IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received January 12, 2019, accepted January 31, 2019, date of publication February 7, 2019, date of current version February 22, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2897339

Shortest Path Routing With Risk Control for Compromised Wireless Sensor Networks

NA WANG AND JIAN LI
School of Computer Science, Beijing University of Posts and Telecommunications, Beijing 100876, China
Corresponding author: Jian Li (lijian_beiyou@163.com)

This work was supported by the National Natural Science Foundation of China under Grant U1636106.

ABSTRACT In wireless sensor networks, it is common that adversaries capture some nodes to intercept, tamper, or drop valuable packages. We can employ reputation systems to identify the compromised nodes (CNs). In this paper, the areas that cover a set of dense CNs are called compromised regions (CRs) and apparently, they are a greater threat to the networks than single CNs. To defend against the attack of CRs, we design a secure shortest path routing algorithm (SPRA) to deliver packages properly around, rather than through, the CRs. Specifically, a source node first computes the shortest geometric path to the sink node without crossing any CRs and then decides agent nodes along with the path by a set of virtual locations in an indirect way. At last, a sophisticated mechanism is designed based on geographic information to guarantee that the packages can be delivered in a relay manner between the agent nodes until they are transmitted to the sink node successfully. We conduct a series of experiments to compare our scheme with the dynamic greedy perimeter stateless routing and directed diffusion algorithms. The simulation results illustrate that SPRA can always find the short routing paths while guaranteeing the security of the packages.

INDEX TERMS Wireless sensor networks, secure routing, compromised regions, energy-efficiency.

I. INTRODUCTION

Wireless sensor networks (WSNs) are composed of a large number smart sensor nodes, which are of strictly limited power, capabilities of computation, storage and communication. Recently, WSNs have been widely used in many fields [1]–[3], such as wild animal monitoring, military surveillance, target tracking, forest fire detection and industry security. After being deployed manually or scattered by aircrafts, the smart nodes automatically construct a network connected to the sink node. Each sensor node is responsible for monitoring surrounding environment and data are delivered to the sink node in a one-hop or multi-hop manner. The collected data are transmitted to the remote server by sink node through satellites or internet.

WSNs are vulnerable to many threats, since most of them are deployed in harsh and unattended environments to carry out information collection tasks. Capturing the nodes is a common attack in WSNs and the adversaries can break through the secure systems of the sensor nodes to get the nodes' identities and secret keys. The adversaries can further modify the protocol stacks to intercept, tamper or drop the valuable packages, which poses a great threat to the WSNs. The Compromised Nodes (CNs) can be identified by deploying reputation systems into the networks [4]–[8]. The basic idea of these schemes is to identify the CNs by analyzing their abnormal behaviors. A concomitant challenge is how to deliver the packages intelligently to avoid being captured by the CNs.

Take the scenario in Fig. 1 as an example, the CNs represented by red nodes form three Compromised Regions (CRs) and it is risky to deliver the packages through the CRs. Consequently, a secure and short routing path needs to be designed between the source node and the sink node. Most existing routing algorithms in WSNs cannot tackle the problem properly, because they assume that all the sensor nodes are reliable and capable of storing and transmitting packages. An intuitive solution is to update the topologies of the WSNs from time to time and remove the CNs periodically. Then, the new routing paths can be reconstructed by any existing algorithm. However, this requires extra data transmission and a large amount of energy, which is impractical for WSNs.

With a changed network topology, Trajectory Based Forwarding (TBF) [9] and Greedy Perimeter Stateless

The associate editor coordinating the review of this manuscript and approving it for publication was Matti Hämäläinen.

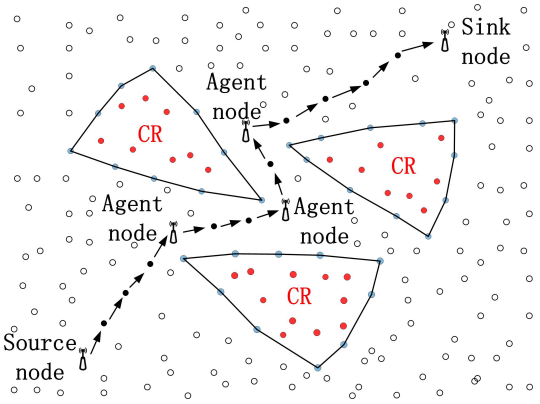

FIGURE 1. Smart routing with compromised regions.

Routing(GPSR) [10] have a large number of redundant steps and they are likely to be trapped in suboptimal results; Directed Diffusion (DD) [11] requires resetting the ''gradients'' for each sensor node and all the routing paths to be recalculated; Low-Energy Adaptive Clustering Hierarchy (LEACH) [12] requires complete reconstruction of clusters and in addition, transmitting packages from the cluster head to the sink node directly is impractical for large WSNs; Dynamic Source Routing protocol(DSR) [13] demands that routing table be updated entirely; The Energy-Balanced Routing Method based on Forward-Aware Factor(FAF-EBRM) [14] has a high computation complexity to reselect next hop for a node, which shortens lifetime of the networks.

Nonetheless, it is imperative to propose a secure, light-weight and energy-efficient routing algorithm against the new threat. In this paper, a secure shortest path routing algorithm named SPRA is designed to overcome the CRs with an acceptable increment in energy consumption. In SPRA, when a package needs to be delivered to the sink node, the source node first searches the shortest geometric path composed of a sequence of connected line segments which cannot cross any CR. This problem can be mapped to the shortest path searching problem over weighted graph and it can be solved completely based on Dijkstra algorithm. Then, the source node locally decides several virtual locations which define the agent nodes in an indirect way. These locations perform an important role in guiding the transmission directions of the packages. At last, a sophisticated mechanism is designed to deliver packages between the agent nodes until the sink node receives the packages successfully. A possible routing path of SPRA is shown in Fig. 1. Simulation results demonstrate that SPRA can always find short routing paths while guaranteeing the security of the packages. A series of experiments are conducted to evaluate the performance of SPRA in terms of reliability and energy efficiency.

The rest of this paper is organized as follows. In Section II, we summarize the existing routing algorithms in WSNs. Some assumptions and preliminary techniques are given in Section III. SPRA is presented in Section IV and its performance is evaluated in Section V. At last, this paper is concluded in Section VI.

II. RELATED WORK

Routing algorithms are of great importance to the WSNs and they are the base of the whole network. As a consequence, many routing algorithms have been proposed in the literature. LEACH algorithm [12] is the most widely used routing algorithm in WSNs in which all the cluster heads are selected in a totally distributed manner. The members in a cluster send their packages to the cluster head in their own time slots and the cluster heads directly communicate with the sink node. Another cluster-based routing algorithm named Hybrid Energy-Efficient Distributed clustering (HEED) is proposed in [15]. The cluster heads are periodically selected according to a hybrid of residual energy and a secondary parameter, such as node proximity to its neighbors or node degree. A better efficient condition for the connectivity of the cluster heads are provided in [16]. Except for cluster-based routing algorithms, some algorithms are designed for the one-layer networks. In the Greedy mode of GPSR [10], the nodes always select the neighbor nearest to the sink node as the next hop whenever possible. Once the Greedy pattern fails, the Face mode recovers and it guarantees that the packages are always deliverable to a node where the Greedy mode can be reused. Certainly, if the destination node is not connected to the network, the Face mode will not return to the Greedy mode and will fail at last. To overcome the coverage holes, an alternative method of face routing [17], [18] is designed by employing relative coordinate systems, avoiding planarizing networks and preserving route optimality properties. Zhu and Nicanfar [19] focus on the 1-dimensional WSNs and propose an opportunistic algorithm to ensure minimum energy cost during data relay and protect the nodes with relative low residue energy. Another classic routing algorithm is directed diffusion [11] in which a prepare phase is needed to construct the gradients of all the nodes to the sink node. The packages are always delivered in the direction with the greatest gradient. Active Trust [20] is designed to defend against the black holes in which a set of detection routes are dynamically built to detect the node trust. However, it cannot be directly employed to defend the CRs considering that this algorithm cannot detect the compromised nodes. An opportunistic routing algorithm named ORR [21] is proposed for duty-cycled WSNs in which both the forwarding cost and residual energy are taken into consideration in the process of selecting next hop. Therefore, ORR can always find a good balance between time-delay of the packages and energy consumption. To our knowledge, most existing routing algorithms are designed to balance the energy consumption, load balance and time delay of the packages and few of them can be directly employed to defend the threat model discussed in this paper.

III. NETWORK MODEL AND REPUTATION SYSTEM

In this section, we first present the network model in Section III.A. Then, the reputation systems are discussed in Section III.B which can be used to identify the CNs. We focus on the beta reputation system which has been widely employed in the literature.

A. NETWORK MODEL

We first assume that a static 2-D network is composed of redundant sensor nodes and each node is capable of executing data transmission, storage and computational operations. All the sensor nodes are homogeneous and have the same communication range R_c. In other words, a pair of nodes can directly communicate with each other if and only if their distance is not larger than R_c. The sink node is assumed to be stronger than the sensor nodes and have adequate energy. The nodes are assumed to be able to locate themselves accurately by GPS devices or other proper manners [22]. Apparently, each node can easily obtain its neighbors' locations through basic communication behaviors. After deployment, the sink node first broadcasts its location to all the nodes in the network and each node can obtain the location of the sink node in time. In the initial, we assume that all the nodes are reliable and hence the reputation system is constructed properly. This is reasonable considering that compromising a set of nodes consumes a lot of time for the adversaries.

B. REPUTATION SYSTEM IN WSNs

Cryptography has been widely used in providing data confidentiality, data integrity and identity certification in WSNs and apparently it is of great importance in constructing secure networks. However, cryptography alone cannot ensure the safety of package delivery process in WSNs [23], [24] and it cannot defend against the attack of CRs proposed in this paper. The sensor nodes in a common WSN are envisioned to be cheap and therefore very unlikely to be equipped with tamper-proof hardware. As a result, sensor nodes can be compromised by an attacker and hence the cryptography materials are compromised. In this case, it is of great importance to identify the CNs and we introduce the reputation system into WSNs. The concept of reputation originated from sociology can be used to overcome the shortcomings of cryptography-based WSNs. Several reputation systems have been proposed in [23] and [25]. In this paper, we focus on the beta reputation system [25] in which reputation R_{ij} is computed by sensor node N_i using beta density function of sensor node N_j's previous actions. For example, sensor node N_i counts the number of good and bad actions of N_j as $r_{ij}$and s_{ij}. Then, N_i records the reputation R_{ij} about node N_j as $R_{ij} = Beta(p|r_{ij}+1, s_{ij}+1)$ and the trust $T_{ij} = E(R_{ij}) = \frac{r_{ij}+1}{r_{ij}+s_{ij}+2}$, where *Beta* represents the Beta distribution which can be expressed by the gamma function Γ as:

$$Beta(p|r_{ij}+1, s_{ij}+1)=\frac{\Gamma(r_{ij}+s_{ij}+2)}{\Gamma(r_{ij}+1)\Gamma(s_{ij}+1)}p^{r_{ij}}(1-p)^{s_{ij}},$$

where $0 \le p \le 1, r_{ij}, s_{ij} \ge 0$. Through the average trust value of the neighbors, we can evaluate the reliability of the sensor nodes and the nodes with a trust value smaller than T are identified as CNs. Further, based on the CNs, we can construct the CRs in the network and at last, SPRA can be employed to defend against the CRs.

IV. SECURE SHORTEST PATH ROUTING WITH COMPROMISED REGIONS

In this section, we first briefly introduce the process of constructing the CRs in Section IV.A. Then, we present the SPRA algorithm in detail. Specifically, SPRA can be divided into three phases: calculating the shortest geometric path, deciding the virtual locations and delivering packages between the agent nodes. The modules of SPRA are provided in Section IV.B, IV.C and IV.D, respectively. Moreover, some open issues and the extended version of SPRA are also discussed.

A. CONSTRUCTION OF THE CRS

SPRA is designed to defend against CRs and the flowchart of constructing CRs is presented in Fig. 2. It can be observed that reputations of the nodes are first calculated based on the reputation system and each node has a reputation list about all the neighbors. Then each node sends the reputations to the sink node periodically and hence all the reputation lists are collected by the sink node to calculate the average reputation of the nodes in the network. To save energy, only the changed reputations are delivered and the sink node uses historical data for the nodes with unchanged reputations. Then the suspicious nodes can be detected by a reputation threshold and they can be divided into clusters by proper clustering algorithms (e.g., k-means and DBSCAN) [26]–[28] based on their locations. In this paper, the similarities between the suspicious nodes are defined as the Euclidean distance. Further, the convex hull of each cluster is constructed by the Graham's scan algorithm [29] and they are the CRs. Each CR is represented by a convex polygon which covers a set of suspicious nodes. In this paper, the sensor nodes on the vertexes of the convex polygons are assumed to be reliable and this is practical considering that the polygons' size can be slightly enlarged to guarantee the reliability of the vertexes when constructing these polygons. At last, the information of CRs is periodically updated by the sink node and only the incremental information of CRs, i.e., the changed information of CRs, is broadcast in the whole network to save energy.

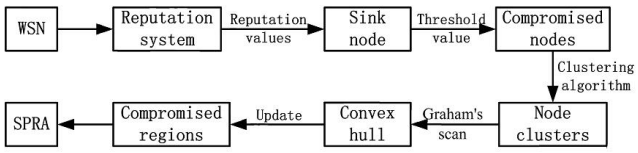

FIGURE 2. The flowchart of constructing the CRs.

B. SHORTEST GEOMETRIC PATH CALCULATION

In this section, we present how to construct the shortest geometric path between source node and sink node without crossing any CRs represented by the convex polygons as discussed in Section IV.A. In fact, the geometric path is employed to guide the transmission of the packages and it decides the basic shape of the final routing path. Apparently, it

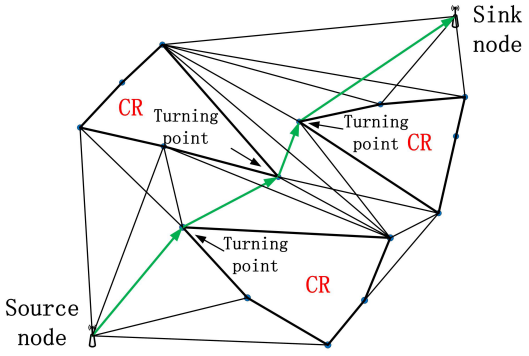

FIGURE 3. Visibility graph and the shortest geometric path.

forms the important base of SPRA. As shown in Fig. 3, each CR is represented by a polygon and we assume that all the vertexes of these polygons, the source node and the sink node forms a node set SN. If any pair of nodes in SN can "observe" each other, we link an edge between them called a *visibility edge*. Note that, two nodes can "observe" each other if and only if the edge between them does not cross any part of the CRs. All the visibility edges compose a graph called *visibility graph*. It has been proved that the shortest geometric path between the source node and the sink node must be one path composed of the visibility edges [30]. Dijkstra algorithm is employed to find the shortest path between the source node and the sink node. As shown in Fig. 3, all the black line segments are visibility edges and the shortest geometric path is the green path with arrows.

It can be analyzed from the above construction process that the geometric path is the shortest path between the source node and the sink node while guaranteeing the security of the packages. If a package is delivered exactly along with the geometric path constructed above, it is less likely that the adversaries can capture it, because the CNs in the CRs cannot access the package. However, this is impractical considering that there is not a set of real sensor nodes which are exactly deployed on this geometric path. Consequently, a routing path strictly along with the geometric path needs to be constructed based on the real sensor nodes and we present the algorithm in the following.

C. SELECTION OF AGENT NODES

It can be observed that several turning points exist on the geometric path and at each turning point, the direction of the path changes with time. In SPRA, the agent nodes act as anchors to turn the directions of the routing paths and apparently, around each tuning point, an anchor is needed. Assume that the geometric path is composed of m line segments, then $m-1$ ordered agent nodes $\{A_1, A_2, \cdots, A_{m-1}\}$ need to be selected around the turning points.

To ensure that the packages do not cross any CR, the agent nodes should be ones far from the CRs. However, this may increase the lengths of routing paths and there is a tradeoff between security and efficiency. It is impractical to pre-assign a set of specific nodes as the agent nodes, because the source node does not know the whole topology of the network. Fortunately, the source node just needs to decide virtual locations and the agent nodes are defined as the nearest nodes to these locations. In this paper, the virtual locations are chosen randomly in the virtual circles. To build a virtual circle around a turning point, we first compute the angular bisector of the turning point, and then the point $\beta R_c(\beta \geq 0)$ away from the turning point in the opposite direction of the CR is defined as the center of the virtual circle as shown in Fig. 4. The radius of the virtual circle is defined as $\gamma R_c(0 \leq \gamma \leq \beta)$ and at last the virtual location is randomly selected from the virtual circle. For each turning point, we decide a virtual location in the same way and obtain a set of ordered virtual locations $\{V_1, V_2, \cdots, V_{m-1}\}$. Note that, the source node does not need to know the locations of agent nodes in the whole routing process. The package delivery mechanism in Section IV.D guarantees that we can always deliver a package to the agent node and this is the most important contribution of this paper.

As presented in Figure 4, β and γ are employed to control the relative position of the turning points and agent nodes. When we increase β and γ, the security of packages increases and more energy is consumed. On the contrary, if we decrease β and γ, the security of packages decreases and energy efficiency increases. It can be observed that there is a trade-off between package security and energy efficiency. Therefore, the network operators need to preset the parameters according to their requirements. SPRA chooses the agent nodes in a random manner to various routing paths. This is important to balance workloads of the nodes and improve security of network.

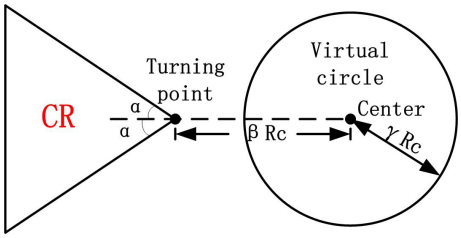

FIGURE 4. Selection of a virtual location.

Though it is almost impossible that the virtual circle intersects with any CR for proper β and γ, if the virtual circle indeed covers a part of any CR, the routing path needs to be totally rebuilt. This is reasonable considering that the final routing path is likely to go through some CRs if the agent nodes are quite close to the CNs. In this case, we can infer that the two CRs must be close to each other. To rebuild the routing paths, the two adjacent CRs are first assumed to be merged and a new CR is generated. Then, a new geometric path needs to be reconstructed based on the updated polygons. At last, we repeat the above process until a set of legal virtual locations is selected successfully.

D. PACKAGE DELIVERY BETWEEN THE SOURCE NODE TO THE SINK NODE

In this section, we discuss how to deliver the packages from the source node to the sink node based on the virtual locations. Specifically, upon the $m-1$ virtual locations $\{V_1, V_2, \cdots, V_{m-1}\}$, a mechanism to deliver the packages between the agent nodes $\{A_1, A_2, \cdots, A_{m-1}\}$ needs to be designed. As discussed previously, $A_i (i = 1, 2, \cdots, m-1)$ is defined as the nearest sensor node to the virtual location $V_i (i = 1, 2, \cdots, m-1)$. Now assume that a package is sent from the source node S to the first agent node A_1. The next hop of the package is always chosen by *Greedy mode* [10] when possible, in which a router always chooses the neighbor nearest to V_1 as the next hop. Meanwhile, the source of the package is updated to the latest node that has received the package. If no neighbor has a smaller distance to V_1 than the router, *Greedy mode* fails and *Face Mode* [9] (i.e., *Perimeter Forwarding* in [10]) recovers. The *Face Mode* uses the "right hand rule" to traverse the polygons in the planar graph of the network, which is constructed by the sensor nodes in a distributed manner [10], until finding the destination node (in this case, the routing process is completed) or an edge E on the graph that intersects the source-destination line (neither the source node nor the destination node locates on E). Then, at either vertex of the edge E, *Greedy mode* can be resumed. If the *Face Mode* cannot find the destination node or the edge E, the routing process fails. In any case, it is guaranteed that the package can be always transmitted successfully to the node locates at V_1 or the polygon that covers V_1 (V_1 can locate on the edge of the polygon). If the *Face Mode* fails on a polygon, the nearest node to V_1 on this polygon (i.e., A_1) is just the agent node and no other node in the network can be closer to V_1. The correctness of this process has been analyzed and proved in [18] . By this step, the source node is replaced by A_1 and V_2 becomes the next destination node. By iterating the above process, the package can be delivered to the sink node at last. Given the source node S, sink node D and a series of ordered virtual locations $\{V_1, V_2, \cdots, V_{m-1}\}$, the pseudo-code of delivering the packages from the source node to the sink node is presented in Algorithm 1.

For convenience, we assume that the nodes in SN' are organized based on stack structure and function SN'.pop() returns the nodes in order. Specifically, function SN'.pop() returns S first and returns D at last. Initially, as shown in line 2 to line 4, the next hop of the package is always selected based on the *Greedy mode* and the package is transmitted to the neighbor nearest to the sink node. As shown in line 5 to line 20, once the *Greedy mode* fails, the package is transmitted based on the *Face Mode* until the *Greedy mode* can be reused. However, if the *Face Mode* cannot return to the *Greedy mode* and it fails, the agent node is found as shown in line 21. Then, the next package delivery process is started.

SPRA can always find short routing paths even in very complex scenarios, because the routing paths always follow the shortest geometric path between the source node and the sink node. As shown in Fig. 5, the three agent nodes twist the routing paths to avoid the packages being delivered through the CRs and this is the most important contribution of this paper. For small β, γ and a set of constant nodes

$$SN' = \{S, V_1, V_2, \cdots, V_{m-1}, D\},$$

SPRA trends to select similar routing paths and the nodes on the paths always act as the intermediate nodes. The SPRA can balance the workload among the sensor nodes by increasing

Algorithm 1 PackageDelivery

Input: $SN' = \{S, V_1, V_2, \cdots, V_{m-1}, D\}$

Output: Delivering a package from S to D along with $\{V_1, V_2, \cdots, V_{m-1}\}$

1: $s \leftarrow SN'$.pop(); $l \leftarrow SN'$.pop();
2: **while** $s \neq l$ and *Greedy mode* does not fail **do**
3: Node s employs *Greedy mode* to select the next hop h of the package; $s \leftarrow h$;
4: **end while**
5: **if** $s = D$ **then**
6: **return;**
7: **else if** $s = l$ **then**
8: $l \leftarrow SN'$.pop(); *go to* step 2;
9: **end if**
10: $s' \leftarrow s$;
11: **while** the package does not come back to s **do**
12: Node s' employ *Face Mode* to select the next hop h of the package; $s' \leftarrow h$;
13: **if** $s' = D$ **then**
14: **return;**
15: **else if** $s' = l$ **then**
16: $s \leftarrow l$; $l \leftarrow SN'$.pop(); *go to* step 2;
17: **else if** an edge E that intersects sl is found (neither s nor l locates on E) **then**
18: $s \leftarrow$ either vertex of E; *go to* step 2;
19: **end if**
20: **end while**
21: $s \leftarrow$ the node nearest to l on the polygon that covers l; $l \leftarrow SN'$.pop(); *go to* step 2;

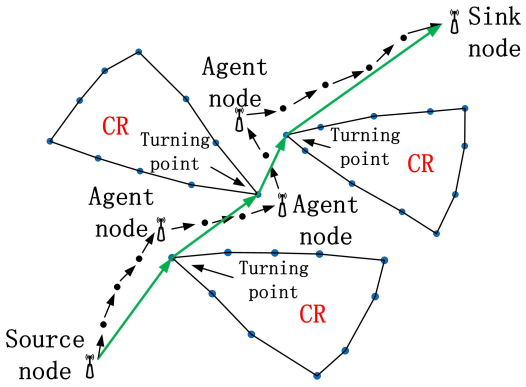

FIGURE 5. Shortest geometric path and shortest routing path.

β and γ, though the range of β and γ are constrained by the distribution of CRs. To further improve the diversity of the routing paths, the source node can first construct a set of geometric paths and sort them based on their lengths. Then a path is selected from the set randomly. In general, a shorter path should be selected with a higher probability. In this way, we can balance the energy load among the sensor nodes and apparently the length of the routing path increases slightly.

Another challenge is how to further improve the security of the packages in the network. In this paper, the routing paths are constructed based on the shortest geometric path and in this case we always attempt to find the shortest path. Though packages are delivered in-between the CRs rather than through them, it is possible that the packages are captured by the CNs when the road between two CRs are narrow. An intuitive approach is constructing the final routing paths based on the second or third shortest geometric path in which the CRs are dispersive. An accompanying problem is how to balance the security of the package delivery and the energy consumption. A key problem is how to measure the risk based on the distribution of the CRs and we will investigate this in our future work.

V. PERFORMANCE EVALUATION

In this section, we evaluate the performance of SPRA. First, the simulation setup is presented in Section V.A. Then, SPRA is compared with GPSR and DD in terms of success rate of package delivery, average hops of routing paths, the amount of data transmission and network lifetime. The simulation results are presented and discussed in Section V.B, V.C, V.D and V.E, respectively. At last, we summarize the performance of SPRA in Section V.F. In simulation, GPSR and DD are divided into dynamic and static patterns based on whether the routing paths are reconstructed when new CRs are generated.

A. SIMULATION SETUP

In simulation, 1000 homogeneous sensor nodes are randomly deployed in a $100m \times 100m$ square plane and R_c of each node is set to $7m$. The source node is located at a corner of the square with coordinate (0,0) and the sink node is located at the opposite corner with coordinate (100,100). It is assumed that the number of CRs increases gradually from 0 to 5. For each phase, 10 packages are delivered from the source node to the sink node and then a new CR is generated. Each CR is composed of 50 CNs and the center (x, y) of each CR is constrained in a square with $x \in [20, 80]$ and $y \in [20, 80]$. For simplicity, we assume that the CNs always tamper or drop the packages. It is further assumed that the length of a package is 1024 bits. Considering that the packages used to build the "gradients" contains less valuable information, the lengths of them are set to be 128 bits in DD. The values of (β, γ) are chosen from $\{(0, 0), (2, 1), (4, 2)\}$ and then SPRA acts as three candidates in the simulation. The energy consumption model of transmitting a l-bit length package is defined as:

$$E_T(l, R_c) = lE_{elec} + l\epsilon R_c^2,$$

where $E_{elec} = 50nJ/bit, \epsilon = 10pJ/bit/m^2$. In addition, receiving a l-bit length package consumes $E_R(l) = lE_{elec}$ energy and each node has $5J$ of energy initially. Each simulation is conducted for 100 times and the average results are presented.

B. SUCCESS RATE OF PACKAGE DELIVERY

We employ the success rate of package delivery to measure the network security with different number of CRs. Note that, we say that the delivery process of a package is failed if the package is intercepted, tampered or dropped. As shown in Fig. 6, the two static algorithms cannot provide any protection on the packages and the success rate significantly decreases with the increasing of CRs. This can be explained by the fact that quite a lot packages are sent to the CNs and hence they are tampered or dropped. In the following simulation, the static algorithms are ignored. The other three algorithms can always provide strong protection and most packages can be delivered successfully. This is reasonable considering that all the CRs are taken into account when designing the routing paths and hence the packages are not transmitted through CRs. With the increasing of the number of CRs, more and more regions in the network are covered by the CNs and it increases the probability that the packages being captured by the CNs. Therefore, the success rate of package delivery for all the schemes decreases with the increasing of the number of CNs. Moreover, the performance of SPRA is affected by the parameters β and γ. When we set $\beta = 0, \gamma = 0$, the success rate of SPRA is much lower than that of other dynamic algorithms, because the agent nodes are too close to the CRs and it is possible that some packages are sent to the CNs. Similarly, the packages in Dynamic DD and GPSR can also be intercepted by the CNs in some cases. With the increasing of β and γ, the distances between the agent nodes and the CRs also increase and in this case, the probability of that the packages being captured by the CNs decreases. As a consequence, the success rate of package delivery increases. When we set $\beta = 2, \gamma = 1$ or $\beta = 4, \gamma = 2$, SPRA and the other two dynamic schemes

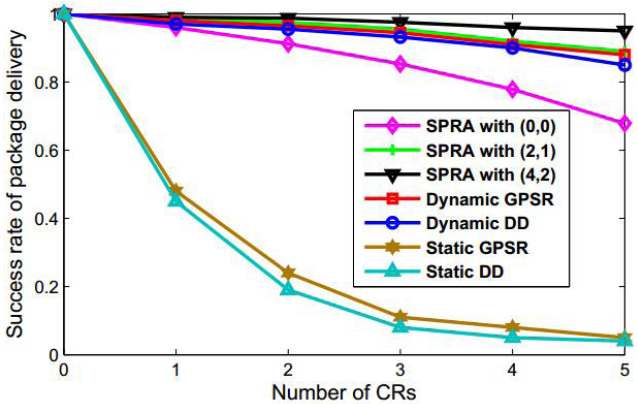

FIGURE 6. Success rate of package delivery.

have similar performance in terms of success rate of package delivery.

C. AVERAGE LENGTH OF THE ROUTING PATHS

The average hops of the routing paths are shown in Fig. 7 and it can be observed that the average hops of all the algorithms monotonously increase with the increment of CRs. This is reasonable, because the CRs block the routing paths and the packages need to be delivered around them. SPRA with $\beta = 0, \gamma = 0$ and Dynamic DD have similar performances and they perform much better than the other schemes. GPSR is likely to be stuck by the local optimums and hence some extra hops are needed to overcome them. In fact, SPRA and Dynamic DD always find the global optimums and try to avoid the local optimums in the initial phase. However, the patterns beneath these two algorithms are totally different. The Dynamic DD algorithm can almost find the shortest path in theory between the source node and the sink node, because it examines all the possible paths between them and the shortest path is selected from all the paths. In other words, Dynamic DD can find the short routing paths based on a complicated preparation phase in which a large amount of energy would be consumed. SPRA finds the short routing paths through the short geometric path and apparently this is a more intelligent way considering that the geometric path can be locally constructed by the source node. Similar to the security of the package, the average hops are also affected by β and γ. It can be observed from Fig. 7, the average hops of SPRA increase with the increasing of β and γ. This can be explained by the fact that when the agent nodes are far away from the turning points, the geometric paths are lengthened and hence the final routing paths are also lengthened.

D. TOTAL AMOUNT OF TRANSMITTED DATA

With the increment of the number of CRs, the total data transmission amount of all the schemes monotonously increases, because the optimal routing paths can be obstructed by the CRs. Though Dynamic DD has a short routing path as presented in Section V.C, the number of total transmitted packages is much larger than that of SPRA as shown in Fig. 8. This is reasonable considering that once the topology of a network is updated, a complicated preliminary phase needs to be operated in DD to guarantee the reliability of the routing paths. In theory, all the nodes need to send at least one package and receive several packages depending on the number of its neighbors to get the gradient to the sink node. Consequently, a large amount of extra data transmission is consumed in the preliminary phase and hence DD has a poor dynamics. Dynamic GPSR performs slightly better than the Dynamic DD algorithm considering that it can be executed in distributed manner. However, the total data transmission amount of Dynamic GPSR is still quite large, because the average hops of the paths are large as discussed in Section V.C. SPRA achieves the best performance with a small β and γ. For example, if we set $\beta = 0, \gamma = 0$ or $\beta = 2, \gamma = 1$, the data transmission amount is greatly smaller than that of other schemes. However, with the increasing of β and γ, the data transmission amount increases.

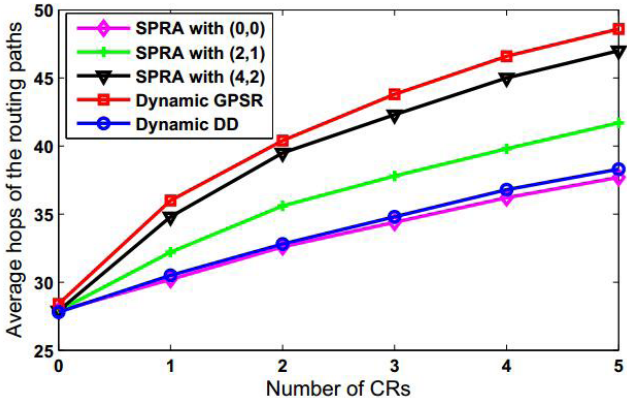

FIGURE 7. Average hops of the routing paths.

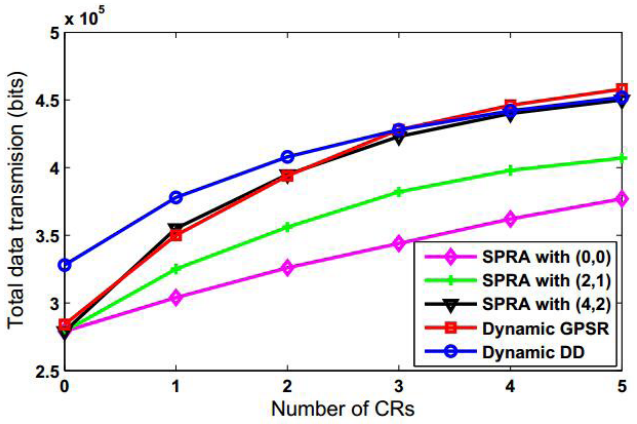

FIGURE 8. Total amount of transmitted data.

E. ENERGY EFFICIENCY OF SPRA

At last, the number of reliable nodes alive in the network is employed to evaluate the routing algorithms in terms of energy efficiency and the simulation result is presented in Fig. 9. With the increasing of received data by the sink node, the numbers of the reliable nodes alive decrease for all the routing algorithms. This can be explained by the following two facts. First, when a new CR is generated, the number of reliable nodes for each algorithm decreases by about 50 and this is a drastic change under our assumption. Second, the number of the reliable nodes also decrease by the fact that some nodes run out of energy because of sending and receiving packages. Dynamic DD consumes the most energy, because it needs to rebuild the "gradients." As a consequence, the number of alive nodes in the Dynamic DD is the smallest. Dynamic GPSR consumes less energy because of its dynamics and it performs better than the Dynamic DD. SPRA has a good performance and the smaller of β and γ, the more energy-efficient of SPRA. When we set

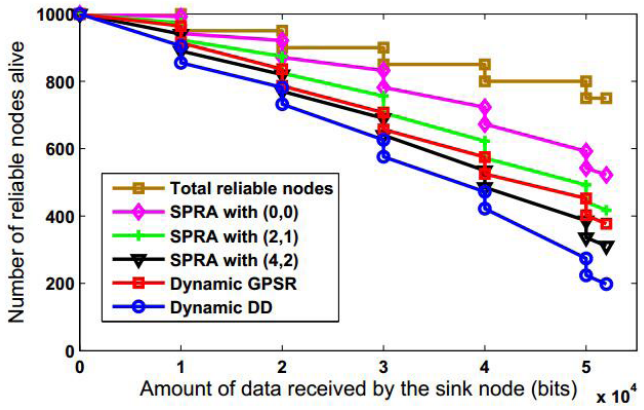

FIGURE 9. Number of reliable nodes alive in the network.

$\beta = 0, \gamma = 0$ or $\beta = 2, \gamma = 1$, our scheme outperforms other schemes significantly. In conclusion, the three dynamic algorithms can provide strong protection to the security of the packages. However, SPRA can always find the paths with an energy-efficient manner and outperforms other approaches against the CRs in WSNs.

F. PERFORMANCE DISCUSSION

Through a series of simulation, we can observe that the static GPSR and DD cannot provide any protection on the security of packages which is reasonable considering that the CNs are ignored. The dynamic GPSR, DD and SPRA can provide strong protection to defend against the CRs and apparently some extra energy is consumed compared with the static algorithms. The average hops of the routing paths generated by SPRA are similar with that of the dynamic DD and smaller than that of the dynamic GPSR. In theory, the DD algorithms can find the optimal path and this proves the effectiveness of SPRA. However, the dynamic DD consumes a large amount of energy because of the complex preparation process. SPRA performs the best in terms of energy-efficiency and greatly prolongs the lifetime of the networks. This can be explained by the fact that SPRA can be executed in a totally distributed manner and the generated routing paths are of shorter lengths.

In addition, it can be observed from the simulation results that parameters β and γ have great influence on the performance of SPRA. The increasing of β and γ can avoid the packages from being captured by the CNs and improve the security of the network. Meanwhile, it also increases the average hops of the routing paths, data transmission amount and energy consumption of the networks. Consequently, there is a trade-off between the reliability of the networks and the energy-efficiency of the networks. Fortunately, when we set $\beta = 2$ and $\gamma = 1$, SPRA performs the best in terms of both the reliability and energy efficiency. In summary, SPRA can always find the shortest routing paths while guaranteeing the security of the packages and it performs better than existing algorithms.

VI. CONCLUSION

In this paper, a secure shortest path routing algorithm called SPRA is proposed to defend against the CRs in WSNs. The CNs are identified by the reputation system and the CRs are constructed based on the convex hulls structure. Then, a geometric shortest path is calculated and several turning points are naturally selected on the path. A complex method is presented to select the agent nodes around the turning points. At last, the packages are transmitted between the agent nodes in a relay manner through a sophisticated mechanism to avoid being captured by the adversaries. Though SPRA is operated totally in a distributed manner, the source node can flexibly control the shapes of the routing paths by employing several agent nodes. Simulations results show that SPRA can always find the secure short routing paths and it is also energy-efficient. As our future work, we plan to extend SPRA to achieve a balance between the security of the packages and the energy consumption. In addition, it is also promising to research how to construct the CRs efficiently in very complex scenarios.

REFERENCES

[1] K. Gerrigagoitia and R. Uribeetxeberria, "Reputation-based intrusion detection system for wireless sensor networks," in *Proc. Complex. Eng.*, 2014, pp. 1–5.
[2] J. S. Fu and Y. Liu, "Double cluster heads model for secure and accurate data fusion in wireless sensor networks," *Sensors*, vol. 15, no. 1, pp. 2021–2040, 2015.
[3] D. Niculescu and B. Nath, "Trajectory based forwarding and its applications," in *Proc. 9th Annu. Int. Conf. Mobile Comput. Netw. ACM*, 2003, pp. 260–272.
[4] B. Karp and H. T. Kung, "GPSR: Greedy perimeter stateless routing for wireless networks," in *Proc. 6th Annu. Int. Conf. Mobile Comput. Netw. ACM*, 2000, pp. 243–254.
[5] C. Intanagonwiwat, R. Govindan, and D. Estrin, "Directed diffusion: A scalable and robust communication paradigm for sensor networks," in *Proc. 6th Annu. Int. Conf. Mobile Comput. Netw. ACM*, 2000, pp. 56–67.
[6] W. B. Heinzelman, A. P. Chandrakasan, and H. Balakrishnan, "An application-specific protocol architecture for wireless microsensor networks," *IEEE Trans. Wireless Commun.*, vol. 1, no. 4, pp. 660–670, Oct. 2002.
[7] D. Johnson, Y. Hu, and D. Maltz, *The Dynamic Source Routing Protocol for Mobile Ad Hoc Networks For IPv4*, document RFC 4728, 2007.
[8] D. Zhang, G. Li, K. Zheng, and X. Ming, "An energy-balanced routing method based on forward-aware factor for wireless sensor networks," *IEEE Trans. Ind. Informat.*, vol. 10, no. 1, pp. 766–773, Feb. 2014.
[9] Y. Zengm, J. Cao, J. Hong, S. Zhang, and L. Xie, "Secure localization and location verification in wireless sensor networks: A survey," *J. Supercomput.*, vol. 64, no. 3, pp. 685–701, 2013.
[10] R. L. Graham, "An efficient algorith for determining the convex hull of a finite planar set," *Inf. Process. Lett.*, vol. 1, no. 4, pp. 132–133, 1972.
[11] M. D. Berg and O. Mark, *Computational Geometry*. Cham, Switzerland: Springer, 2000, pp. 1–17.
[12] J. So and H. Byun, "Load-balanced opportunistic routing for duty-cycled wireless sensor networks," *IEEE Trans. Mobile Comput.*, vol. 16, no. 7, pp. 1940–1955, Jul. 2017.
[13] J.-S. Fu, Y. Liu, and H.-C. Chao, "ICA: An incremental clustering algorithm based on OPTICS," *Wireless Pers. Commun.*, vol. 84, no. 3, pp. 1–20, 2015.
[14] M. Ester, H.-P. Kriegel, J. Sander, and X. Xu, "A density-based algorithm for discovering clusters in large spatial databases with noise," in *Proc. Int. Conf. Knowl. Discovery Data Mining*, 1996, pp. 226–231.
[15] A. Jsang and R. Ismail, "The beta reputation system," in *Proc. 15th Bled Electron. Commerce Conf.*, 2002, pp. 41–55.
[16] H. Huang, H. Yin, G. Min, X. Zhang, W. Zhu, and Y. Wu, "Coordinate-assisted routing approach to bypass routing holes in wireless sensor networks," *IEEE Commun. Mag.*, vol. 55, no. 7, pp. 180–185, Jul. 2017.

[17] J. Luo, J. Hu, D. Wu, and R. Li, "Opportunistic routing algorithm for relay node selection in wireless sensor networks," *IEEE Trans Ind. Informat.*, vol. 11, no. 1, pp. 112–121, Feb. 2015.
[18] Y. Liu, M. Dong, K. Ota, and A. Liu, "ActiveTrust: Secure and trustable routing in wireless sensor networks," *IEEE Trans. Inf. Forensics Security*, vol. 11, no. 9, pp. 2013–2027, Sep. 2016.
[19] C. Zhu, H. Nicanfar, V. C. M. Leung, and L. T. Yang, "An authenticated trust and reputation calculation and management system for cloud and sensor networks integration," *IEEE Trans. Inf. Forensics Security*, vol. 10, no. 1, pp. 118–131, Jan. 2015.
[20] S. Ganeriwal, L. K. Balzano, and M. B. Srivastava, "Reputation-based framework for high integrity sensor networks," *ACM Trans. Sensor Netw.*, vol. 4, no. 3, p. 15, May 2008.
[21] P. Kuila and P. K. Jana, "Energy efficient clustering and routing algorithms for wireless sensor networks: Particle swarm optimization approach," *Eng. Appl. Artif. Intell.*, vol. 33, pp. 127–140, Aug. 2014.
[22] A. Arora *et al.*, "A line in the sand: A wireless sensor network for target detection, classification, and tracking," *Comput. Netw.*, vol. 46, no. 5, pp. 605–634, Dec. 2004.
[23] S. Ozdemir, "Functional reputation based data aggregation for wireless sensor networks," in *Proc. IEEE Int. Conf. Wireless Mobile Comput. Netw. Commun. (WIMOB)*, Oct. 2008, pp. 592–597.
[24] K. Sohraby, D. Minoli, and T. Znati, *Wireless Sensor Networks: Technology, Protocols, and Applications*. Hoboken, NJ, USA: Wiley, 2007, pp. 129–139.
[25] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "Wireless sensor networks: A survey," *Comput. Netw.*, vol. 38, no. 4, pp. 393–422, 2002.
[26] N. Wang and J. Zeng, "All-direction random routing for source-location privacy protecting against parasitic sensor networks," *Sensors*, vol. 17, no. 3, p. 614, 2017.
[27] S. Ganeriwal, L. K. Balzano, and M. B. Srivastava, "Reputation-based framework for high integrity sensor networks," *ACM Trans. Sens. Netw.*, vol. 4, no. 15, pp. 1–15, 2008.
[28] O. Younis and S. Fahmy, "HEED: A hybrid, energy-efficient, distributed clustering approach for ad hoc sensor networks," *IEEE Trans. Mobile Comput.*, vol. 3, no. 4, pp. 366–379, Oct./Dec. 2004.
[29] C. H. Lin and M. J. Tsai, "A comment on 'HEED: A hybrid, energy-efficient, distributed clustering approach for ad hoc sensor networks,'" *IEEE Trans. Mobile Comput.*, vol. 5, no. 10, pp. 1471–1472, Oct. 2006.
[30] S. L. Yang, Y.-Q. Liang, and J. Fan, "Optimization study and application on the K value of K-means algorithm," *Syst. Eng.-Theory Pract.*, vol. 26, no. 2, pp. 97–101, 2006.

NA WANG received the Ph.D. degree from the School of Mathematical Sciences, Xiamen University, in 2018. She is currently a Postdoctoral Fellow with the School of Computer Science, Beijing University of Posts and Telecommunications. Her research interests include cryptography, message sharing, and information security issues in distributed and cloud systems.

JIAN LI received the Ph.D. degree from the Beijing Institute of Technology, in 2005. He is currently a Professor with the School of Computer Science, Beijing University of Posts and Telecommunications. He has authored or co-authored 12 books and has published more than 100 professional research papers. His research interests include information security and quantum cryptography.