False data detection and dynamic selection of aggregator nodes with pair-wise key establishment in homogeneous wireless sensor networks

M.K. Sandhya*
Department of Computer Science and Engineering,
Meenakshi Sundararajan Engineering College,
Chennai, India
Email: mksans@gmail.com
*Corresponding author

K. Murugan
Ramanujan Computing Centre,
Anna University,
Chennai, India
Email: murugan@annauniv.edu

P. Devaraj
Department of Mathematics,
College of Engineering,
Anna University, Guindy Campus,
Chennai, India
Email: devaraj@annauniv.edu

Abstract: Compromised sensor nodes inject false data in wireless sensor networks which distorts data integrity and consumes battery power unnecessarily. In the existing false data detection schemes, the aggregator nodes suffer from rapid battery drain due to the computational overhead, leading to reduced network lifetime. To avoid this, the aggregator nodes must be dynamically selected from the sensor nodes in the network. This dynamic selection of aggregator nodes introduces security challenges in symmetric key exchange among the sensor nodes. In this paper, a scheme called, false data detection-dynamic selection of aggregator nodes is proposed to address these issues. This scheme discards the false data injected into the network and also prolongs the network lifetime by the dynamic selection of aggregator nodes. The problem of symmetric key exchange arising due to the dynamic selection of aggregator nodes is resolved by the proposed Chebyshev polynomial-based pair-wise key establishment which has lesser computational overhead and offers better security strength. Simulation results indicate that the scheme eliminates false data injected by multiple compromised nodes and also offers higher network lifetime in homogeneous wireless sensor networks.

Keywords: aggregator node selection; Chebyshev polynomial; false data detection; network lifetime; pair-wise key establishment; homogeneous wireless sensor networks.
1 Introduction

Wireless sensor networks consist of tiny sensor nodes that have limitations on their power supplies, bandwidth, memory and computing capacity. Energy is an extremely critical resource for battery-powered wireless sensor networks. In the recent years, revolutionary applications for monitoring, tracking, and controlling based on wireless sensor networks have emerged for a wide range of domains (Bai et al., 2011). The sensor nodes in the network sense and gather data from the field of interest, and transmit the gathered data to the external base station. An event occurring in the field can be sensed by multiple sensor nodes in close proximity in the network. This causes data redundancy, and the transmission of redundant data to the base station leads to the depletion of the precious resources like bandwidth and power. To avoid redundancy, the aggregator nodes use an appropriate function to aggregate data and transmit the aggregated data to the base station. Data aggregation results in better utilisation of bandwidth and power. The accuracy of the aggregated data is dependent on the correctness of the original data.
Injection of false data disrupts the data accuracy and poses a serious threat to data integrity. The false data is injected into the network by the compromised sensor nodes, either during data aggregation or data transmission. Moreover, the transmission of false data to the base station consumes bandwidth and power unnecessarily. Hence, false data detection is essential for providing data integrity and effective utilisation of power and bandwidth.

Most of the existing false data detection schemes in wireless sensor networks (Yang and Lu, 2004; Ye et al., 2004; Yu and Guan, 2006; Zhu et al., 2007) address the issue of false data injection during data forwarding. They deal neither with data aggregation nor with confidentiality. To provide confidentiality, these schemes need to be modified suitably. When data aggregation is incorporated, the false data detection scheme must correctly distinguish whether the data alteration is due to data aggregation or false data injection. There are schemes (Du et al., 2003b; Przydatek et al., 2003; Yang et al., 2006; Wu et al., 2007) which deal with secure data aggregation. In data aggregation and authentication (DAA) protocol (Ozdemir and Cam, 2010), the false data is detected during data aggregation as well as forwarding. The intense computations carried out at the aggregator nodes for data aggregation as well as false data detection lead to battery drain. The failure of aggregator nodes due to battery drain leads to the failure of the false data detection scheme, and also disrupts the entire communication system. Moreover, this reduces the network lifetime. The above mentioned schemes do not aim at improving the network lifetime by reducing the battery drain at the aggregator nodes. There are schemes (Gupta et al., 2013; Zha and Ng, 2013) which aim at energy efficiency and improving network lifetime, but they do not address the issue of false data injection. In DARE (Sicari et al., 2013), hybrid wireless sensor/mesh network architecture is proposed to minimise the amount of data exchanged and provide battery lifetime improvement to the wireless sensor nodes. This hybrid architecture assigns computationally intensive tasks like secure localisation and data aggregation to the mesh routers, leaving the sensor nodes in charge of data gathering functionalities. But it is not possible to implement hybrid wireless sensor/mesh networks architecture in many applications due to the challenges in interconnecting sensor and mesh networks. Hence it is necessary to reassign the aggregator nodes at regular intervals of time to balance energy consumption and improve network lifetime. In Sandhya et al. (2014, 2015), false data detection schemes with aggregator node selection for improving network lifetime are proposed for heterogeneous wireless sensor networks. These schemes exploit the heterogeneity of sensor nodes for aggregator node selection and hence cannot be used in homogeneous wireless sensor networks.

In this paper, the false data detection-dynamic selection of aggregator nodes (FDD-DSAN) scheme is proposed for homogeneous wireless sensor networks. This scheme eliminates the false data injected by the compromised sensor nodes in the network during data aggregation and forwarding. The issue of battery drain at the aggregator nodes is resolved, by dynamically selecting the aggregator nodes from the other sensor nodes. This helps in balancing the power utilisation at the aggregator nodes. The scenario of dynamic selection of the aggregator nodes introduces more security challenges in exchanging the pair-wise keys than the scenario that makes use of a fixed set of aggregator nodes. The existing key exchange schemes for wireless sensor networks cannot be applied for the scenario of dynamic aggregator node selection. The conventional public key schemes like RSA, ECC and Diffie-Hellman key exchange are not suited for wireless sensor networks, due to the computational overhead of those
algorithms. The idea of using a key ring for key exchange in this situation also fails, because of the fact that the compromise of one sensor node leads to the compromise of all other nodes. The issues in key exchange arising due to the dynamic selection of the aggregator nodes is resolved by the proposed cryptographic scheme, based on Chebyshev polynomial. The proposed key exchange scheme has lesser computational overhead and offers better security.

The remainder of the paper is organised as follows: Section 2 deals with the research work related to false data detection, aggregator node selection and key exchange schemes in wireless sensor networks; Section 3 describes the proposed FDD-DSAN scheme for wireless sensor networks; Section 4 evaluates the performance of the FDD-DSAN scheme; Section 5 presents the concluding remarks and future enhancements.

2 Related work

The statistical en-route detection and filtering (SEF) scheme (Ye et al., 2004) enables the relaying nodes and the base station to detect false data with a certain probability and drop 80%–90% of the injected false reports within 10 hops. In the interleaved hop-by-hop authentication scheme (Zhu et al., 2007), false data injected by the compromised sensor nodes is detected by those sensor nodes that collaborate to verify data integrity. In this scheme, the sensor nodes are not allowed to perform data aggregation during data forwarding. The commutative cipher-based en-route filtering (CCEF) scheme (Yang and Lu, 2004) drops the false data en-route without symmetric key sharing. In CCEF, the source node establishes a secret association with the base station on a per-session basis, while the intermediate forwarding nodes are equipped with a witness key. By using a commutative cipher (Diffie and Hellman, 1976), a forwarding node can use the witness key to verify the authenticity of the reports without knowing the original session key. In the dynamic en-route filtering scheme (Yu and Guan, 2006), false data are filtered in a probabilistic nature; i.e., a forwarding node can validate the authenticity of a report only if it has a corresponding authentication key. The aforementioned schemes do not tackle the issue of false data injection during data aggregation.

Przydatek et al. (2003) proposed a scheme with random sampling mechanisms and interactive proofs, to check the correctness of the aggregated data at the base station. Du et al. (2003b) proposed a scheme in which the witness nodes aggregate data and compute the message authentication code (MAC) to verify the correctness of the aggregated data at the base station. Wu et al. (2007) proposed a scheme in which the sensor nodes use the cryptographic algorithms only when a cheating activity is detected. In secure hop-by-hop data aggregation protocol (SDAP) more trust is placed on the high-level nodes (i.e., the nodes that are closer to the root) compared to the low-level sensor nodes (Yang et al., 2006). The above schemes address secure data aggregation, but do not aim at improving the network lifetime. DAA scheme (Ozdemir and Cam, 2010) detects the false data injected during data aggregation and forwarding, by verifying the MAC of the data packet. This scheme applies the SANE protocol (Sirivianos et al., 2007) for selecting the data aggregators at the time of the deployment. The drawback of the SANE protocol is that, it does not always ensure the required number of monitoring and forwarding nodes between any two selected data aggregators. Hence, the SANE protocol is run until the required number of monitoring and forwarding nodes is present between any two selected
aggregators. The other cluster head/aggregator node selection schemes like LEACH (Heinzelman et al., 2000) and PANEL (Buttyan and Schaffer, 2007) also do not ensure the required number of monitoring nodes and forwarding nodes between the two selected cluster heads/aggregators. DAA scheme also does not focus on improving the network lifetime.

The main aim of this paper is to eliminate the injected false data along with improving the network lifetime by dynamically selecting the aggregator nodes. The dynamic selection of aggregator nodes introduces more security challenges in the symmetric key exchange among the sensor node pairs formed, each time after the aggregator node selection. The existing random key distribution protocols (Du et al., 2003a; Liu et al., 2005) allow key establishment using multi-hop communication. There are some other techniques that allow non-neighbouring sensor nodes to establish pair-wise keys (Deng and Han, 2005; Dong and Liu, 2007). Chan and Perrig (2003) proposed a pair-wise key pre-distribution scheme which offers perfect resilience and authentication. It lacks scalability and requires more storage capacity. Zhang and Varadharajan (2010) presented a survey and a taxonomical classification of key management schemes for wireless sensor networks. The key management schemes mentioned above will not suit the scenario of dynamic selection of aggregator nodes. The other conventional schemes like RSA, ECC, Diffie-Hellman key exchange are computationally intensive, and hence do not suit wireless sensor networks.

3 Proposed work

The main objective of the proposed FDD-DSAN scheme is to discard the false data injected by the compromised nodes. This scheme dynamically selects the aggregator node from the sensor nodes in each cluster, to effectively utilise the battery power and to improve the network lifetime. It addresses the security challenges in key exchange between sensor nodes arising due to the dynamic selection of aggregator nodes.

The FDD-DSAN scheme comprises of two phases:

1. dynamic selection of aggregator nodes
2. false data detection.

The phase, dynamic selection of aggregator nodes involves three sub phases:

1. sensor node deployment
2. aggregator node selection
3. verifier and forwarding nodes selection.

The false data detection phase comprises of three sub phases:

1. pair formation among sensor nodes
2. symmetric key exchange
3. false data detection.

The summary of the notations used in this paper is presented in Table 1.
Table 1  Summary of notations

<table>
<thead>
<tr>
<th>Notation</th>
<th>Summary</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ai</td>
<td>Aggregator node of the $i^{th}$ cluster</td>
</tr>
<tr>
<td>Vi</td>
<td>Verifier node of the $i^{th}$ cluster</td>
</tr>
<tr>
<td>Ni</td>
<td>Neighbouring node of the $i^{th}$ cluster</td>
</tr>
<tr>
<td>Fi</td>
<td>Forwarding node of the $i^{th}$ cluster</td>
</tr>
<tr>
<td>Ki–j</td>
<td>Unique key shared between the base station and the $j^{th}$ node in $i^{th}$ cluster</td>
</tr>
<tr>
<td>$K_e$</td>
<td>Group key between the aggregator node and its neighbouring nodes</td>
</tr>
<tr>
<td>$K_{ij}$</td>
<td>Pair-wise key shared between two sensor nodes $i$ and $j$</td>
</tr>
<tr>
<td>$E_{K_{ij}}(D)$</td>
<td>Data $D$ encrypted using the key $K_{ij}$</td>
</tr>
<tr>
<td>$MAC_{K_{ij}}(D)$</td>
<td>MAC of the data using the key $K_{ij}$</td>
</tr>
<tr>
<td>$T_s(x), T_r(x)$</td>
<td>Recurrence relation in Chebyshev polynomial</td>
</tr>
</tbody>
</table>

3.1 Sensor node deployment

The sensor nodes deployed in the target field are stationary as shown in Figure 1. The sensor network applications like temperature sensing, environmental observation and forecasting etc., employ stationary sensor nodes. The random deployment strategy is applied to deploy the sensor nodes in the target field. There are $n \times n$ clusters present in the network. In each cluster there are atleast $n \times n$ nodes. Each node is represented in terms of its cluster ID and node ID. These sensor nodes are grouped into clusters, and each cluster has an aggregator node. For the very first time i.e., at the time of deployment, the aggregator node for each cluster is selected by the base station. An event occurring in the field is sensed by atleast ‘$m$’ sensor nodes in a cluster, where $m \leq n$.

Figure 1  Deployment of sensor nodes for the FDD-DSAN scheme
3.2 Aggregator node selection

The base station selects the aggregator node for each cluster for the first instance i.e., at the time of deployment. From the next instance, the base station initiates the aggregator node selection process by sending an initiation message to the current aggregator node of each cluster. This initiation message is sent by the base station each time, to start the aggregator node selection process. In each cluster, the current aggregator node dynamically selects the next aggregator node from the pool of sensor nodes which sends higher energy advertisement. This avoids the selection of a dead node as the aggregator node. If a dead node is selected, then the wireless sensor network would be broken and needs to be re-organised. The selection of aggregator nodes based on the higher energy advertisement avoids the choice of a nearly dead node.

3.3 Verifier and forwarding nodes selection

The aggregator node selection process is followed by the selection of verifier and forwarding nodes. The nodes surrounding the aggregator node within the cluster are its neighbouring nodes. Each aggregator node has at least \((n^2 - 1)\) neighbouring nodes. Then \(m\) verifier nodes are chosen from those neighbouring nodes \((m \leq n)\). There are \(m\) forwarding nodes chosen between any two aggregators \((m \leq n)\). The neighbouring nodes, verifier nodes and the forwarding nodes of the \(i^{th}\) cluster are represented as \(N_i\), \(V_i\) and \(F_i\) respectively.

3.4 Pair formation among sensor nodes

The pair formation among sensor nodes shown in Figure 2 is essential for eliminating the false data by verifying the MAC of the data packet. Any two consecutive aggregators in the network form the \(A_i-A_{i+1}\) pair. There is only one pair between two aggregators. The verifier nodes of the \(i^{th}\) cluster and neighbouring nodes of the \(i + 1^{th}\) cluster also establish pairing, called the \(V_i-N_{i+1}\) pair. The number of such pairs that are established is \(m\). The verifier nodes and the forwarding nodes of the \(i^{th}\) cluster establish \(V_i-F_i\) pairs. The number of such pairs is also \(m\). The total number of pairs that is established between the sensor nodes of two clusters in the network is \((2m + 1)\).

**Figure 2** Pair formation among sensor nodes (see online version for colours)

Note: Suffix \(i\) and \(i + 1\) indicate the current and the next cluster respectively.
3.5 Symmetric key exchange

This section presents the proposed pair-wise key establishment scheme for exchanging symmetric keys between the sensor node pairs, based on Chebyshev polynomial. The original cryptosystem using Chebyshev polynomial (Korev and Tasev, 2003) is easily breakable. The proposed scheme is not easily breakable and it is best suited for wireless sensor networks as it is not computationally intensive.

3.5.1 Drawback of the original cryptosystem

The original cryptosystem based on Chebyshev polynomial is easily breakable (Bergamo et al., 2005) because of the transmission of x along with \( T_s(x) \) in the public key (x, \( T_s(x) \)). An adversary, knowing the public key (x, \( T_s(x) \)) and the ciphertext (\( T_{r,s}(x) \), x), can recover \( M \) by computing an \( r' \) such that \( T_{r}(x) = T_{r'}(x) \). Then \( T_{r,s}(x) \) is evaluated as \( T_{r'}(T_{r}(x)) \) to recover \( M \) by computing x / \( T_{r}(x) \). The adversary is able to perform the above attack, because x is published as a component of the key i.e., (x, \( T_s(x) \)). The attack is always successful because, if an \( r' \) value is computed such that \( T_{r'}(x) = T_{r}(x) \), then \( T_{r,r}(x) = T_{r'}(T_{r}(x)) \).

Figure 3 Algorithm for Chebyshev polynomial based pair-wise key establishment

----

Algorithm for Pair-wise key establishment

**Input:** Random number \( x \in [-1, 1] \)

**Output:** Exchange of the symmetric key ‘M’ between Node A & B

// At Base Station

Select a random number \( x \in [-1, 1] \)
Encrypt x as \( E_{K_{A}}(x) \) and \( E_{K_{B}}(x) \)
Send \( E_{K_{A}}(x) \) to Node A and \( E_{K_{B}}(x) \) to Node B

// At Node A

Decrypt \( E_{K_{A}}(x) \) to get x
Select a large integer \( s \)
Compute \( T_s(x) \) and send \( T_s(x) \) to Node B

// At Node B

Decrypt \( E_{K_{B}}(x) \) to get x
Choose the symmetric key \( M \in [-1, 1] \)
Select a large integer \( r \)
Compute \( T_r(x) \)
Compute \( T_{r,s}(x) = T_s(T_r(x)) \)
Compute \( V = M \cdot T_{r,s}(x) \)
Send the ciphertext \( C = (T_r(x), V) \) to Node A

// At Node A

Compute \( T_{r,r} = T_r(T_r(x)) \)
Compute the symmetric key \( M = V / T_{r,r}(x) \)
3.5.2 Proposed Chebyshev polynomial-based pair-wise key establishment

The drawback of the original scheme is overcome in the proposed scheme. The modification is that the key establishment process is initiated by the base station by selecting a random number $x \in [-1, 1]$. Each node in the network shares a unique key $K_{i,j}$ with the base station. The key $K_{i,j}$ represents the key shared by the $j$th node in $i$th cluster with the base station. The random number $x$ has to be securely transmitted to the two nodes that need to establish the pair-wise keys, i.e., Node A and Node B. Node A shares key $K_{i-A}$ and Node B shares key $K_{i-B}$ with the base station. The chosen random number $x$ is encrypted separately by the keys $K_{i-A}$ and $K_{i-B}$, corresponding to Node A and Node B. The encrypted values are sent to the respective nodes. The algorithm for the proposed Chebyshev polynomial-based pair-wise key establishment is given in Figure 3.

3.5.2.1 Computation of $T_s(x)$, $T_r(x)$ and $T_s(T_r(x))$

The value of $T_s(x)$ is computed using the following recurrence relation

$$T_{s+1}(x) = 2xT_s(x) - T_{s-1}(x) \quad (1)$$

Similarly, $T_r(x)$ is computed using the following recurrence relation

$$T_{r+1}(x) = 2xT_r(x) - T_{r-1}(x) \quad (2)$$

Substituting $T_0 = 1$ and $T_1 = x$, the values of $T_2(x)$, $T_3(x)$, $T_4(x)$ are calculated as:

$$T_2(x) = 2x^2 - 1 \quad (3)$$
$$T_3(x) = 4x^3 - 3x \quad (4)$$
$$T_4(x) = 8x^4 - 8x^2 + 1 \quad (5)$$

The value of $T_s = T_s(T_r(x))$ is calculated by computing $T_r(x)$ first and then computing $T_s(x)$, treating $T_r(x)$ as $x$.

The base station maintains a key table for storing these keys i.e., $K_{i,j}$. The size of the key table is equal to the number of nodes in the network. The base station is not limited in terms of energy, computing and memory resources. Hence, the cost incurred in maintaining the key table at the base station is not a serious issue. Further, this expense is unavoidable for providing transmission security and data confidentiality in a network.

3.5.3 Analysis of the proposed key exchange scheme

The algorithm is correct due to the semi-group property of the Chebyshev polynomials.

$$V = M \cdot T_s(T_r(x)) \quad (6)$$

Since Chebyshev polynomials commute under composition, it follows that

$$T_r(T_s(x)) = T_s(T_r(x)) \quad (7)$$
Hence, $V$ can be computed as:

$$V = M \cdot T_r(T_r(x))$$

(8)

Therefore, $M$ value is calculated as:

$$M = V / T_{s+1}(x)$$

(9)

The original scheme based on Chebyshev polynomial is easily breakable. The attacker is able to recover $M$, because $x$ is sent as a component of the key i.e., $(x, T_r(x))$. The drawback of the original scheme is resolved by preventing the transmission of $x$ along with $T_r(x)$ from one sensor node to another node in the pair. In the proposed scheme, the $x$ value is securely transmitted by the base station to both the sensor nodes, established as a pair. It may be noted that the value of $x$ is neither published as a component of the key nor transmitted as plaintext. Hence, the proposed scheme is secure against the attack presented by Bergamo et al.

This scheme is suited for wireless sensor networks, as it involves successive substitution of values and does not require complex computations. This is evident from the recurrence relations given in equations (1) and (2). This scheme offers resilience against node compromise, authentication and scalability.

3.6 False data detection

The false data detection scheme detects and discards the false data injected by the compromised nodes during data aggregation and data transmission. This is done with the help of the node pairs formed by the verifier nodes, neighbouring nodes and the forwarding nodes along with the aggregator nodes. The total number of pairs that is established between the sensor nodes of two clusters in the network is $(2m + 1)$. This scheme can handle multiple node compromises because a single event is sensed and endorsed by ‘$m$’ sensor nodes in the cluster. The number of compromised nodes that can be handled by this scheme is $m - 1$. False data detection involves the checking of the MAC of the data packet at three stages:

1. previous aggregator $A_{i-1}$ to the current aggregator $A_i$
2. current aggregator $A_i$ to the next aggregator $A_{i+1}$
3. between the forwarding nodes.

3.6.1 Previous aggregator $A_{i-1}$ to the current aggregator $A_i$

$A_{i-1}$ encrypts the data using the key shared between $A_{i-1}$ and $A_i$. Then it generates MAC for the data, and the encrypted data. These values are bundled as a packet shown in Figure 4 and it is sent to $A_i$. $A_i$ decrypts the data and then encrypts it using the group key. Then it is broadcast to the nodes along with the MAC value. Verification is carried out in each of the $V_{i-1-N_i}$ node pairs. The data is discarded even if the verification fails in one of the $V_{i-1-N_i}$ node pairs.
3.6.2 Current aggregator $A_i$ to the next aggregator $A_{i+1}$

A$_i$ encrypts the aggregated data using the key shared between A$_i$ and A$_{i+1}$ and broadcasts it. The verifier node computes MAC(E$_{K_{i,i+1}}$(D)) using the key it shares with V$_r$-Fi node pair and MAC(D) using the key it shares with V$_r$-Ni$_{i+1}$ node pair and transmits them to A$_i$. A$_i$ again computes these MAC values for verification. On successful verification, it forms a packet and forwards it to A$_{i+1}$.

3.6.3 Between forwarding nodes

The V$_r$-Fi node pairs verify the data packet by checking MAC(E$_{K_{i,i+1}}$(D)). On successful verification, the packet is passed to A$_{i+1}$. The data packet is discarded if the verification fails in any one of the node pairs.

The MAC of the data and the encrypted data, present in the data packet is used for verifying the correctness of the data at the sensor node pairs. The format of the data packet is shown in Figure 4. In this scheme, the data packet structure includes a 29 byte payload and two 4 byte MACs. The destination address and source address comprises of two sub fields – the cluster ID and ID of the sensor node. The message type field can be either an event or a forward. If the cluster ID of the source address field and the cluster ID of the destination address field are the same then the message type can be either an event or a forward. If the cluster ID of the source address field and the cluster ID of the destination address are different, then the message type should be a forward. If the message type is an event, then the packet is discarded due to the fact that the event has occurred in a different cluster. The algorithm for false data detection is given in Figure 5.
Algorithm for False Data Detection

**Input:** 'n' neighbouring nodes, 'm' verifier nodes and 'm' forwarding nodes of each aggregator $A_i$

**Output:** False data injected by $m - 1$ compromised sensor nodes is eliminated

/* Previous aggregator $A_{i-1}$ to current aggregator $A_i$ */
$A_{i-1}$ encrypts data $D$ using the key $K_{i-1,i}$ shared between the previous aggregator and the current aggregator as $E_{K_{i-1,i}}(D)$
$A_{i-1}$ transmits $E_{K_{i-1,i}}(D)$, MAC($E_{K_{i-1,i}}(D)$), MAC($D$) as a data packet to $A_i$
$A_i$ decrypts $E_{K_{i-1,i}}(D)$ to get $D$
Using the group key $K_{Group}^u$, $A_i$ encrypts $D$ as $E_{K_{Group}^u}(D)$ and transmits it along with MAC($D$)
Verification is carried out at $V_{i-1}$-$N_i$ pair mates and the data is discarded if the verification fails in one pair of sensor nodes.

/* Current aggregator $A_i$ to Next aggregator $A_{i+1}$ */
$A_i$ encrypts the data $D$ as $E_{K_{i,i+1}}(D)$ using key $K_{i,i+1}$
The verifier node computes MAC($E_{K_{i+1,i}}(D)$) using the key it shares with $V_i$-$F_i$ pair and MAC($D$) using the key it shares with $V_i$-$N_{i+1}$ pair and sends them to $A_i$
$A_i$ computes MAC($E_{K_{i,i+1}}(D)$) and MAC($D$) for verification.
On successful verification, $A_i$ forms a packet comprising of $E_{K_{i,i+1}}(D)$, MAC($E_{K_{i,i+1}}(D)$) and MAC($D$) and this packet is sent to $A_{i+1}$

/* Between forwarding nodes */
if a sensor node is not in a $V_i$-$F_i$ pair then
Forward data packet to the next forwarding node
else
$F_i$ verifies the MAC($E_{K_{i,i+1}}(D)$)
if verification is successful then
pass data packet to $A_{i+1}$
else
discard data packet and inform $A_i$
end if
end if

4 Performance evaluation

The simulation of this work was carried out using NS-2 (2012) simulator with 300 nodes for an area of 500 m $\times$ 500 m, and the simulation parameters are listed in Table 2. The simulation was run for 200 seconds and 50 runs of simulation were averaged to obtain each observation. The performance of the FDD-DSAN scheme is evaluated by comparing it with the DAA scheme. The comparison is made with DAA scheme because, this deals with false data detection during data aggregation as well as data forwarding. The performance of the FDD-DSAN is evaluated based on security, computational overhead, communication overhead and energy consumption. Further, the performance of the
proposed Chebyshev polynomial based pair-wise key establishment scheme is also evaluated by comparing it with the existing pair-wise scheme proposed by Chan and Perrig.

Table 2  Simulation parameters

<table>
<thead>
<tr>
<th>Simulation parameters</th>
<th>Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>Simulation area</td>
<td>500 m × 500 m</td>
</tr>
<tr>
<td>Sensor nodes</td>
<td>50 to 300</td>
</tr>
<tr>
<td>Number of base station</td>
<td>1</td>
</tr>
<tr>
<td>Packet size</td>
<td>512 bytes</td>
</tr>
<tr>
<td>Antenna</td>
<td>Omni directional</td>
</tr>
<tr>
<td>MAC protocol</td>
<td>IEEE 802.11</td>
</tr>
<tr>
<td>Traffic type</td>
<td>CBR</td>
</tr>
<tr>
<td>Initial energy</td>
<td>1 Joule</td>
</tr>
</tbody>
</table>

The performance metrics used for the comparison and evaluation of the proposed FDD-DSAN scheme are:

- **data transmission:** this is the total amount of data transmitted (in bytes) in the network
- **energy consumption:** it is the amount of energy expended by a sensor node during communication in the network
- **network lifetime:** it is the time duration between the start of the network operation to the time when the first node in the network expires
- **number of MAC computations:** this is the total number of MAC computations done in the FDD-DSAN scheme
- **data redundancy:** The data that is redundantly sensed and transmitted in the network.

4.1 Security analysis

The probability of filtering false data is the metric used to evaluate the security of the proposed FDD-DSAN scheme. It refers to the elimination of the injected false data from the network during data aggregation and data forwarding. The impact of the number of nodes on the probability of filtering false data for the FDD-DSAN scheme is compared with the DAA scheme. From Figure 6, it is observed that the probability of filtering false data of the FDD-DSAN scheme is higher than that of the existing DAA scheme. The higher filtering capacity is achieved because of the dynamic aggregator node selection which offers longer lifetime of the sensor nodes. This longevity of the sensor nodes helps in retaining the sensor node pairs which verifies the data packet transmitted in the network.

The FDD-DSAN scheme is able to successfully eliminate false data by verifying the MAC of the data packet at the sensor node pairs. The neighbouring nodes of \( A_i \) verify the data broadcast by \( A_i \) and hence \( A_i \) cannot inject any false data. The verifier nodes of \( A_i \) also compute MAC and hence any false data injected by \( A_i \) will be identified by the neighbouring nodes of \( A_{i+1} \). Similarly the ‘m’ \( V_i-F_i \) pairs also verify the MAC of the data.
packet and eliminate false data. The value of ‘m’ is chosen based on the density of node deployment and the security requirement. This indicates that scheme can effectively defend against m – 1 node compromises. Apart from false data elimination, this scheme can defend against replay messages, Denial-of-service (DoS) attacks. The sequence number along with the timestamp/nonce in the data packet helps in identifying the replayed messages. The compromised nodes can encrypt the data packets with a legal key and launch DoS attacks. This scheme can defend against such DoS attacks launched by the compromised nodes. This is achieved by the sensing of the single event by the multiple sensor nodes in each cluster. There is at least ‘m’ sensor nodes in a cluster that sense a single event and each sensor node endorse the data and transmit them to the aggregator node. Hence the DoS attack launched by m – 1 compromised nodes can be defended.

Figure 6 Impact of number of nodes on the probability of filtering false data (see online version for colours)

4.2 Computational overhead

The computational overhead of the FDD-DSAN scheme is due to the computation of MACs, data aggregation and encryption/decryption processes. The computation of MAC in FDD-DSAN is necessary, for verifying the data packet and detecting the false data injected by the compromised nodes. The aggregator node and the verifier nodes in each cluster compute m + 1 MACs for plain data and m + 1 MACs for encrypted data. Therefore 2m + 2 MACs are computed in each cluster for a single data sensed. For verification, another 2m + 2 MACs are computed. The total MAC computations involved in this scheme is 4m + 4 for a single data. Further this scheme requires m + 1 aggregation processes and m + 2 encryption/decryption processes for each data sensed in network.

The impact of the number of nodes on the number of MAC computations indicates the computational overhead and it is plotted as a graph, shown in Figure 7. It is observed that the number of MAC computations increases as the number of nodes increases, for both FDD-DSAN scheme and DAA scheme. The number of MAC computations involved in the FDD-DSAN scheme is almost equal to that of the DAA scheme. This indicates that the aggregator node selection algorithm in the FDD-DSAN scheme does not influence the number of MAC computations.
4.3 Communication overhead

The communication overhead of the FDD-DSAN scheme is due to the transmission of additional MAC for false data detection and secure data aggregation. The impact of the number of nodes in the network on the total data transmission is represented in Figure 8. The total data transmitted by the sensor nodes increases as the number of sensor nodes increases for both FDD-DSAN scheme and DAA scheme. Simulation results indicate that the amount of data transmitted by the FDD-DSAN scheme is higher than that of the DAA scheme. Both these schemes deal with secure data aggregation and false data detection. But the FDD-DSAN scheme additionally involves an aggregator node selection process. The additional messages transmitted during the process of the aggregator node selection and key exchange lead to the increase in total data transmission of the FDD-DSAN scheme compared to the DAA scheme.

Figure 8  Impact of the number of nodes on total data transmission (see online version for colours)
The amount of false data injected into the network can be higher than that of the legitimate data. Hence the ratio of false-to-legitimate data is greater than 1. The false data detection is carried out between two aggregator nodes. If there are “h” hops between any two aggregator nodes, then the false packets can travel at most “h” hops. In Figure 9, the impact of the ratio of false-to-legitimate data on the total data transmission is represented. It is seen that as the ratio of false-to-legitimate data increases, the data transmitted by the sensor nodes also increases. Moreover, the amount of data transmitted by the FDD-DSAN scheme is slightly higher than that of the DAA scheme, due to the additional messages involved in aggregator node selection process and key exchange. Simulation results indicate an increase of about 2–3% in the data transmission for the FDD-DSAN scheme, when compared to the DAA scheme.

**Figure 9** Impact of the ratio of false-to-legitimate data on total data transmission (see online version for colours)

![Figure 9](image1)

**Figure 10** Impact of the percentage of data redundancy eliminated on total data transmission (see online version for colours)

![Figure 10](image2)
False data detection and dynamic selection of aggregator nodes

The impact of the percentage of data redundancy eliminated on the total data transmission is shown in Figure 10. The elimination of data redundancy through data aggregation leads to a lesser transmission overhead. As the percentage of data redundancy elimination increases, the total data transmission in the network reduces. Comparing the two schemes it is found that the FDD-DSAN scheme has a slightly higher data transmission than the DAA scheme, due to the additional messages involved in the aggregator node selection process and key exchange.

4.4 Energy consumption and network lifetime

The impact of the number of nodes in the network on the average energy consumption of the sensor nodes is shown in Figure 11. The graph indicates that the average energy consumption by the sensor nodes increases as the number of sensor nodes increases, for both FDD-DSAN scheme and DAA scheme. Simulation results indicate that the average energy consumption for both the schemes is almost the same. The dynamic selection of aggregator nodes prevents battery drain and improves network lifetime by balancing the energy consumption among the sensor nodes in the network but does not have an impact on the average energy consumption of the sensor nodes in the network.

Figure 11 Impact of number of nodes on the average energy consumption (see online version for colours)

The influence of the number of nodes on the network lifetime is shown in Figure 12. From the graph it is seen that the increase in the number of nodes leads to a decrease in the network lifetime for both FDD-DSAN scheme and DAA scheme. On comparing the two schemes, the network lifetime is higher if the FDD-DSAN scheme is employed. In DAA, there is a fixed set of aggregator nodes in the network, which is overloaded by the process of data aggregation and false data detection. This causes a quick battery drain at the aggregator nodes, leading to a lesser network lifetime. In FDD-DSAN, the aggregator node is chosen from the pool of sensor nodes having higher energy. Therefore the load is shared among the sensors nodes that are chosen as aggregator nodes in each cluster. This reduces the uneven battery drain at the aggregator nodes and thereby improves the network lifetime.
4.5 Performance comparison of proposed pair-wise key establishment scheme

The performance of the proposed pair-wise key establishment based on Chebyshev polynomial is compared with the existing pair-wise key establishment scheme proposed by Chan and Perrig. The comparison is presented in Table 3. The proposed pair-wise key establishment scheme based on Chebyshev polynomial is secure and resistant to node compromises. Even if a node is compromised, it does not compromise the keys of the other sensor nodes. It outperforms the existing key exchange scheme in terms of storage and scalability. The proposed scheme does not store the keys in the sensor node, rather it gets the encrypted seed value from the base station and computes the pair-wise keys. This brings in a great reduction in the storage of keys at the resource constrained sensor nodes. The key table is maintained at the base station as it does not have storage or computing limitations. Therefore the storage complexity at the sensor nodes is less. The proposed key establishment scheme has a high scalability because the addition of new nodes in the network is not an issue in exchanging keys.

### Table 3 Comparison between the existing and proposed key exchange scheme

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Pair-wise key establishment scheme by Chan and Perrig</th>
<th>Proposed Chebyshev polynomial based key establishment</th>
</tr>
</thead>
<tbody>
<tr>
<td>Computational complexity</td>
<td>Search</td>
<td>Search</td>
</tr>
<tr>
<td>Communication complexity</td>
<td>$1 \times 1$</td>
<td>$1 \times 1$</td>
</tr>
<tr>
<td>Storage complexity</td>
<td>$2N$</td>
<td>$1$</td>
</tr>
<tr>
<td>Scalability</td>
<td>Low</td>
<td>High</td>
</tr>
<tr>
<td>Connectivity</td>
<td>$1$</td>
<td>$1$</td>
</tr>
<tr>
<td>Resilience against node capture attacks</td>
<td>Yes</td>
<td>Yes</td>
</tr>
</tbody>
</table>
5 Conclusions

The FDD-DSAN scheme addresses the issue of false data injection by the compromised nodes along with the effective utilisation of battery power, by dynamically selecting the aggregator nodes based on the energy advertisement. The injected false data is detected by checking the MAC of the data packet at every sensor node pair. This scheme can successfully detect the false data injected by $m-1$ compromised nodes. The problem of pair-wise key establishment between the sensor nodes due to the dynamic selection of the aggregator nodes is resolved by the proposed scheme based on Chebyshev polynomial. This pair-wise key establishment scheme proposed in this paper overcomes the drawback of the original scheme based on Chebyshev polynomial and offers better security. Simulation results indicate that this scheme has a higher network lifetime by balancing the energy consumption of the sensor nodes in the network. Although, compromised nodes can launch various types of attacks, this paper resolves the issue of false data injection by the compromised nodes. As a future work, the other attacks will be considered. It is also planned to reduce the data transmission overhead due to the aggregator node selection process, and to estimate the optimum time interval for the aggregator node selection. Further, the mobility of the sensor nodes in the network will also be considered.

References


