Front. Electr. Electron. Eng. China 2009, 4(3): 300–306
DOI 10.1007/s11460-009-0034-x

RESEARCH ARTICLE

Hangyang DAI, Hongbing XU

Triangle-based key management scheme for wireless sensor networks

Abstract For security services in wireless sensor networks, key management is a fundamental building block. In this article, we propose a triangle-based key predistribution approach and show that it can improve the effectiveness of key management in wireless sensor networks. This is achieved by using the bivariate polynomial in a triangle deployment system based on deployment information about expected locations of the sensor nodes. The analysis indicates that this scheme can achieve higher probability of both direct key establishment and indirect key establishment. On the other hand, the security analysis shows that its security against node capture would increase with a decrease of the sensor node deployment density and size of the deployment model and an increase of the polynomial degree.

Keywords wireless sensor networks, security, key management, triangle

1 Introduction

Recent technological advances in wireless sensor networks (WSNs) [1] are fueling interest across a wide range of applications, including battlefield surveillance, environment monitoring, scientific exploration, target tracking and sensing, etc. Wireless sensor networks usually consist of a large number of ultra-small autonomous devices. Each device, called a sensor node, is battery-powered and equipped with integrated sensors, data processing, and short-range radio communication capabilities. When WSNs are deployed in a hostile environment, security services/issues [2] such as integrity, authentication and confidentiality, are critical for communication between sensor nodes. As the basic requirement for providing security functionality, key management plays a central role in data encryption and authentication.

Some key management methods used in general wireless networks are not suitable for WSNs. First, public-key algorithms, such as the Diffie-Hellman key agreement or RSA, are not practical in WSNs because of limited computation and energy resources in sensor nodes. Second, online key management by the base station is also unfeasible because of communication overhead. Third, the pairwise key scheme that requires each node to store a distinct pairwise key for any other node is impractical because of its high memory requirement.

There are currently three types of key management schemes commonly used in WSNs: the trusted server scheme, self-enforcing scheme, and key predistribution scheme. The trusted server scheme relies on a trusted server for key management, e.g., Kerberos. This scheme is not very suitable for WSNs because there is usually a lack of trusted infrastructure in application environments in which WSNs are used. The second type of key management scheme, the self-enforcing scheme depends on asymmetric cryptography, such as key distribution and management using public key certificates. However, limited computation and energy resources in sensor nodes usually make it undesirable to use public key algorithms, such as Diffie-Hellman key agreement or RSA, for the sake of energy conservation. Third, the key predistribution scheme is an approach in which cryptographic keys are predistributed among all sensor nodes prior to deployment.

1.1 Related works

Recently, several key predistribution protocols have been developed to address the key agreement problems in WSNs. Eschenauer and Gligor [3] proposed a random key predistribution scheme (E-G scheme) based on Random Graph Theory [4] and Probability Theory [5]. According to this scheme, each sensor node receives a random subset of

Received July 2, 2008; accepted October 15, 2008

Hangyang DAI (✉), Hongbing XU
School of Automation Engineering, University of Electronic Science and Technology of China, Chengdu 610054, China
E-mail: daihang1981@sina.com

keys from a large key pool as the node's key ring before deployment and stores them in its memory. After the sensor nodes have been deployed in a designated area, two neighboring nodes will find at least one common key in their key rings and use the keys as their shared keys. Chan et al. [6] further improved the E-G scheme and developed the q-composite key establishment scheme and random pairwise key scheme. The q-composite key establishment scheme requires two sensor nodes to share at least q pre-computed keys as the basis to establish a pairwise key between them. In the random pairwise key scheme, random pairwise keys are established between a specific sensor node and a random subset of other nodes.

Liu and Ning [7] developed a framework in which pairwise keys are predistributed by using bivariate polynomials. Their scheme takes advantage of expected node locations to predistribute appropriate keys to sensor nodes and thus can improve the performance of key establishment. Moreover, Liu and Ning [8] utilized node deployment knowledge to improve the local secure connectivity. Their approach assumes a group-based deployment model, in which the entire network is divided into many non-overlapping square cells where a group of sensor nodes is deployed in each cell.

However, many key distribution schemes failed to take into account the information on deployment locations and signal propagation. Therefore, they lowered the probability of successful key establishment and thus increase the cost. To address this issue the scheme proposed in this article is based on triangle-based key predistribution. We first describe the scheme in WSNs in which the triangle to simulate signal propagation is used. We then analyze the connectivity of our scheme and show that the triangle-based key predistribution scheme can remarkably improve the probability of successful key establishment. Furthermore, compared with the square-based key predistribution approach in Ref. [8], our scheme significantly enhances security performance.

1.2 Main contributions of our scheme

In this article, we propose an efficient and secure key management scheme based on triangle-based key predistribution for WSNs. The main contributions of this article are summarized as follows:

1) We develop a new framework for key management by combining node deployment knowledge and polynomial-based key predistribution. The proposed scheme can substantially improve the probability of successful key establishment by constructing a sensor cellular network to predistribute the key polynomials.

2) We systematically analyze direct key establishment and multi-hop indirect key establishment in the proposed scheme.

3) The proposed scheme substantially improves network resilience against node capture over existing schemes by changing sensor node deployment density, size of the deployment model and polynomial degree.

The remainder of this article is organized as follows. Section 2 introduces polynomial-based key establishment. Section 3 describes the triangle-based key predistribution scheme. Section 4 provides detailed analysis on network connectivity and security properties. Finally, Sect. 5 concludes this article and points out future research directions.

2 Polynomial-based key predistribution scheme

A polynomial-based key predistribution scheme (Polynomial-based Scheme) over a finite field was proposed by Liu and Ning [7]. Compared to the E-G scheme, the Polynomial-based Scheme has a security threshold and better ability in resilience against node capture.

To predistribute pairwise keys, the key predistribution server first randomly generates a bivariate t-degree polynomial $f(x,y)=\sum_{i,j=0}^{t} a_{ij}x^i y^j$ over a finite field F_q, where q is a prime number large enough to accommodate a cryptographic key such that it has the property of $f(x,y)=f(y,x)$. For each sensor node i, the set-up server then computes a polynomial share of $f(x,y)$, i.e., $f(i,y)$ and stores it in the sensor node i. For any two sensor nodes i, j, node i can compute the pairwise key $f(i,j)$ by evaluating $f(i,y)$ at point j, and node j can compute the pairwise key $f(j,i)$ by evaluating $f(j,y)$ at point i. From the property of symmetry of $f(x,y)$, $f(i,j)=f(j,i)$. Thus, the pairwise key between nodes i and j can be established.

In this scheme, each sensor node needs to store a bivariate t-degree polynomial's coefficients, which would occupy $(t+1)\log_2 q$ storage space. The security proof in Ref. [9] ensures that this scheme is unconditionally secure and t-collision resistant. Thus, the coalition of no more than t compromised sensor nodes knows nothing about the pairwise keys between any two non-compromised sensor nodes.

3 Triangle-based key predistribution scheme

3.1 Deployment model

In this article, the entire network is divided into many non-overlapping triangle cells, each of which is associated with a unique random bivariate polynomial. Instead of assigning each sensor node the pairwise keys for the closest sensor nodes, we distribute to each sensor node a set of polynomial shares that belong to the cells closest to the one

that this sensor node is expected to locate in. Deployment of the network region is shown in Fig. 1.

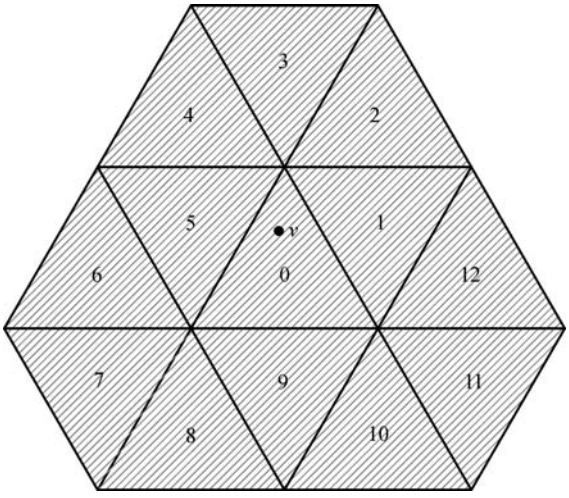

Fig. 1 Triangular deployment region

3.2 Proposed scheme

There are four phases in key management of the proposed scheme: key predistribution, direct key establishment, indirect key establishment, and sensor node revocation and addition. In the key predistribution phase, sensor nodes are initialized by distributing a bivariate polynomial subset built by the key setup server according to the expected locations of the sensor nodes. After deployment, two sensor nodes can establish a direct key between them if they share the same bivariate polynomial. Otherwise, the two sensor nodes should establish the indirect keys with the help of other intermediate nodes. Finally, sensor node revocation and addition ensures the normal operation of the network.

3.2.1 Key predistribution

For each sensor node i, the key set-up server first determines its home triangle T_i, in which the sensor node i is expected to locate. The key set-up server then discovers three cells $\{T_i|i = 1,2,3\}$ adjacent to the sensor node's home triangle. For convenience, the key set-up server assigns a unique ID to each polynomial. Finally, the key set-up server distributes the polynomial shares of the polynomials for its home triangle and its three neighboring triangles and assigns their corresponding IDs to the sensor node.

3.2.2 Direct key establishment

After deployment, if two sensor nodes want to establish a pairwise key, they first need to identify a shared bivariate polynomial. If they can find out at least one such polynomial, a common pairwise key can be directly established by using the polynomial-based key establishment scheme proposed in Sect. 2. To find out whether the two sensor nodes hold the shared polynomial, they should exchange their polynomial IDs. To protect information associated with their polynomial IDs, the nodes may challenge each other to solve puzzles. For example, using the method in Ref. [3], the sensor node i may broadcast an encryption list, $\alpha, E_{\mathrm{ID}_1}(\alpha),...,E_{\mathrm{ID}_i}(\alpha)$, where $\mathrm{ID}_i, i = 1,...,4$ is the ID of the polynomials that sensor node i holds. If the other sensor node can correctly decrypt one of them, they then share the same polynomial and can proceed to establish a direct pairwise key using this shared polynomial.

3.2.3 Indirect key establishment

If direct key establishment fails, the two sensor nodes need to find an intermediate neighbor sensor node that shares pairwise keys with both of them to help establish an indirect key. Otherwise, the intermediate node would broadcast this message continuously until it discovers a sensor node that shares a pairwise key with the two sensor nodes respectively. The indirect key can then be established along the message broadcast path in reverse.

3.2.4 Sensor node revocation and addition

During the operation of the network, it is possible that some sensor nodes are compromised by adversaries. At the same time, the disclosure of bivariate polynomials will compromise the security of key management. If more than t sensor nodes that share the same bivariate polynomial are compromised, this polynomial is no longer considered secure. Hence we should remove this polynomial as well as the IDs of all sensor nodes that share the same polynomial to save memory resources. When some sensor nodes are destroyed, some holes may exist in the network and the security of key management is lowered by disclosing shared polynomial information. Furthermore, some new sensor nodes need to be deployed and predistributed with their own IDs along with the corresponding bivariate polynomial coefficients based on their deployment locations. After deployment, the new sensor nodes can establish a direct key or an indirect key with surrounding nodes as long as they share at least one common bivariate polynomial.

4 Performance analysis

4.1 Network connectivity

In WSNs, network connectivity is determined by the probability of direct and indirect key establishment. In this subsection, we plan to compare the proposed scheme with that in Ref. [8] in terms of network connectivity.

4.1.1 Probability of direct key establishment

Similar to the analysis in Ref. [8], the probability of establishing a common key directly between any sensor node v and its neighboring nodes in the triangle-based key predistribution scheme is

$$P_v = \frac{n_v^s}{n_v} = \frac{\sum_{T_j \in S_i} p(T_j, T_i)}{\sum_{\forall j} p(T_j, T_i)}, \quad (1)$$

where n_v^s is the average number of sensor nodes that can establish a pairwise key with v directly, n_v is the average number of sensor nodes that v can directly communicate with, and S_i is the set of triangles of sensor nodes that share at least one common polynomial with sensor node v.

In our scheme, each sensor node takes its home triangle as the center. There are 13 adjacent triangles where sensor nodes sharing polynomials with the sensor node in its home triangle are deployed. In Fig. 1, all sensor nodes deployed in triangles 1–12 can share common polynomials with the sensor node deployed in triangle 0. Assume that sensor node deployment density in each triangle cell is ρ, the radius of node signal propagation is r, and d is the side length of a triangle cell. The radius of a node's signal propagation is the minimal distance between a sensor node and those that are within the signal range of the sensor node with which a direct pairwise key can be established. Based on our analysis, the relation between r and d is $r = 2d$ in the triangle-based key predistribution scheme. Thus, the probability of directly establishing a common key between the sensor node v and its neighboring nodes in the triangle-based key predistribution scheme is

$$P_v = \frac{n_v^s}{n_v} = \frac{13\rho\frac{\sqrt{3}}{4}d^2}{\pi r^2\rho} = \frac{13\rho\frac{\sqrt{3}}{4}d^2}{\pi(2d)^2\rho} \approx 0.45. \quad (2)$$

On the other hand, the probability of direct key establishment in the square-based key predistribution scheme [8] is

$$P_u = \frac{n_u^s}{n_u} = \frac{13\rho L^2}{\pi r^2\rho} = \frac{13\rho L^2}{\pi(\sqrt{10}L)^2\rho} \approx 0.41, \quad (3)$$

where L is the side length of a square in Fig. 2, and the relation between r and L is $r = \sqrt{10}L$. As shown in Fig. 2, only the sensor nodes deployed in 13 shaded squares can establish direct pairwise keys with the sensor nodes u deployed in $C_{2,2}$.

From the above analysis and calculation, it is shown that the probability of direct key establishment in the triangle-based key predistribution scheme is approximately 10% higher than that in the square-based key predistribution scheme presented in Ref. [8].

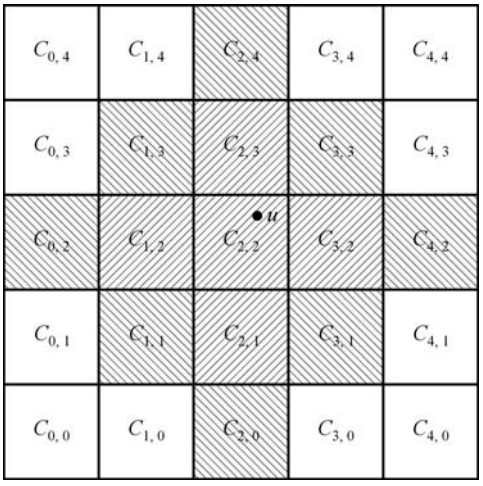

Fig. 2 Square deployment region

4.1.2 Probability of two-hop indirect key establishment

For indirect key establishment, we first discuss the establishment of a two-hop path key between two sensor nodes. Similar to the analysis in direct key establishment, each sensor node can establish an indirect two-hop key with sensor nodes deployed in its 37 adjacent triangles in the triangle-based key predistribution scheme and 41 adjacent squares in the square-based key predistribution scheme. Hence, the probabilities of indirect two-hop key establishment in these two schemes are

$$P_v' = \frac{37\rho\frac{\sqrt{3}}{4}d^2}{\pi r^2\rho} = \frac{37\rho\frac{\sqrt{3}}{4}d^2}{\pi(3d)^2\rho} \approx 0.57, \quad (4)$$

where the minimal distance r means that the sensor node v can establish an indirect two-hop pairwise key with the other nodes, and $r = 3d$;

$$P_u' = \frac{41\rho L^2}{\pi r^2\rho} = \frac{41\rho L^2}{\pi(\sqrt{26}L)^2\rho} \approx 0.50, \quad (5)$$

where $r = \sqrt{26}L$.

As a result, the probability of indirect two-hop key establishment in the triangle-based key predistribution scheme is approximately 10% higher than that in the square-based key predistribution scheme proposed in Ref. [8].

4.1.3 Probability of multi-hop indirect key establishment

If a two-hop path key cannot complete indirect key establishment, we need multi-hop indirect key establishment. Based on two-hop path key sharing, the number of

triangles T_i deployed in the i-hop indirect key establishment in the triangle-based key predistribution scheme and the signal range r_i of one sensor node that can establish an indirect pairwise key with other sensor nodes can be calculated using the following expressions, respectively:

$$\begin{cases} T_0 = 1, \\ T_1 = 13, \\ T_i = T_{i-1} + 12i, \end{cases} \quad \begin{cases} r_0 = d, \\ r_1 = 2d, \\ r_i = (i+1)d. \end{cases} \tag{6}$$

In the square-based key predistribution scheme, a similar calculation is in accord with the formulas below:

$$\begin{cases} S_0 = 1, \\ S_1 = 13, \\ S_i = 2S_{i-1} - S_{i-2} + 16, \end{cases} \quad \begin{cases} r_0 = L, \\ r_1 = \sqrt{10}L, \\ r_i = \left(\sqrt{(2i+1)^2 + 1}\right)L. \end{cases} \tag{7}$$

Thus, the probability of these two indirect key establishments is

$$\begin{cases} P_v^i = \dfrac{n_v^s}{n_v} = \dfrac{T_i \rho \dfrac{\sqrt{3}}{4} d^2}{\pi r_i^2 \rho} = \dfrac{\sqrt{3} T_i}{4\pi (i+1)^2}, \\ P_u^i = \dfrac{n_u^s}{n_u} = \dfrac{S_i \rho L^2}{\pi r_i^2 \rho} = \dfrac{S_i}{\pi[(2i+1)^2 + 1]}. \end{cases} \tag{8}$$

In Fig. 3, with an increase of the number of hops, the connectivity probability would be augmented. Additionally, Fig. 3 shows that the probability of key establishment in the triangle-based key predistribution scheme is always much higher than that in the square-based key predistribution scheme regardless of the number of hops. Hence, the performance of network connectivity has been enhanced remarkably.

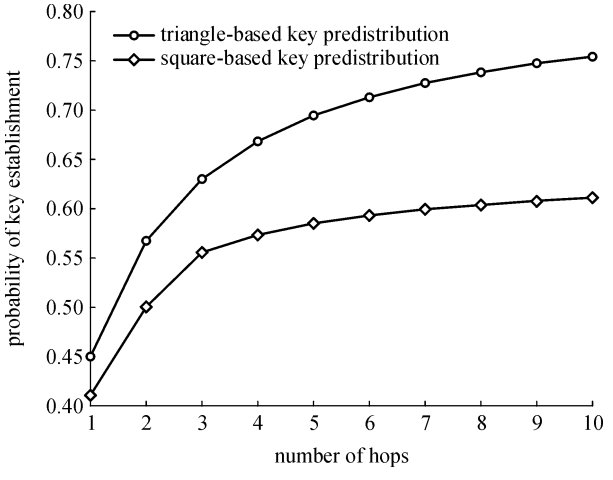

Fig. 3 Relation between probability of key establishment in these two key predistribution schemes and number of hops

4.2 Security against node capture

We assume that an adversary can launch a physical attack on the fraction p_c of sensor nodes in the network, while they are deployed to read secret information from their memories. We need to find how a successful attack on the fraction p_c of sensor nodes affects the rest of the network. In particular, we want to find the probability of compromised pairwise keys between uncompromised nodes.

4.2.1 Fraction of compromised pairwise keys

According to the analysis of the polynomial-based key predistribution scheme, unless more than t shares of a bivariate polynomial are disclosed, an attacker would not know about the non-compromised pairwise keys established through this polynomial. Thus, the security of our scheme depends on the average number of sensor nodes that share the same polynomial. The density of the sensor node deployment can be estimated by $\rho = m/(\pi r^2)$, where m is the average number of sensor nodes in the signal range of each sensor node. The average number of sensor nodes expected to be located in a triangle is $\dfrac{m}{\pi r^2}\dfrac{\sqrt{3}}{4}d^2$. Hence, the average number of sensor nodes that shares at least one common polynomial in the triangle-based key predistribution scheme is

$$N_s = \frac{m}{\pi r^2}\frac{\sqrt{3}}{4} d^2 4 = \frac{\sqrt{3} m d^2}{\pi r^2}. \tag{9}$$

If only $N_s \leqslant t$, the attacker could not know the information about non-compromised pairwise keys.

In the preceding assumption, a fraction p_c of sensor nodes in the network has been compromised. This represents the probability p_c that each sensor node has of being compromised. Thus, among N_s sensor nodes that hold the polynomial shares, the probability that i sensor nodes have been disclosed can be estimated to be

$$P_c(i) = \binom{N_s}{i} p_c^i (1-p_c)^{N_s - i}. \tag{10}$$

Thus, the probability that the bivariate polynomial is compromised is $P_c = 1 - \sum_{i=0}^{t} P_c(i)$. For any pairwise key established directly between non-compromised sensor nodes, the probability that it is compromised is the same as P_c.

4.2.2 Relationship between security against node capture and various parameters

In this subsection, we study how various parameters, such as deployment density m, the size of cell d, and polynomial degree t affect security against node capture. In this experiment, we assume that storage capacity of each node is equivalent to 200 cryptographic keys, and signal propagation distance r is simplified to the basic unit for distance measurement ($r = 1$).

1) Security against node capture (network resilience) versus deployment density m

For the sake of simplicity, we fix $d = 1$ and $t = 40$. Figure 4 indicates that the less the deployment density, the less the fraction of keys are compromised for non-compromised sensor nodes. It clearly shows that network resilience and deployment density are two conflicting properties. Higher deployment density leads to lower network resilience.

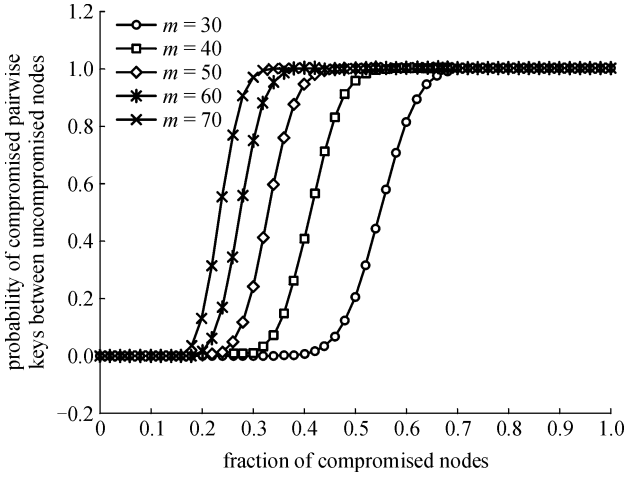

Fig. 4 Network resilience under different m, when $d = 1$, $t = 40$

2) Security against node capture versus the size of cell d

The resilience of our scheme is also affected by cell size. When the size of each triangle is bigger, more nodes would be compromised and the damage to communication should be more severe. To verify this hypothesis, we set $m = 50$ and $t = 40$. Figure 5 depicts the results, which illustrate that the bigger the size of the triangle, the higher the probability of keys being disclosed for uncompromised sensor nodes. From the above analysis, we conclude that security against node capture would be improved by decreasing deployment density and by controlling the size of the triangle deployment model.

3) Security against node capture versus polynomial degree t

To compare with the square-based scheme, we plot the resilience results for a number of different polynomial degrees. In Fig. 6, on the one hand, the higher the degree of polynomial, the lower the probability of keys being disclosed for uncompromised sensor nodes. On the other hand, the triangle-based key predistribution scheme has a higher security threshold than that in the square-based key distribution scheme under the same polynomial degree. Thus, our scheme has stronger security tolerance.

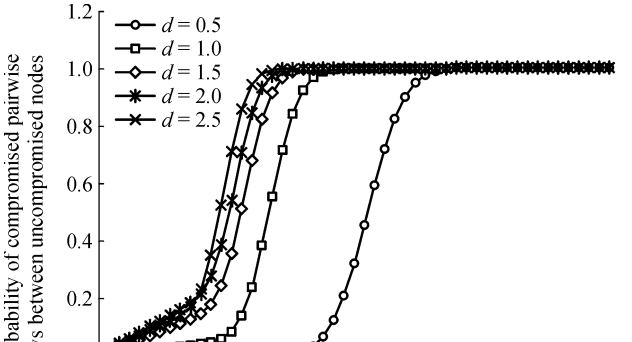

Fig. 5 Network resilience under different d, when $m = 50$, $t = 40$

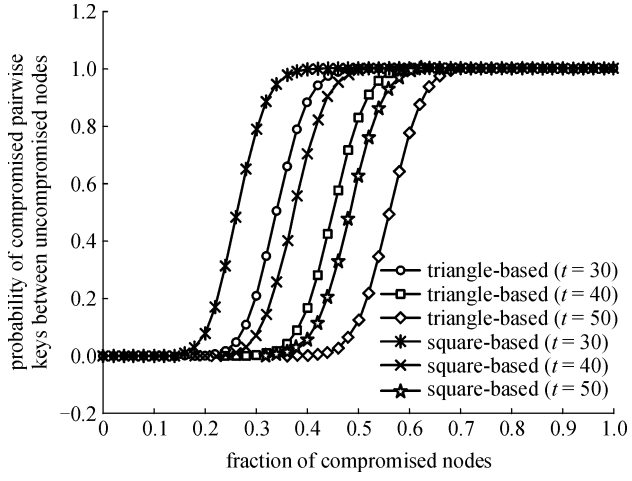

Fig. 6 Security comparison of triangle-based scheme and square-based scheme under different polynomial degree t, when $m = 50$, $d = 1$

5 Conclusions

In this article, we presented a triangle-based key predistribution scheme using sensor node location information and established pairwise keys between nodes by using the bivariate t-degree polynomial in a triangle deployment region. In our discussion, the proposed scheme remarkably

increased the probability of direct key establishment and that of indirect key establishment. We then analyzed security against node capture of key management in our scheme and compared it with that in the square-based key predistribution scheme under the same polynomial degree. In future work, we plan to focus on optimizing the deployment strategy and developing the approach of adjusting the polynomial distribution.

References

1. Akyildiz I F, Su W, Sankarasubramaniam Y, Cayirci E. A survey on sensor networks. IEEE Communications Magazine, 2002, 40(8): 102–114
2. Dai H Y, Xu H B. Overview of security in wireless sensor networks (WSN). Application Research of Computers, 2006, 23(7): 12–17 (in Chinese)
3. Eschenauer L, Gligor V D. A key management scheme for distributed sensor networks. In: Proceedings of the 9th ACM Conference on Computer and Communications Security. New York: ACM, 2002, 41–47
4. Spencer J. The Strange Logic of Random Graphs. Algorithms and Combinatorics 22. Berlin: Springer, 2000
5. DeGroot M H, Schervish M J. Probability and Statistics. 3rd ed. New Jersey: Addison Wesley, 2001
6. Chan H W, Perrig A, Song D. Random key predistribution schemes for sensor networks. In: Proceedings of 2003 IEEE Symposium on Security and Privacy. 2003, 197–213
7. Liu D G, Ning P. Establishing pairwise keys in distributed sensor networks. In: Proceedings of the 10th ACM Conference on Compute and Communications Security. New York: ACM, 2003, 52–61
8. Liu D G, Ning P. Location-based pairwise key establishments for static sensor networks. In: Proceedings of the 1st ACM Workshop on Security of Ad Hoc and Sensor Networks. New York: ACM, 2003, 72–82
9. Blundo C, De Santis A, Herzberg A, Kutten S, Vaccaro U, Yung M. Perfectly secure key distribution for dynamic conferences. In: Proceedings of Cryptology-CRYPTO'92. Berlin: Springer, 1992, 471–486