS* with *A-RSLR* always chooses the minimum number of resolving nodes while *Pre-ICH* finds the largest number of resolving nodes.

As shown in Figure 13, it is analyzed that with the communication radius

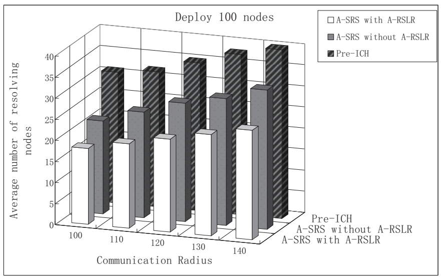

Figure 12: *A-SRS* vs. *Pre-ICH* in average number of resolving set with 100 nodes deployed.

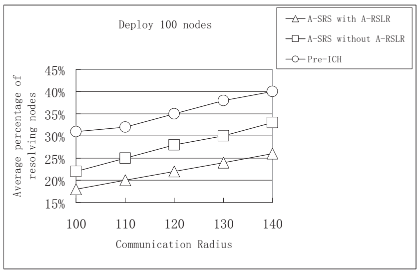

Figure 13: *A-SRS* vs. *Pre-ICH* in average percentage of resolving nodes with 100 nodes deployed.

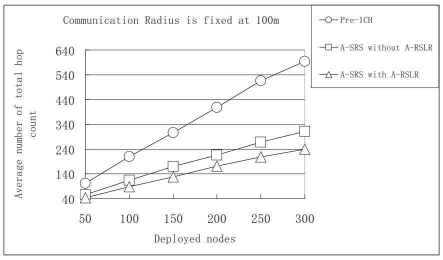

Figure 14: *A-SRS* vs. *Pre-ICH* in average number of total hop count with a fixed r=100m.

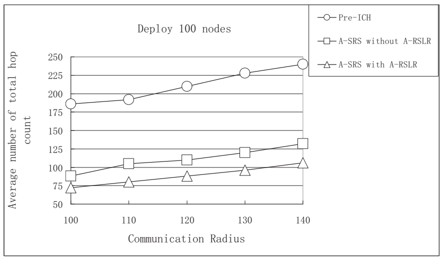

Figure 15: *A-SRS* vs. *Pre-ICH* in average number of total hop count with 100 nodes deployed.

growth from 100m to 140m, *A-SRS* with *A-RSLR* designates 18% up to 28% of total deployed nodes as resolving nodes, compared with 30% to 40% of *Pre-ICH* and 20% to 35% of *A-SRS* without *A-RSLR*.

Note that the locating time is always considered as an important metric for the efficiency of faulty nodes detection. Therefore all malfunctioned nodes should be located in a short time. As we mentioned in section 1.3, a router needs to communicate with all detecting routers before it gets pinpointed. Thus the total distance measured by hops from a router to all detecting ones decides how fast a router is located. As shown in Fig 14 and 15, *A-SRS* with/without *A-RSLR* require less locating time than *Pre-ICH* in terms of total distance.

5.3 Discussion

When the number of deployed nodes is fixed, Figure 12 and 13 show that more resolving nodes will be chosen with the growth of communication radius. While the communication radius is fixed, Figure 10 and 11 show that with the number of deployed nodes grows, there are more resolving nodes chosen. If we take the corresponding graph model into account, both situations above will eventually result in a complete graph G, then more nodes are chosen as resolving nodes. Observed from Figure 10, 11, 12 and 13, the *A-SRS* without *A-RSLR* exceeds *Pre-ICH* in less number of resolving nodes. And the *A-SRS* with *A-RSLR* performs even better that almost two thirds of total resolving nodes discovered by *Pre-ICH* is found by *A-SRS* with *A-RSLR*. It is worth to mention that since *A-SRS* is a centralized algorithm, it is applicable to all kinds of networks that can support centralized algorithms and it does not rely on any particular WMN features.

6 Conclusions and Future Works

The problem of positioning on faulty nodes in all kinds of complex networks has drawn great attention. In this paper, we focus on the detection of malfunctioned routers in Wireless Mesh Networks(*WMN*s) with the characteristic of small worlds. A resolving set based faulty router detection algorithm - *A-SRS* is proposed for *WMN*s. The *A-SRS* employs two important components *A-RSDS* and *A-RSLR* to discover malfunctioned routers such that the efficiency of recovery is ensured. Meanwhile the *availability* that network services operate properly and tolerate failures is guaranteed. Simulation results show that the *A-SRS* deploys less detecting routers than the other strategies in the size of resolving set. Here are some open issues that will be addressed in the future.

Mobile computing is able to provide a seamless service to the user whenever and wherever it is needed irrespective of users' movement. Because of its unique advantages, there is a wide range of potential mobile cloud applications such as the image processing, sensor data applications, the querying and the crowd computing. Among these applications, there are two scenarios of crowd computing, such as "Lostchild" and "Disaster relief' [Fernando et al., 2013], both of which require the location information. Therefore, how to apply the resolving set based algorithm to achieve a better positioning service is worth to study.

Acknowledgment

The authors wish to thank National Natural Science Foundation of China (NO. 61072080, U1405255). Fujian Normal University Innovative Research Team (No. IRTL1207). The Natural Science Foundation of Fujian Province (Nos. 2013J01221, 2013J01222, J01223). The Education Department of Fujian Province science and technology project (JB11027).

References

[Beerliova et al., 2006] Beerliova, Z., Eberhard, F., Erlebach, T., Hall, A., Hoffmann, M., Mihal'ak, M., and Ram, L. S. (2006). Network discovery and verification. *Selected Areas in Communications, IEEE Journal on*, 24(12):2168–2181.

[Bella, 2008] Bella, G. (2008). What is correctness of security protocols? *J. UCS*, 14(12):2083–2106.

[Clark et al., 1998] Clark, W. E., Fisher, D., Shekhtman, B., and Suen, S. (1998). Upper bounds for the domination number of a graph. *preprint*.

[Ding et al., 2005] Ding, M., Chen, D., Xing, K., and Cheng, X. (2005). Localized fault-tolerant event boundary detection in sensor networks. In *INFOCOM 2005. 24th Annual Joint Conference of the IEEE Computer and Communications Societies, 13-17 March 2005, Miami, FL, USA*, pages 902–913.

[Fernando et al., 2013] Fernando, N., Loke, S. W., and Rahayu, W. (2013). Mobile cloud computing: A survey. *Future Generation Computer Systems*, 29(1):84–106.

[Guo et al., 2009] Guo, S., Zhong, Z., and He, T. (2009). FIND: faulty node detection for wireless sensor networks. In *Proceedings of the 7th International Conference on Embedded Networked Sensor Systems, SenSys 2009, Berkeley, California, USA, November 4-6, 2009*, pages 253–266.

[Gustafsson and Gunnarsson, 2005] Gustafsson, F. and Gunnarsson, F. (2005). Mobile positioning using wireless networks: possibilities and fundamental limitations based on available wireless network measurements. *Signal Processing Magazine, IEEE*, 22(4):41–53.

[Halldórsson et al., 2001] Halldórsson, B. V., Halldórsson, M. M., and Ravi, R. (2001). On the approximability of the minimum test collection problem. In *AlgorithmsESA 2001*, pages 158–169. Springer.

[Hauptmann et al., 2012] Hauptmann, M., Schmied, R., and Viehmann, C. (2012). Approximation complexity of metric dimension problem. *J. Discrete Algorithms*, 14:214–222.

[Hoffmann and Wanke, 2013] Hoffmann, S. and Wanke, E. (2013). Metric dimension for gabriel unit disk graphs is np-complete. *CoRR*, abs/1306.2187.

[Khuller et al., 1996] Khuller, S., Raghavachari, B., and Rosenfeld, A. (1996). Landmarks in graphs. *Discrete Applied Mathematics*, 70(3):217–229.

[Mellado and Rosado, 2012] Mellado, D. and Rosado, D. G. (2012). An overview of current information systems security challenges and innovations. *J. UCS*, 18(12):1598–1607.

[Patwari et al., 2005] Patwari, N., Ash, J. N., Kyperountas, S., Hero, A. O., Moses, R. L., and Correal, N. S. (2005). Locating the nodes: cooperative localization in wireless sensor networks. *Signal Processing Magazine, IEEE*, 22(4):54–69.

[Slater, 1975] Slater, P. J. (1975). Leaves of trees. *Congr. Numer*, 14(549-559):37.

[Taleb et al., 2009] Taleb, A., Pradhan, D., and Kocak, T. (2009). A technique to identify and substitute faulty nodes in wireless sensor networks. In *Sensor Technologies and Applications, 2009. SENSORCOMM '09. Third International Conference on*, pages 346–351.

[Uzunov et al., 2012] Uzunov, A. V., Fernández, E. B., and Falkner, K. (2012). Engineering security into distributed systems: A survey of methodologies. *J. UCS*, 18(20):2920–3006.

[Verma et al., 2011] Verma, C. K., Reddy, T. B., Manoj, B. S., and Rao, R. R. (2011). A realistic small-world model for wireless mesh networks. *IEEE Communications Letters*, 15(4):455–457.

[Zhou and Haas, 1999] Zhou, L. and Haas, Z. J. (1999). Securing ad hoc networks. *IEEE Network*, 13(6):24–30.