Accomplishment of the key setting up: The flexible approach.

Huyen Thi Thanh Nguyen, Sang-Ho Na, Ga-won Lee, and Eui-Nam Huh
Department of Computer Engineering
Kyung Hee University
1 Seocheon-Dong, Giehung-Gu, Yongin,
Gyeonggi-Do 446-701, South Korea
Email: huyen@icns.khu.ac.kr, philiosnaz@gmail.com, gawon@icns.khu.ac.kr, johnhuh@khu.ac.kr

Abstract - The chief predicament against the accomplishment of key setup in sensor networks is the deficient fraction of the keys shared between the neighboring nodes, and the efforts to keep the nodes around their expected position becomes more inadequate in many applications. In this paper, we have focused our attention on a progressing flexible polynomial pre-distribution scheme to increase the connectivity whilst maintaining standard security and communication levels even in the case of large deployment errors.

Keywords: Key distribution, polynomial Pre-distribution, key management, sensor networks.

1 Introduction

Various applications such as military, transportation, health care, etc., utilize sensor networks as the beneficial medium for assembling the information from their environment. At the same time, the intensified deployment densities of the sensor nodes and their vulnerable nature make them susceptible to capture and control by the adversary forces, and hence, the secure maintenance of sensor nodes is vital for the confidentiality, integrity and availability of the transmitted data [1]. Key setup is one of the most crucial aspects in the key management from the viewpoint of the security issues, and the chief predicament against the accomplishment of the key setup in sensor networks is the deficient fraction of the keys shared between the neighboring nodes. Some reports have focused on pre-deployment knowledge [2], [3], [4], post-deployment knowledge [3], [5], state of sensors [6] and overlapping key string [7]. It is widely known that the information received from the sensor networks is not usable in real-terms as it becomes inept to maintain the nodes around their expected position in many applications. As the result, the performance of the network is lowered unintentionally along with an increase in its maintenance cost. In the present study, we focused on a progressive flexible polynomial pre-distribution scheme to increase the connectivity whilst maintaining the standard levels of security and communication even in the case of large deployment errors.

The remainder of this paper is organized as follows: In section 2, we look at the related works by other researchers and their advantages and drawbacks. The proposed scheme is shown in section 3 and its analysis in section 4. Finally, we present the conclusions in section 5.

2 Related work

In this section, previous research reports related to key management in WSN security are reviewed. Schnauzer and Gligor [8] proposed the basic probabilistic key pre-distribution where each node stores a random subset of keys from a large key pool before deployment. As a result, two nodes have a certain probability to share at least one key after deployment. Chan et al. [4] extended this scheme to significantly enhance the security and resilience of the network by requiring two sensor nodes to share at least q pre-distributed keys to establish a pair-wise key. Liu et al. [9] proposed several schemes which use location information as the pre-deployment knowledge or the post-deployment knowledge. These schemes focus on saving memory cost while remaining highly secure environment. Li et al. [2] and Delgosha et al. [10] extended the grid-based key pre-distribution in the closest polynomials scheme as reported by Liu et al. [3] to the hexagon-based key pre-distribution to improve the probability of a successful key establishment. Du et al. [11], [12] considered many factors prior to the deployment of data packets in order to avoid unnecessary key assignments, and was found to be quite useful in improving the performance of the key management in short distance peer to peer (P2P) secure communication. Anjum [5] deployed the anchor nodes to collect location information and distribution of keys to the sensor nodes. Park et al. [6] designed a state-based key management system. To reduce the energy consumption, Lai et al. [7] proposed an overlapping key sharing scheme (OSK). A detailed survey of the key management in ad hoc networks is given in [13]. A latest survey of key distribution mechanisms for WSN is presented in [14]. Zhu et al. [15] combined the probabilistic key sharing and threshold secret sharing to prevent collusive attacks. Some other few researchers have also proposed using combination schemes. Younis et al. [1] worked on Exclusion Basis Systems (EBS) - a combinatorial formulation of group key management problems. Carman [16] combined the benefits...
of both identity-based cryptography and random key pre-distribution.

3 The proposed scheme

Table I defines the symbols throughout the paper.

### 3.1 The polynomial pre-distribution

Two sensors I and J, respectively belong to two adjacent cells $C_{ij}$ and $C_{ij'}$. The polynomial distribution is as follows (Fig. 1):

1) Step 1: set-up server creates two polynomials $f_{i,j}(ID, y)$, $f_{i,j'}(ID, y)$ to assign for node I, and $f_{i,j}(ID, y)$, $f_{i,j'}(ID', y)$ to assign for node J.

2) Step 2 (after deployment): if two nodes I and J want to communicate with together, they send the polynomial IDs to each other.

3) Step 3: from the polynomial IDs, each node can calculate the common keys: $k_{1,1'} = f_{i,j}(ID, ID')$ and $k_{1,1'} = f_{i,j'}(ID, ID')$

4) Step 4: the direct key is encrypted information between these two sensors $k_{1,1'} = k_{1,1'}|k_{1,1'}$

### 3.2 The polynomial pre-distribution with the polygon grid cell

Assumed that $u$, $v$ and $\bar{u}$, $\bar{v}$ present the deployment location and the expected location of node $u$ and node $v$ respectively. Both $u$ and $v$ have the maximum deployment error $\varepsilon$, so the area where any node $v$ with its expected location has a probability to communicate with node $u$ is inside the communication limitation. The radius of the communication limitation circle is equal to $R + 2\varepsilon$.

In some of the existing schemes, the deployment error and signal range were not given due considerations, and along with an infirmly fixed number of the adjacent cells which resulted in the low polynomial sharing ratio with the small side-length of the cell along with large communication limitation (Fig. 2). Due to this disadvantage, in the present study, we propose a flexible approach for the polynomial pre-distribution scheme, in which we take the signal range and deployment error into consideration to define an efficient number of adjacent cells that promotes the polynomial sharing rate.

#### 3.2.1 The cell grid model:

Among the popular grid cell types like the triangle, square, pentagonal and the hexagonal grid cells, the apothem of the polygon cell is estimated by the Eq.(1):

$$\text{Apothem} = \frac{L}{2 \tan \frac{\pi}{n}}$$

If $n_{ex}$ denotes the number of the adjacent cells of node $u$ along one direction, then the length of the $n_{ex}$ apothems should be satisfied by the following Eq.(2):

$$\frac{L}{2 \tan \frac{\pi}{n}} * (4n_{ex} + 1) \geq (R + 2\varepsilon)$$

Eq.(2) is defined the nodes $v$ that can communicate with node $u$ and share at least one polynomial with $u$. From $n_{ex}$, we estimate the number of adjacent cells $N_{adjCells}$ and the number of polynomial sharing cells $N_{neighborCells}$ shown as in Eq.(3) and (4).
\[ N_{\text{adCells}} = n \ast \left( \frac{2\pi}{\text{internalangle}} - 2 \right) \ast \sum_{i=0}^{n_{\text{ex}}} i \]
\[ = n \ast \left( \frac{2n}{n-2} - 2 \right) \ast \sum_{i=0}^{n_{\text{ex}}} i \]
\[ \text{with } 3 \leq n \leq 6 \]  \hspace{1cm} (3)

\[ N_{\text{neighborCells}} = (n \ast \left( \frac{2n}{n-2} - 2 \right) \ast \sum_{i=0}^{2n_{\text{ex}}} i \) + 1 \]
\[ \text{with } 3 \leq n \leq 6 \]  \hspace{1cm} (4)

where \text{internalangle} = \left( 1 - \frac{2}{n} \right) \ast \pi

3.2.2 Architect of the system: The architects of the schemes is described in detail in this section.

(1) Pre-distribution phase: A grid of the target field is used to pre-determine the possible deployment position of nodes or the groups of nodes. Each cell of the grid is assigned by a co-ordinate \( C_{i,c} \) denoting row \( i \) and column \( c \). Let \( N \) denote the maximum number of sensor nodes in the network. The setup server randomly generates \( N \) bivariate 1-degree polynomials \( \{f_{i,c}(x,y)\}_{i=0,1,2,..\ldots r-1} \) and assigns \( \{f_{i,c}(x,y)\} \) to cell \( C_{i,c} \).

For each sensor node, the setup server determines its \textbf{home cell}, where the node is expected to locate. Depending on the kinds of the cells: the square grid cells, the hexagonal grid cells, the triangle grid cells, etc., the setup server then discovers the cells adjacent to this node’s home cell. Finally, the setup server distributes to the sensor node its home cell coordinate and the shares of the polynomials for its home cell and the selected cells.

Example, we consider the implement of the hexagonal cell grid case \( n = 6 \) with the length of the cell \( L = 1 \), and the signal range \( R = 2 \), and the maximum deployment error \( e = 3 \), from (2), we calculate \( n_{\text{ex}} \) as follows:

\[
\begin{cases}
    n_{\text{ex}} \geq 1 \ast \left( \frac{R + 2e}{L} \right) - 1 = 2.056 \\
    n_{\text{ex}} \in N
\end{cases}
\]

so \( n_{\text{ex}} = 2 \).

Using the (3) and (4), we have

\[ N_{\text{adCells}} = n \ast \left( \frac{2n}{n-2} - 2 \right) \ast \sum_{i=0}^{n_{\text{ex}}} i \]
\[ = 6 \ast 1 \ast (0 + 1 + 2) \]
\[ = 18. \]

\[ N_{\text{neighborCells}} = (n \ast \left( \frac{2n}{n-2} - 2 \right) \ast \sum_{i=0}^{2n_{\text{ex}}} i \) + 1 \]
\[ = 6 \ast 1 \ast (0 + 1 + 2 + 3) + 1 \]
\[ = 37. \]

(2) Direct Key Establishment phase: After deployment, if two sensor nodes want to setup a pairwise key, they first need to identify a shared bivariate polynomial. If they can find at least one such polynomial, a common pairwise key can be established directly using the basic polynomial-based key pre-distribution as presented in Section 3.1. The simple way is to let the source node disclose its home cell coordinate to the destination node. From the coordinate of the home cell of the source node, the destination node can immediately determine the IDs of polynomial shares the source node has.

4 ANALYSIS

In this section, we analyze our proposed scheme in detail. We focused on the evaluation of three importance criteria for WSN security: connectivity, communication overhead, and security analysis.

4.1 The connectivity

The connectivity of the network or the probability of the direct key establishment is the probability of any two neighboring nodes sharing at least one key. As the target of the scheme, we increase the key sharing probability much higher than the other schemes. Assumed \( n'_u \) is the average number of sensor nodes that can establish a pairwise key with \( u \) directly, \( n_u \) is the average number of sensor nodes that \( u \) can directly communicate with.

Let \( S_{i,c,\text{ neighbourCells}} \) define the set of the sensor nodes that share at least one common polynomial with sensor node \( u \) with its home cell \( C_{i,c} \) and \( \omega \) denote the average sensor deployment density, we apply the formula as mentioned in [2]:

\[ p = \frac{n'_u}{n_u} \sim \frac{\omega \ast S_{i,c,\text{ neighbourCells}}}{\omega \ast S_{\text{commL}}} \]  \hspace{1cm} (5)

where \( S_{\text{commL}} \) denotes the area inside the communication circle with the assumption that the number of the nodes deployed out of this area are equal to the number of the nodes deployed inside with maximum deployment error \( e \). \( S_{\text{commL}} \) is estimated as Eq.(6)

\[ S_{\text{commL}} = \pi (R + e)^2 \]  \hspace{1cm} (6)

From (3), (5) and (6), we have

\[ p = \frac{n'_u}{n_u} \sim \frac{\omega \ast S_{i,c,\text{ neighbourCells}}}{\omega \ast S_{\text{commL}}} \]
\[ = \frac{N_{\text{neighborCells}} \ast Area_{\text{cell}}}{\pi (R + e)^2 \omega} \]
\[ = \frac{((n \ast \left( \frac{2n}{n-2} - 2 \right) \ast \sum_{i=0}^{2n_{\text{ex}}} i \) + 1) \ast Area_{\text{cell}}}{\pi (R + e)^2} \]  \hspace{1cm} (7)

Where \( Area_{\text{cell}} = \frac{n^2 L^2}{4 \tan \frac{\pi}{3}} \).

The probability of the direct key establishment for the polynomial pre-distribution using the square grid in [3] is as follows:

\[ p = \frac{n'_u}{n_u} \sim \frac{13 \omega L^2}{\pi d^2 \omega} \]  \hspace{1cm} (8)

The probability of the direct key establishment for the polynomial pre-distribution using the hexagonal grid in [2] is as follows:

\[ p = \frac{n'_u}{n_u} \sim \frac{57 \sqrt{3} \omega L^2}{2 \pi d^2 \omega} \]  \hspace{1cm} (9)
Where $d_R = (R + e)^2$

Theoretically, the probability of the key sharing is equal to 1. However, it’s not feasible to exactly get the maximum deployment errors. The maximum deployment error definitions in Eq.(2) and Eq.(6) should be different. The first one could be referred to as the expected maximum deployment and the second one could be defined as the real maximum deployment error. From this, we have the simulation result as depicted in Fig. 3. Our scheme achieves the higher probability of the direct key establishment than the closest polynomial pre-distribution scheme [3] where the maximum deployment error alters from 1 to 10. The reason is that, we consider the deployment error and the signal range as the deployment knowledge resulting in the increase of the probability of the polynomial sharing. With the same real deployment error equal to 6, the connectivity of the scheme is nearly 0.17 and 0.5 respectively with $e = 1$ and $e = 2$ while the CPPS is only 0.11. Besides, the larger the expected maximum deployment error is, the higher the connectivity is, because the fewer deployed nodes are overlooked. The connectivity in the case $e = 2$ is higher than $e = 1$.

In Fig. 4, with same side length of the cell $L$, same expected deployment error, same real maximum deployment errors and increasing signal range from 1 to 10, the connectivity in [2],[3] decreases from 1 to 0.15 and 1 to 0.08 respectively. Compared with the improved hexagonal CPPS, the connectivity changes from 0.1 to 1 and with the improved square CPPS, the connectivity alters in around from 0.5 to 1. By considering the signal range while deciding the number of adjacent cells, our schemes tend to achieve better connectivity compared to others.

### 4.2 The communication overhead

When the two neighboring nodes are not connected directly, they should find a route to connect to each other. We need to determine the number of hops required for this route. The highest number of hops in the network is the communication overhead of the whole network. Similar to above analysis, each sensor node can establish two-hop path key with sensor nodes deployed in its 61 adjacent hexagons in the hexagon-based key pre-distribution scheme [2] and 41 adjacent squares in the closest polynomial pre-distribution scheme [3]. The communication overhead is equal to

$$\text{overhead} = \frac{((n \times \left\lceil \frac{2n}{n-2} \right\rceil - 2) \times \sum_{i=0}^{4n} i + 1) \times \text{Area}_{cell}}{\pi (R + e)^2}$$

(10)

The analytical results are depicted in Fig. 5. When real maximum deployment error alters from 1 to 10, 2-hop communication over-head in our scheme increases and higher than [2] and [3]. With $R = 5$, 2-hop communication overhead in [2] is 0.25 and in [3] is 0.05, but in our scheme, it is equal to 0.85 and 0.8, respectively for the square and hexagonal cases.

### 4.3 Security Analysis

Let $p_e$ denote the probability of each node being compromised, $x$ defines the number of sensor nodes have
been compromised among \( N_s \) sensor nodes that have polynomial shares of a particular cell. We apply the same method used in [3], the estimation of the expected fraction of total keys being compromised is calculated by Eq. (11):

\[
P_c(x) = \frac{N_s}{(N_s - x)} P_c(x)_{N_s-x}
\]

In that, \( N_s \) is calculated as following:

\[
N_s = \frac{N_{adCells}(m + 1) \times Area_{cell}}{\pi R^2}
\]

From (11) and (12), we find that \( m \) is ratio to the compromised fraction. It means that the more the number of nodes know about the polynomial sharing, the weaker the resilience of the network. As mentioned in section 3.2, the existing CPPS only considered a fixed number of the adjacent cells. It means that, even if we want to maintain a uniform higher probability of direct key establishment, there is no way to increasing the length of cells, and accordingly, the signal range of cells increases, accordingly, the security of network is also reduced because of the large \( m \).

For example: In [3], if we have the signal range \( R = 2 \), and \( e = 1 \), then to maintain the connectivity equal to 1, we should keep \( L = 3 \), and so \( m \approx 13 \times 3^2 = 117 \). In our scheme, let \( f(x, y) = 1 \), apply to (2) with \( L = 2 \), we have \( n_{ex} = 1 \) then use (3) and (4), we get \( N_{adCells} = 8 \) and \( N_{cells} = 25 \), so \( m = 25 \times 2^2 = 100 \). It is clear that our scheme require a smaller number of neighbor nodes to share information about the polynomial.

Briefly, by using signal range and deployment error as the deployment knowledge, we optimize the number of the neighbor cells that results in strengthening the security of network.

5 Conclusion

In this paper, we have exploited the advantages of deployment knowledge with regard to signal range and their deployment error in order to find their relationship with location information. With the above knowledge, our scheme was found to extremely increase the connectivity of the network, the communication overhead whilst maintaining a good connectivity. We also compared our present scheme with the existing ones and observed that our scheme achieved a superior performance even with different kinds of the grid cell. In future work, we can focus on analyzing polynomial pre-distribution scheme for the spherical deployment or unequal grid cell. Another challenging direction for our future research is to exploit the relationship between the size of the sensor group and the signal range of the sensor for improving the performance of the network.

Acknowledgment

This research was supported by the MKE (Ministry of Knowledge Economy, Korea, under the ITRC(Information Technology Research Center) support program supervised by the ITTA(Institute of Information Technology Advancement)" (ITTA-2008-(C1090-0801-0002))

References