SECURITY AND COMMUNICATION NETWORKS
*Security Comm. Networks* 2012; **5**:485–495
Published online 12 May 2011 in Wiley Online Library (wileyonlinelibrary.com). DOI: 10.1002/sec.337

RESEARCH ARTICLE

# A beneficial analysis of deployment knowledge for key distribution in wireless sensor networks

Nguyen Thi Thanh Huyen[1], Minho Jo[2]*, Tien-Dung Nguyen[1] and Eui-Nam Huh[1]*

[1] Department of Computer Engineering, Kyung Hee University, Seocheon-Dong, Giheung-Gu, Yongin, Gyeonggi-Do 446–701, South Korea
[2] College of Information and Communications, Korea University, Seoul 136–701, South Korea

## ABSTRACT

Because of the resource constraints on sensor networks, many recent key management schemes exploit the location of nodes in order to establish pairwise keys. Despite these efforts, location discovery for sensor networks is very difficult because existing schemes have failed to consider changes in the signal range and the deployment error. As a result, network performance may accidentally decrease, whereas costs increase. In this paper, we used the theory of the signal range and deployment error knowledge to analyze sensor nodes location information. Combining with the probabilistic pre-distribution method, we proposed a scheme that provides high connectivity and secure communication with better communication overhead by defining an efficient number of adjacent cells and determining the adequate length of the cell. With such optimistic results, novel deployment knowledge is also expected to provide superior performance with different types of key distribution schemes. Copyright © 2011 John Wiley & Sons, Ltd.

**KEYWORDS**
polynomial pre-distribution; key pre-distribution; wireless sensor network; key management

***Correspondence**
Minho Jo, College of Information and Communications, Korea University, 5-1 Anam-dong, Seongbuk-gu, Seoul 136–701, South Korea.
Eui-Nam Huh, Kyung Hee University, Seocheon-Dong, Giheung-Gu, Yongin, Gyeonggi-Do 446–701, South Korea.
E-mail: minhojo@korea.ac.kr; johnhuh@khu.ac.kr

## 1. INTRODUCTION

Sensor networks are comprised of a large number of sensor nodes that collect environmental data, such as temperature, humidity, and pressure. The attained data are used in a wide array of applications, including military sensing and tracking, health monitoring, disaster recovery, and forest fire detection. When deployed in hostile environments for a specific application, an adversary may capture sensor nodes and thus control the packet. Therefore, security services are essential in preserving the confidentiality, integrity, and availability of transmitted data [1]. Providing security in wireless sensor networks (WSNs) is more difficult than in conventional wired networks because of the insecure nature of wireless communication, the lack of a fixed infrastructure, the vulnerability of nodes to physical capture, constraints on resources that require great focus on the consumption of energy, the computations involved, and the memory resources. Key management is one of the most important factors in the authentication and encryption of sensitive data [2]. WSNs are expected to be deployed on a scale that is much larger than their traditional embedded counterparts. This scenario, coupled with the aforementioned operational constraints, makes secure key management an absolute necessity in most WSN designs. Based on the cryptography protocol, there are two types of key management systems: public key management and symmetric key management. Despite recent efforts [5–7] aimed at reducing the computation and energy costs of public key operations, public key protocols, which are successfully applied in traditional wired networks, are not suitable in low-power devices, such as WSNs. The main obstacle hindering private key operations is the high cost. Therefore, symmetric cryptography is a common choice when the computational complexity of asymmetric cryptography [3] is too expensive. In terms of speed and low energy costs, symmetric key cryptography is superior to public key cryptography. However, key distribution schemes based on symmetric key cryptography are not perfect. Firstly, the schemes are derived from a major shortcoming of the key exchange protocol. Existing symmetric key protocols are weak with regard to supporting authentication between communicating nodes.

The main challenge is ensuring that a shared key is shared not from an eavesdropper but between two sensors who wish to communicate. Secondly, we need to consider efficient and flexible key distribution schemes that are described as a trade-off between the probability of establishing a common key between communicating nodes, the ability to tolerate compromised nodes, and the memory available on the sensor nodes [4]. For large-scale WSNs, polynomial key pre-distribution [1] has emerged as an efficient solution because of the fact that it integrates both the encryption and authentication functions in polynomial keys and then provides strong resilience. The second constraint on the criteria of key distribution schemes can be improved from a realization about the lifespan of the sensor node. The energy consumption in sensor nodes can be categorized into three parts [8]:

(1) The energy consumed by the sensor nodes due to security. This is related to the processing required for security functions (e.g., encryption, decryption, signing data, and verifying signatures).
(2) The energy required to store security parameters in a secure manner (e.g., cryptographic key storage).
(3) The energy required for communication among sensor nodes.

After deployment, each node needs to process a computation only once in order to establish the share keys. In the polynomial pre-distribution scheme, the storage cost for a polynomial share increases with the group size. However, this cost is, for the most part, not consumed after the initiation. In the meantime, communication is a frequent activity between nodes in the sensor network, and thus, energy consumption is the most costly factor. To reduce this cost, it is necessary to decrease the number of hops between two communicating nodes (i.e., increase the connectivity).

Our scheme uses the key to determine nodes, which are communicating, and the information that is shared is exploited with regard to changes in the signal range and the deployment error. Existing schemes fail to take these changes into account. Thus, the main contributions of this paper are as follows:

(1) The application of knowledge regarding the signal range and the deployment error through use of the Gaussian function. This serves to highlight the communication relationship between nodes in the network.
(2) A modeling of the polynomial pre-distribution for widely used types of polygonal grid cells. From such a modeling, the scheme becomes more flexible with regard to the application of various geographical situations and node density levels.
(3) The proposal of two possible approaches: One defines an efficient number of adjacent cells so as to increase the connectivity of the network and the second approach determines the adequate length of the grid cell in order to reduce unnecessary keys.
(4) A comparison of the aforementioned two approaches in order to draw conclusions about the security, connectivity, and communication overhead.

The remainder of this paper is organized as follows: In Section 2, related works and their drawbacks are examined. The proposed scheme is detailed in Section 3 and analyzed in Section 4. Finally, conclusions are presented in Section 5.

## 2. RELATED WORKS

In this section, previous research related to key management in WSN security is reviewed. Eschenauer and Gligor [9] proposed a basic probabilistic key pre-distribution where each node stores a random subset of keys from a large key pool before deployment. As a result, there is a certain probability that two nodes will share at least one key after deployment. Chan *et al.* [10] extended this scheme in order to significantly enhance the security and resilience of the network. Such a task was accomplished by requiring that two sensor nodes share at least $q$ pre-distributed keys so as to establish a pairwise key. Liu and Ning [11] proposed several schemes that use location information as the pre-deployment knowledge or the post-deployment knowledge. The goal of such schemes was to save memory costs while maintaining a high level of security. Li *et al.* [12] and Delgosha *et al.* [13] extended grid-based key pre-distribution in the closest polynomials scheme of Liu *et al.* [14] to hexagon-based key pre-distribution. The aim of the researchers was to improve the probability of successful key establishment. Du *et al.* [15] considered the priority of deployment packets in order to avoid unnecessary key assignments. A consideration of priority is useful in improving the performance of key management in short-distance peer-to-peer secure communication. Anjum [17] deployed anchor nodes in order to collect location information and distribute keys to the sensor nodes. Park *et al.* [18] designed state-based key management. To reduce energy consumption, Lai *et al.* [5] proposed an overlap key sharing scheme. A survey of key management in ad hoc networks is given in [16]. The latest survey of key distribution mechanisms for WSNs is presented in [20].

## 3. THE PROPOSED SCHEME WITH KEY PRE-DISTRIBUTION

As we concerned in Section 1, the polynomial key pre-distribution is known as an efficient solution because of the combination of encryption and authentication in polynomial keys, so it provides strong resilience. In order to apply this key pre-distribution to our scheme, a modified polynomial-based key pre-distribution is introduced as a

DOI: 10.1002/sec

suitable mechanism to our scheme. The symbols used throughout this work are shown in Table I.

### 3.1. Polynomial-based key pre-distribution

The setup server first randomly generates a pool of bivariate *t*-degree polynomials $f(x,y) = \sum_{i,j=0}^{t} a_{ij}x^i y^j$ over a finite field $F_q$ such that it has the property of $\mathrm{f}(x,y) = \mathrm{f}(y,x)$, where $q$ is a prime number that is large enough to accommodate a cryptographic key, $a_{ij} \in GF(q)$, for $i,j = 0, 1, 2, \ldots, t$ [1]. The setup server then chooses a subset of polynomials and distributes the polynomial shares and polynomial IDs to each node. The expected location of each node can be used to assign the polynomials. In such a case, the grid of the target field is used. Each cell is associated with a unique random bivariate polynomial, and the polynomials are distributed to a node belonging to the cells where the node is expected to be located. To establish pairwise keys after deployment, two sensor nodes must identify a common polynomial that they share by exchanging their polynomial IDs. The polynomial-based scheme is then used to compute the pairwise key if such a common polynomial is identified.

**Table I.** Symbols list.

| Symbol | Meaning |
|---|---|
| $\alpha$ | The error coeffiecient (Gaussian distribution value) |
| $e$, $e_\alpha$ | Maximum deployment error and the deployment error with Gaussian distribution value $\alpha$, respectfully |
| $R$ | The signal range of the node |
| $L$ | The side length of the cell (edge) |
| $n$ | The number of the edges of each polygon cell |
| $N_{adCells}$ | The number of adjacent cells |
| $N_{Cells}$ | The number of polynomial sharing cells |

The architecture of the polynomial-based key pre-distribution scheme using a hexagonal grid is shown in Figure 1. It is assumed that two sensors, $I$ and $I'$, belong to two adjacent cells, $C_{i,j}$ and $C'_{i',j'}$, respectively. The process is as follows:

Step 1. The setup server creates two polynomials, $\mathrm{f}_{i,j}(ID',y)$ and $f_{i',j'}(ID',y)$, to assign for node I and $\mathrm{f}_{i,j}(ID',y)$ and $f_{i',j'}(ID',y)$ to assign for node $I'$.

Step 2. After deployment, if two nodes, $I$ and $I'$, want to communicate, they send their polynomial IDs to each other.

Step 3. From the polynomial IDs, each node can calculate the common keys: $k_{I,I',1}=f_{i,j}(ID,ID')=f_{i,j}(ID',ID)$ and $k_{I,I',2}=f_{i',j'}(ID,ID)=f_{i',j'}(ID',ID)$. A direct key is established in order to encrypt information between the two sensors $k_{I,I'} = k_{I,I',1}||k_{I,I',2}$.

### 3.2. Polynomial-based key pre-distribution with a polygon grid

The deployment limitation of nodes is illustrated in Figure 2. It is assumed that $(\dot{u},\dot{v})$ and $(\bar{u},\bar{v})$ represent the deployment location and the expected location of node $u$ and node $v$, respectively. Both $u$ and $v$ have a maximum deployment error $e$. Therefore, the area where any node $v$, with its expected location, can communicate with node $u$ is inside the communication limitation. The radius of the communication limitation circle is equal to $R+2e$.

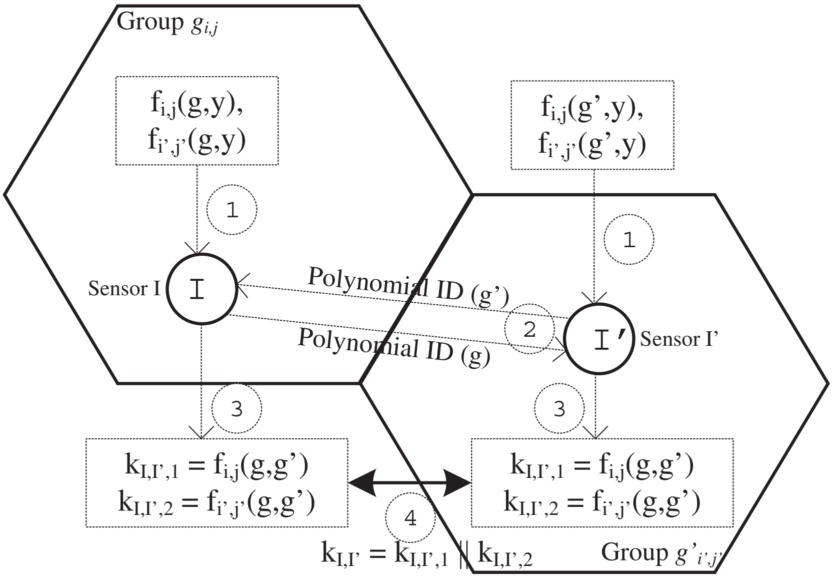

**Figure 1.** The architecture of the polynomial-based key pre-distribution scheme.

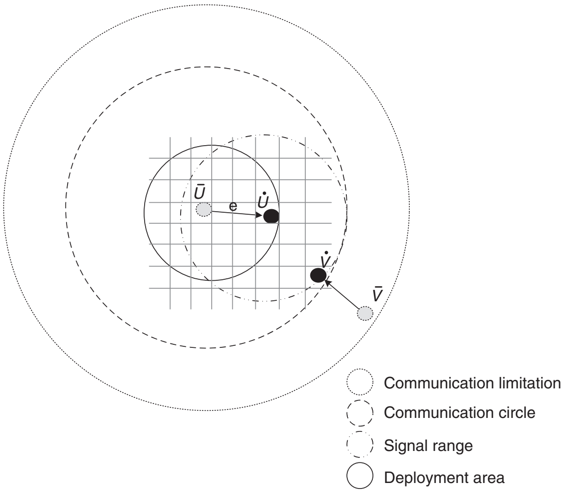

**Figure 2.** The communication limitation of a node.

In the existing polygon coordinate system, the deployment error and the signal range are not considered to be deployment knowledge and then give unsteadily a fixed number of adjacent cells. For this reason, two cases can be considered:

- If the side length of a cell is small but the communication limitation is large, the polynomial sharing area is not sufficiently large, and hence, the key sharing probability is low (see Figure 3(a); the shaded region is the polynomial sharing area).
- When the signal range changes, an increase in the side length leads to an increase in the number of unnecessary sharing polynomials (the cross area). As a result, the resilience of the network is reduced because more nodes know about the polynomial.

In this work, the polynomial-based key pre-distribution scheme is novel in that the signal range and the deployment error were used to determine an efficient number of adjacent cells. This serves to improve the performance of the network.

### 3.2.1. The grid establishment phase.

Depending on the geography conditions and the size of the deployment group, there are suitable polygon grids, such as the triangular grid, square grid, pentagonal grid, or hexagonal grid. Assuming that a regular polygon grid is used, the apothem of the polygon is estimated by

$$Apothem = \frac{L}{2\tan\frac{\pi}{n}} \qquad (1)$$

If $n_{cx}$ denotes the number of adjacent cells of node $u$ along one direction, then the length if $n_{cx}$ apothems should be satisfied by the following expression:

$$\frac{L}{2\tan\frac{\pi}{n}} * (4n_{cx} + 1) \geq (R + 2e_{\alpha}) \qquad (2)$$

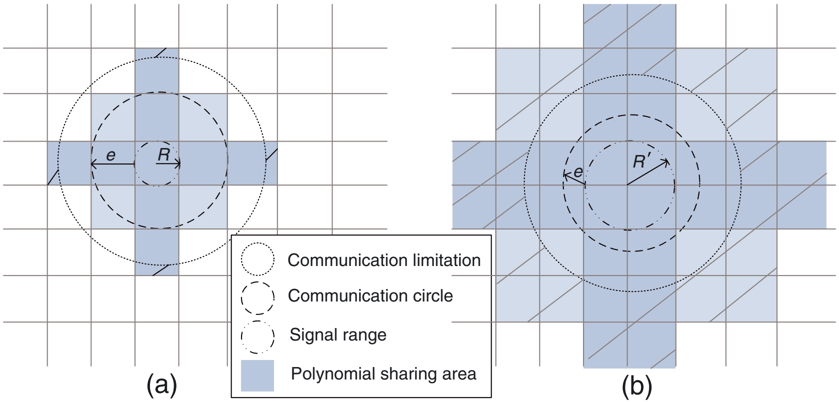

**Figure 3.** One example of image differencing: (a) original image and (b) differencing table.

The expression in Equation (2) is defined by the nodes $v$ that can communicate with node $u$ and share at least one polynomial with $u$. From $n_{cx}$ and the number of polynomial sharing cells, $N_{\text{adCells}}$, can be expressed, respectively, as follows:

$$\begin{aligned} N_{\text{adCells}} &= n * \left(\frac{2\pi}{internal\_angle} - 2\right) * \sum_{i=0}^{n_{cx}} i \\ &= n * \left(\left\lfloor\frac{2n}{n-2}\right\rfloor - 2\right) * \sum_{i=0}^{n_{cx}} i \end{aligned} \tag{3}$$

with $3 \leq n \leq 6$

$$N_{\text{neighborCells}} = \left(n * \left(\left\lfloor\frac{2n}{n-2}\right\rfloor - 2\right) * \sum_{i=0}^{2n_{cx}} i\right) + 1 \tag{4}$$

with $3 \leq \mathrm{n} \leq 6$

where $internal\_angle = (1 - (2/n)) * \pi$.

### 3.2.2. The Gaussian distribution density.

As shown in Equation (2), the maximum deployment error was used to define the number of adjacent cells. To determine the maximum deployment error, the Gaussian distribution function was employed. This function can be expressed as follows:

$$f(x,y) = a * e^{-\frac{(x-x_i)^2 + (y-y_i)^2}{2c^2}} \tag{5}$$

The parameter $a$ denotes the height of the Gaussian peak, $b(x_i, y_i)$ is the position of the center of the peak, and $c$ controls the width of the "bump". We try to keep the nodes to be deployed in their home cell so that each sensor node has a probability $1/\pi t^2$ to be deployed at the expected area. Such a condition leads to $a = 1/\pi t^2$ and $c = t$. If the coordinate of $(x_i, y_i)$ is $(0, 0)$, then Equation (5) is rewritten as follows:

$$f(x,y) = \frac{1}{\pi L^2} * e^{-\frac{x^2+y^2}{2L^2}} \tag{6}$$

In Equation (6), $\alpha = e^{-\frac{x^2+y^2}{2L^2}}$ is the probability that one node is deployed at a distance far away from the expected location $<(x_i, y_i)$, which is known as the Gaussian distribution value. An example of a Gaussian probability density function with different $x$ and $y$ is shown in Figure 4. The function $e^{-\left[(x)^2+(y)^2\right]/2t^2}$ does not reach zero over all $x$ and $y$. Hence, by setting $e^{-\left[(x)^2+(y)^2\right]/2t^2}$ to be larger than a defined value, we naturally find a deployment error $e_\alpha^2 = x^2 + y^2$ that can be used in Equation (2) as the "expected maximum deployment error". As the error coefficient becomes smaller, fewer deployed nodes are overlooked. This means that the system must consider an area with a small deployment probability. Therefore, the choice of a suitable $\alpha$ value is an important issue when using a Gaussian distribution.

### 3.2.3. Architecture of the system.

*3.2.3.1. Pre-distribution.* Two different approaches that result in two distinct architectures for the pre-distribution phase are used.

Scheme 1: defining an efficient number of adjacent cells. A grid of the target field is used to pre-determine the possible deployment position of nodes or groups of nodes. Each cell of the grid is assigned a coordinate $C_{(i_c,i_r)}$ denoting row $i_r$ and column $i_c$. Let $N$ define the maximum number of sensor nodes in the network. The setup server randomly generates $N$ bivariate $t$-degree polynomials $\left\{f_{i_c,i_r}(x,y)\right\}_{i_{c=0,1,2\ldots C-1}, i_{r=0,1,2\ldots R-1}}$ and assigns $\left\{f_{i_c,i_r}(x,y)\right\}$ to cell $C_{(i_c,i_r)}$.

For each sensor node, the setup server first determines its *home cell*, where the node is expected to locate.

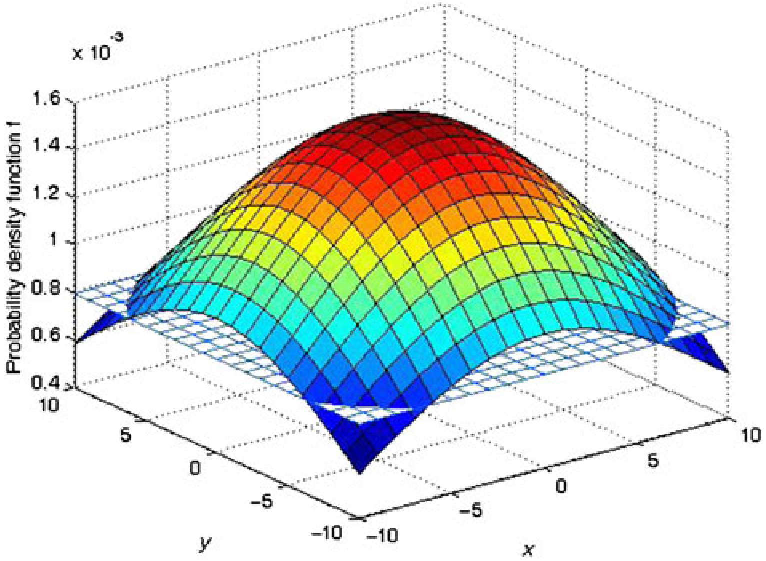

**Figure 4.** The Gaussian probability density function.

Depending on the type of cells (square grid cells, hexagonal grid cells, triangular grid cells, etc.), the setup server discovers the cells adjacent to this node's home cell. The algorithm for the adjacent cell determination is as follows:

Pseudocode: the adjacent cell determination.

```
Begin
$e_\alpha = Gaussian\_distribution(\alpha, L>)$
$n_{cx} = Find\_cells\_x(R, e_\alpha, L, n)$
$n_{adCells} = Find\_adjacent\_cells\_x(n, n_{cx})$
$N_{Cells} = Find\_neighbor\_cells(n, n_{cx})$
End
```

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. Here, we consider the implementation of a hexagonal grid case with $n=6$, cell length $L=1$, and signal range $R=2$. Let $\alpha = e^{-\frac{x^2+y^2}{2L^2}} \geq 0.3$; this means that one does not need to consider the area where the node has a deployment probability lower than $\alpha=0.3$. Thus, $e^2 \leq -t^2 \ln 0.3 = 1.203$ and $e=1.09$. Using Equation (6), we have $n_{cx} \geq 1.91$. Because the error coefficient is equal to 0.3, use of Equations (5) and (6) gives $N_{\text{adCells}}=6$ and $N_{\text{Cells}}=19$.

For example, node $u$ is expected to be deployed in cell $C_{2,2}$, as shown in Figure 5. Obviously, cell $C_{2,2}$ is its home cell, and cells $C_{1,1}$, $C_{1,2}$, $C_{1,3}$, $C_{2,1}$, $C_{3,2}$, and $C_{2,3}$ are the six cells adjacent to its home cell. Thus, the setup sever gives this node the coordinate (2,2) and the polynomial shares $\mathrm{f}_{1,1}(u,y)$, $\mathrm{f}_{1,2}(u,y)$, $\mathrm{f}_{1,3}(u,y)$, $\mathrm{f}_{2,1}(u,y)$, $\mathrm{f}_{3,2}(u,y)$, and $\mathrm{f}_{2,3}(u,y)$.

Scheme 2: determining an adequate length for the grid. A grid of the target field is used to pre-determine the possible deployment position of nodes or groups of nodes.

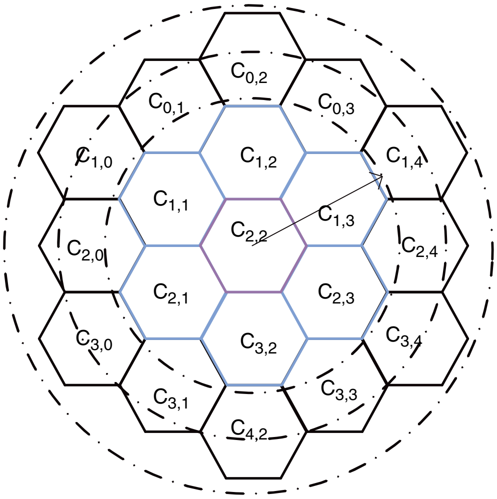

**Figure 5.** A hexagonal grid with six adjacent cells.

Each cell of the grid is assigned a coordinate $C(i_c, i_r)$ denoting row $i_r$ and column. The length of the cell is calculated using Equations (2) and (6) as follows:

$$L \geq \frac{2R\tan\frac{\pi}{n}}{(4n_{cx}+1)-2\sqrt{-2\ln\alpha}} \tag{7}$$

where $n_{cx}=1$ in the minimum case. Let $N$ define the maximum number of sensor nodes in the network.

The setup server randomly generates $N$ bivariate $t$-degree polynomials $\{f_{i_c,i_r}(x,y)\}_{i_c=0,1\ldots C_{o-1},,i_r=0,1\ldots R_{o-1}}$ and assigns $f_{i_C,i_r}(x,y)$ to cell $C(i_C, i_r)$. For each sensor node, the setup server first determines its home cell where the node is expected to locate. Depending on the type of cells (square grid cells, hexagonal grid cells, triangular grid cells, etc.), the setup server 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.

*3.2.3.2. Direct key establishment.* If two sensor nodes want to setup a pairwise key after deployment, they first need to identify a shared bivariate polynomial. If they can find at least one such polynomial, then a common pairwise key can be directly established using basic polynomial-based key pre-distribution. A simple method is to let the source node disclose its home cell coordinate to the destination node. From the coordinates of the home cell of the source node, the destination node can immediately determine the IDs of the polynomial shares possessed by the source node.

## 4. ANALYSIS

In this section, the scheme proposed in this work is analyzed. The focus is on the evaluation of three important criteria for WSNs: connectivity, communication overhead, and security.

### 4.1. Connectivity (the probability of direct key establishment)

The connectivity of a network is the probability that any two neighboring nodes share at least one key. As the target of the scheme, the key sharing probability is increased to a much higher level than in other schemes. The probability of direct key establishment is given in [11,19,21] as follows:

$$p = \frac{n_u^{'}}{n_u} = \frac{\sum_{G_{j_C,j_R} \in S_{i_C,i_R}} \Pr\left(||\dot{u},\ \dot{\upsilon}|| \leq r | C_{i_C,i_R}, C_{j_C,j_R}\right)}{\sum_{\forall G_{j_C,j_R}} \Pr\left(||\dot{u},\ \dot{\upsilon}|| r | C_{ic.i_R}, C_{jC,jR}\right)} \tag{8}$$

where $n'_u$ is the average number of sensor nodes that can directly establish a pairwise key with $u$, $n_u$ is the average number of sensor nodes that $u$ can directly communicate with, and $S_{(i_C,i_R)}$ is the set of hexagons of the sensor nodes that share at least one common polynomial with sensor node $u$.

Let $\overline{\varpi}$ denote the average sensor deployment density. The application of Equation (8), as mentioned in [14], gives the following:

$$p = \frac{n'_u}{n_u} \sim \frac{\overline{\varpi} * S_{i_C,i_R}}{\overline{\varpi} * S_{\text{CommL}}} \tag{9}$$

where $S_{\text{CommL}}$ denotes the area inside the communication circle with the assumption that the number of nodes deployed outside this area is equal to the number of nodes deployed when the deployment error is $e$.

The $S_{\text{CommL}}$ term may be estimated as follows:

$$S_{\text{CommL}} = \pi(R+e)^2 \tag{10}$$

where $e_{0.05}$is the maximum deployment error when $\alpha = 0.05$. From Equations (6), (9), and (10), we have

$$n_u \sim \frac{S_{i_C,i_R}}{S_{\text{Comm}}} = \frac{(N_{\text{Cells}}+1) * Area\_cell}{\pi(r+2e)^2\ \overline{\omega}}$$
$$= \frac{\left(\left(n * \left(\left\lfloor \frac{2n}{n-2} \right\rfloor - 2\right) * \sum_{i=0}^{n_{cx}} i\right) + 1\right) * Area\_cell}{\pi(R+e)^2\ \overline{\omega}} \tag{11}$$

where *Area_cell* is the area of a cell.

The probability of direct key establishment for the polynomial pre-distribution scheme using the square grid in [11] is as follows:

$$p' = \frac{n'_u}{n_u} \sim \frac{13\,\overline{\varpi}\,t^2}{\pi d_r^2\,\overline{\varpi}} \tag{12}$$

The probability of direct key establishment for the polynomial pre-distribution scheme using the hexagonal grid in [14] is as follows:

$$p' = \frac{n'_u}{n_u} \sim \frac{57\sqrt{3}\,\overline{\varpi}\,t^2}{2\pi d_r^2\,\overline{\varpi}} \tag{13}$$

where $d_r = (R+e)^2$.

The simulation depicted in Figure 6 illustrates a comparison between our proposed schemes and the schemes detailed in [11]. When the maximum deployment error is varied from 1 to 10, a higher probability of direct key establishment is yielded with both of our schemes than that attained with the closest polynomial pre-distribution scheme (CPPS). The reason for the increase in the probability of sharing the polynomial is that we consider the deployment error and the signal range as deployment knowledge. Using an identical deployment error of $e = 6$, the connectivity of scheme 1 and scheme 2 is nearly 0.2, whereas that of the CPPS is only 0.11.

The simulation results attained using Equations (8) and (9) yield plots that are quite close to one another. Hence, the CPPS (the original) using Equation (8) is similar to the CPPS (the relativity) using Equation (9). The cause for the difference is the assumption regarding the uniform distribution of nodes in the network. As mentioned in Section 3.2.2, the distribution, in reality, is unequal. The density of sensor nodes around the expected target is higher than in other areas. Thus, Equation (9) should be rewritten as follows:

$$p = \frac{n'_u}{n_u} \sim \frac{\bar{\varpi}_1 * S_{i_C,i_R}}{\bar{\varpi}_2 * S_{\text{CommL}}} \tag{14}$$

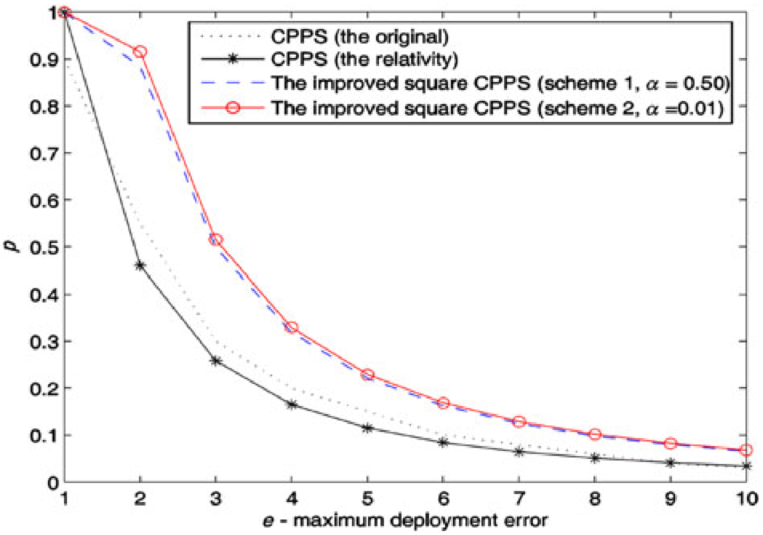

**Figure 6.** The connectivity versus the maximum deployment error with the square grid cell. CPPS, closest polynomial pre-distribution scheme.

DOI: 10.1002/sec

Because $S_{i_C,i_R} \leq S_{\text{CommL}}$ and $\varpi_1 \leq \varpi_2$, we have

$$p = \frac{n'_u}{n_u} \sim \frac{\varpi_1 * S_{i_C,i_R}}{\varpi_2 * S_{\text{CommL}}} \leq \frac{\varpi * S_{i_C,i_R}}{\varpi * S_{\text{CommL}}} \tag{15}$$

From Equation (15), one can deduce why the connectivity result from the relativity formula is higher than the result from the original formula. The number of sensor nodes is huge, and thus, Equation (9) may be used without significant variance in the result.

Shown in Figure 7 is a plot of the connectivity versus the signal range at an identical Gaussian distribution value *a* and maximum deployment error *e*. As the signal range is increased from 1 to 10, the connectivity in [14] and [11] decreases from 1 to 0.1 and 1 to 0.03, respectively. When compared with scheme 1, the connectivity of the hexagonal CPPS is increased from 0.4 to 1, and the connectivity of the square CPPS is raised from 0.6 to 1. By considering the signal range when deciding the number of adjacent cells, scheme 1 achieves better connectivity. The simulation results also show that scheme 2 achieves a higher connectivity than that attained with scheme 1 when the signal range is larger. In the square CPPS, when the signal range is larger than 6, the connectivity of scheme 2 is equal to 1.0, which is higher than the 0.95 value attained from scheme 1. The results also reveal that the connectivity in scheme 2 is more stable because the graph from this scheme increases smoothly. The connectivity of the square CPPS in scheme 1 varies from 1.0 at $R = 4$ to 0.4 at $R = 7$, then increases to 0.86 at $R = 8$. In scheme 2, the connectivity steadily increases.

The connectivity versus maximum deployment error with schemes 1 and 2 is shown in Figures 8 and 9, respectively. As shown in Figure 8, when the maximum deployment error is varied, the connectivity of the hexagonal CPPS decreases from 1 to 0.32 in scheme 1 and from 1 to 0.10 in scheme 2. However, as evident in Figure 9, the connectivity of the hexagonal CPPS decreases from 1 to 0.25 in scheme 1 and from 1 to 0.40 in scheme 2. Thus, the change in the signal range is the only reason why the connectivity of scheme 2 is lower or higher than scheme 1. In other words, scheme 2 is more suitable for cases where the signal range is large, whereas scheme 1 works better with a small signal range.

## 4.2. The communication overhead

When two neighboring nodes are not directly connected, a route to connect the nodes should be found. The number of hops required for this route will need to be determined. The highest number of hops in the network is the communication overhead of the entire network. Each sensor node can establish a two-hop path key with the sensor nodes deployed in its 61 adjacent hexagons (in the hexagon-based key pre-distribution scheme [14]) or its 41 adjacent squares (in the CPPS [11]). In the proposed scheme, the number of cells for this area is equal to

$$p_{\text{overhead}} = \frac{\left(\left(n * \left(\left\lfloor \frac{2n}{n-2} \right\rfloor - 2\right) * \sum_{i=0}^{4*n_{cx}} i\right) + 1\right) * Area\_cell}{\pi (R+e)^2} \tag{16}$$

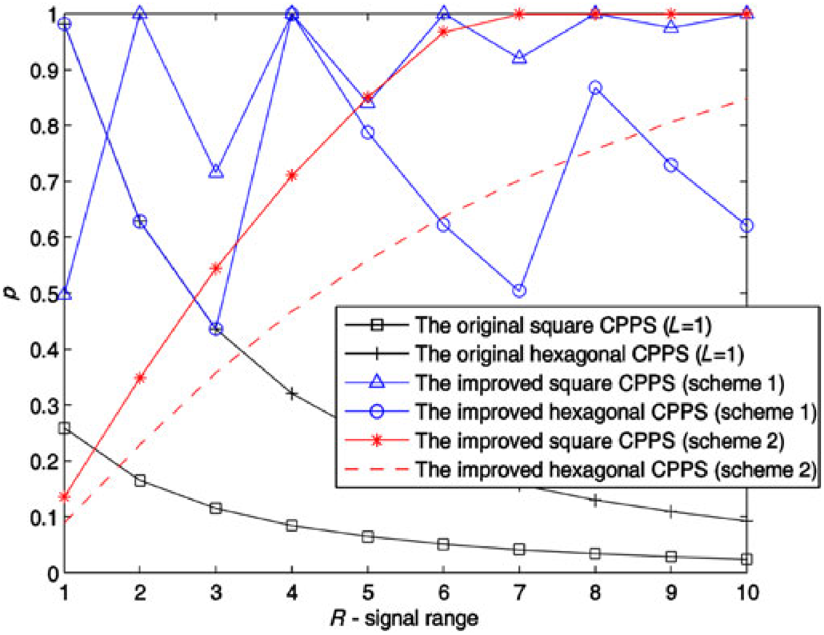

**Figure 7.** The connectivity versus the signal range when $e = 3$ and $a = 0.5$. CPPS, closest polynomial pre-distribution scheme.

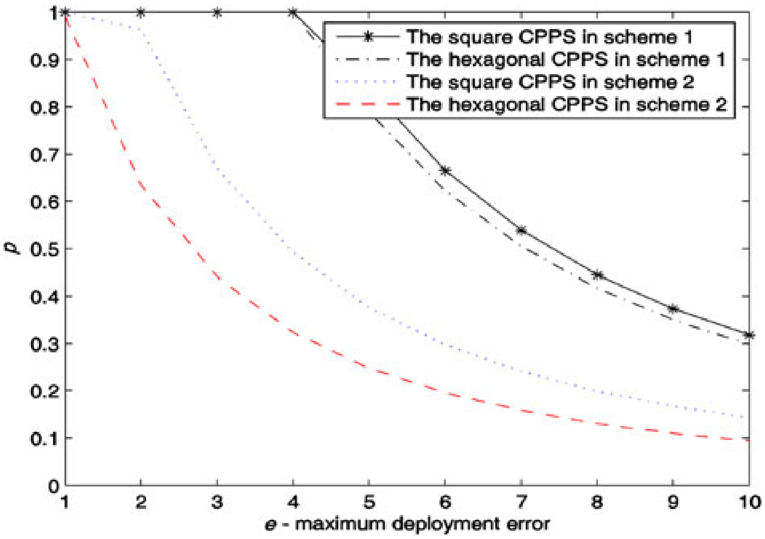

**Figure 8.** The connectivity versus the maximum deployment error for scheme 1 and scheme 2 when $R = 3$. CPPS, closest polynomial pre-distribution scheme.

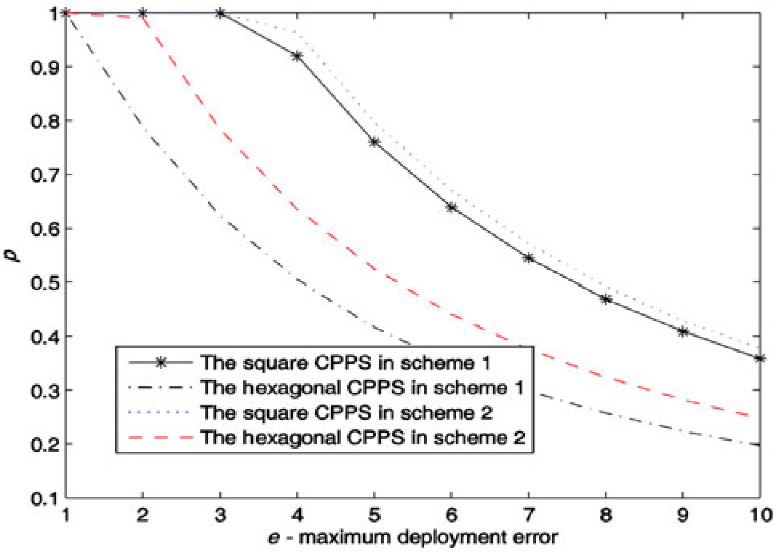

**Figure 9.** The connectivity versus the maximum deployment error for scheme 1 and scheme 2 when R = 6. CPPS, closest polynomial pre-distribution scheme.

One can easily see that the communication overheads of our schemes are higher than those of [14] and [11]. As shown in Figure 10, when $e$ is increased from 1 to 10, the two-hop communication overhead of scheme 2 is equal to one. This means that most of the key setups can be conducted within two hops. For the schemes in [14] and [11], the two-hop communication overhead is 0.5 and 0.14, respectively, when $R = 5$, Thus, the schemes in [14] and [11] require more than two hops to communicate when the signal range is large.

### 4.3. Security analysis

Let $P_c(i)$ denote the probability that each node could be compromised and $i$ define the number of sensor nodes that have been compromised among $N_s$ sensor nodes that have polynomial shares of a particular cell. Applying the same method used in [10], an estimation of the expected fraction of the total keys being compromised is calculated as follows:

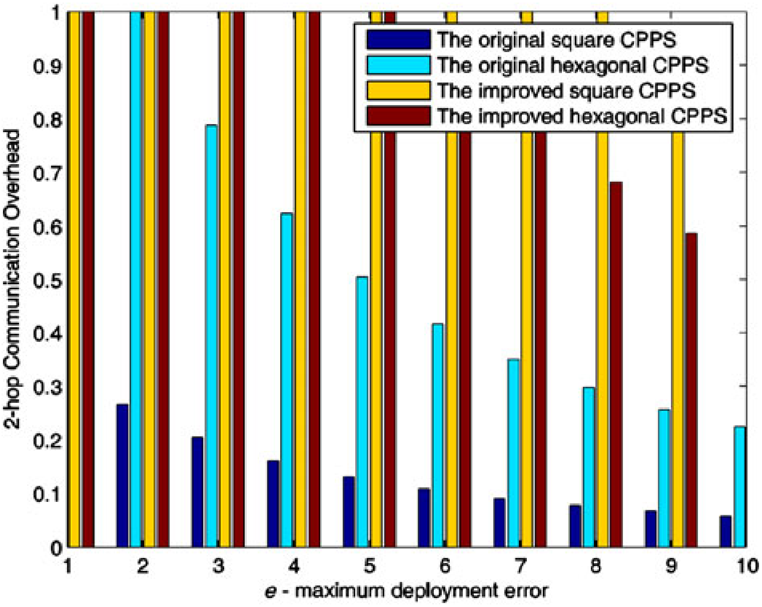

**Figure 10.** Communication overhead (|Á = 0.5). CPPS, closest polynomial pre-distribution scheme.

$$P_c(i) = \frac{N_S!}{(N_S-i)!i!} p_c^i (1-p_c)^{N_S-i} \tag{17}$$

In Equation (17), $N_s$ is calculated as follows:

$$N_S = \frac{N_{\text{adCells}} * (m+1) * Area\_cell}{\pi R^2} \tag{18}$$

where $m$ is the number of neighbor nodes. From Equations (17) and (18), it can be found that $m$ is the ratio to the compromised fraction. As the number of nodes that know about the polynomial sharing increases, the resilience of the network decreases. Because scheme 2 exploits deployment knowledge in order to optimize the length of the cell, the number of nodes sharing information about the polynomial is reduced to a minimum. As a natural consequence, the security of the network increases.

An example: In [11], if $R = 3$ and $e = 1$, $L$ should be equal to 4 if the connectivity is to be maintained at 1. Thus, $m = \ \bar{\varpi}*13*42 = 208\ \bar{\varpi}$. In scheme 2, $\alpha = 0.5$ and the connectivity is about 0.99. With the application of Equations (3) and (7), we have $L = 2.68$.

If $L = 2.7$, then $m = \ \bar{\varpi}*25*(2.7)2 = 135\ \bar{\varpi}$. Correspondingly, scheme 2 requires a smaller number of neighboring nodes in order to share information about the polynomial.

### 4.4. Summary

Scheme 1 leads to higher connectivity by increasing the number of adjacent cells. This, in turn, leads to more nodes with knowledge about the polynomial. With this characteristic, scheme 1 can be applied to applications that focus on the ability of key establishment, such as in transport.

The analysis revealed that the use of the signal range and the deployment error as deployment knowledge in scheme 2 optimizes the length of the cells. This results in better resilience against a compromised node attack.

## 5. CONCLUSIONS

In this paper, the signal range and the deployment error were exploited using a Gaussian distribution in order to find their relationship with location information. With such knowledge, we proposed two approaches for the polynomial pre-distribution scheme: defining an efficient number of adjacent cells and determining the adequate length of the grid cell. When compared with existing schemes, our schemes significantly increased both the connectivity of the network and the communication overhead, while maintaining moderate security. The attained results provide a new direction for applying deployment knowledge in order to improve the performance of other key distribution schemes.

## ACKNOWLEDGEMENT

This work was supported by the Korea Science and Engineering Foundation grant funded by the Korea government (Ministry of Education, Science and Technology) (no. 2010–0016959) and the Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology.

## REFERENCES

1. Blundo C, De Santis A, Herzberg A, Kutten S, Vaccaro U, Yung M. Perfectly-secure key distribution for dynamic conferences. In *Proceedings of the 12th Annual International Cryptology Conference on Advances in Cryptology, CRYPTO'92*, LNCS 740, Springer-Verlag: London, 1993; 471–486.
2. Huh H-N, Sultnan N. Application driven cluster based group key management with identifier in wireless sensor networks. *KSII Transactions on Internet and Information Systems* 2007; **1**: 5–18.
3. Haque MM, Pathan A-SK, Hong CS, nam Huh E. An asymmetric key-based security architecture for wireless sensor networks. *KSII Transactions on Internet and Information Systems* 2008; **2**(5): 265–279.
4. Mohaisen A, Nyang D, Maeng Y, Lee K, Hong D. Grid-based key pre-distribution in wireless sensor networks. *KSII Transactions on Internet and Information Systems* 2009; **3**(2): 195–208.
5. Lai B-CC, Hwang D, Kim SP, Verbauwhede I. Reducing radio energy consumption of key management protocols for wireless sensor networks. In *Proceedings of the 2004 International Symposium on Low Power Electronics and Design, ISLPED'04*, ACM: New York, NY, 2004; 351–356.
6. Liu D, Ning P. Establishing pairwise keys in distributed sensor networks. In *Proceedings of the 2008 International Conference on Information Processing in Sensor Networks, IPSN'08*, Washington, DC, USA, ACM: New York, NY, 2008; 245–256.
7. Zhu S, Xu S, Setia S, Jajodia S. Establishing pairwise keys for secure communication in ad hoc networks: a probabilistic approach. In *Proceedings of the 11th IEEE International Conference on Network Protocols, ICNP'03*, IEEE Computer Society: Washington, DC, 2003; 326–335.
8. Watro R, Kong D, Cuti S, Gardiner C, Lynn C, Kruus P. TinyPK: securing sensor networks with public key technology. In *Proceedings of the Second ACM Workshop on Security of Ad Hoc and Sensor Networks, SASN'04*, ACM Press: New York, NY, 2006; 865–882.
9. Eschenauer L, Gligor VD. A key-management scheme for distributed sensor networks. In *Proceedings of the Ninth ACM Conference on Computer and Communications Security, CCS'02*, ACM Press: New York, NY, 2002; 41–47.
10. Chan H, Perrig A, Song DX. Random key predistribution schemes for sensor networks. In *Proceedings of the 2003 IEEE Symposium on Security and Privacy, SP'03*, IEEE Computer Society: Washington, DC, 2003; **197**.
11. Liu D, Ning P. Improving key predistribution with deployment knowledge in static sensor networks. *ACM Transactions on Sensor Networks* 2005; **1**(2): 204–239.
12. Li G, He J, Fu Y. A hexagon-based key predistribution scheme in sensor networks. In *Proceedings of the 2006 International Conference Workshops on Parallel Processing, ICPPW'06*, IEEE Computer Society: Washington, DC, 2006; 175–180.
13. Delgosha F, Ayday E, Fekri F. MKPS: a multivariate polynomial scheme for symmetric key-establishment in distributed sensor networks. In *Proceedings of the 2007 International Conference on Wireless Communications and Mobile Computing, IWCMC'07*, 2007; 236–241.
14. Liu A, Ning P. TinyECC: a configurable library for elliptic curve cryptography in wireless sensor networks. In *Proceedings of the 2008 International Conference on Information Processing in Sensor Networks, IPSN'08*, IEEE Computer Society: Washington, DC, 2008; 245–256.
15. Du W, Deng J, Han YS, Varshney PK. In A key management scheme for wireless sensor networks using deployment knowledge. In *Proceedings of the 23rd Annual Joint Conference of the IEEE Computer and Communications Societies, INFOCOM'04*, IEEE Computer Society: Washington, DC, 2004; 586–597.
16. Hegland AM, Winjum E, Mjolsnes SF, Rong C, Kure O, Spilling P. A survey of key management in ad hoc networks. *IEEE Communications Surveys & Tutorials* 2006; **8**(3): 48–66.
17. Anjum F. Location dependent key management in sensor networks without using deployment knowledge. In *Proceedings of the Second International Conference on Communication Systems Software and Middleware, COMSWARE'07*, Bangalore, 7–12 January 2007.
18. Park J, Kim Z, Kim K. State-based key management scheme for wireless sensor networks. In *Proceedings of the 2005 IEEE International Conference on Mobile Adhoc and Sensor Systems Conference*, IEEE Computer Society: Washington, DC, 2005; 7–10.
19. Huyen NT, Huh E. An efficient signal range based key pre-distribution scheme ensuring the high connectivity in wireless sensor network. In *Proceedings of the Second International Conference on Ubiquitous Information Management and Communication, ICUIMC'08*, ACM Press: New York, NY, 2008; 441–447.
20. Camtepe SA, Yener B. Key distribution mechanisms for wireless sensor networks: a survey. In *Proceedings of the 19th International Conference on Pattern Recognition*, Orlando, FL, USA, 4–6 December 2008.
21. Karlof C, Sastry N, Wagner D. *TinySec: a link layer security architecture for wireless sensor networks*. Proceedings of the Second International Conference on Embedded Networked Sensor Systems, SenSys'04, ACM Press: New York, NY, 2004; 162–175.