RPoK: A Strongly Resilient Polynomial-based Random Key Pre-distribution Scheme for Multiphase Wireless Sensor Networks

Hisashige Ito, Atsuko Miyaji and Kazumasa Omote
Japan Advanced Institute of Science and Technology (JAIST)
Ishikawa, JAPAN
{h-ito,miyaji,omote}@jaist.ac.jp

***Abstract*— In this paper[1], we propose a strongly resilient polynomial-based random key pre-distribution scheme for multiphase wireless sensor networks (RPoK): a private sub-key is not directly stored in each sensor node by applying the polynomial-based scheme to the RoK scheme. Such a polynomial is linearly transformed using forward and backward keys in order to achieve the forward and backward security of polynomials. As a result, our scheme achieves a large reduction of the ratio of compromised links by enhancing the security of the previous RoK scheme. The results obtained analytically and by simulations show that our scheme can dramatically improve the ratio of compromised links compared with the RoK scheme.**

I. Introduction

Wireless Sensor Networks (WSNs) consist of small, battery-operated, limited memory and limited computational power devices called sensor nodes. Hence most existing key management schemes are based on symmetric key cryptography. One of the most popular schemes, referred to as *RKP* (Random Key Pre-distribution) in this paper, was proposed by Eschenauer and Gligor [5]. The security of the whole network in RKP degrades with time when assuming the attacker. An attacker who corrupts several nodes can partially reconstruct, from the compromised nodes key rings, the key pool of system. If the attacker is constantly corrupting nodes, it will eventually learn the whole key pool and all newly deployed nodes will establish links that will immediately be compromised. This is a non-desirable property.

The WSNs are usually deployed to operate for a long time period of time. Multiphase WSNs form a network, in which the sensor nodes are periodically redeployed since their batteries are depleted. Resilient multiphase WSNs possess the feature that a network automatically **self-heals** against node-capture attacks. The *resiliency* (self-healing) means that the ratio of compromised links is suppressed even if the adversary regularly corrupts sensor nodes of the network (i.e., *constant attacker model*). Although the existing RKP scheme cannot achieve such self-healing even in multiphase WSNs, the RoK scheme [2] can achieve it. However, the ratio of compromised links is not too low in the existing resilient RKP schemes for multiphase WSNs including the RoK scheme. In other words, the existing schemes for multiphase WSNs do not satisfy the high resiliency of links against node-capture attacks. For instance, the RoK scheme shows that the ratio of compromised links goes down to 10% against the constant attacker.

In this paper, we propose a strongly resilient polynomial-based RKP scheme for multiphase WSNs (RPoK): a private sub-key is not directly stored in each sensor node by applying the polynomial-based [7] scheme to the RoK scheme. Such a polynomial is linearly transformed using forward and backward keys in order to achieve the forward and backward security of polynomials. As a result, our scheme achieves a large reduction of the ratio of compromised links by enhancing the security of the previous RoK scheme. Furthermore, if the attacker operates only during a limited period of time, the network will automatically self-heal more rapidly than in the RoK scheme, when the attacker stops compromising nodes. Through an analysis and simulations, we show that our scheme can significantly decrease the ratio of compromised links as compared with the RoK scheme.

II. Related Work

Most existing pairwise key pre-distribution schemes are based on symmetric key cryptography. One of the most popular schemes, referred to as *RKP* in this paper, was proposed by Eschenauer and Gligor [5]. In this basic probabilistic scheme, each sensor node randomly picks a set of keys from a key pool before the deployment so that any two of the sensor nodes have a certain probability to share at least one common key. Chan, Perrig and Song [3] further extended this idea and presented a q-composite key pre-distribution scheme, in which any two sensors share at least q pre-distributed keys.

Inspired by the basic RKP scheme and the polynomial-based key pre-distribution scheme [1], Liu, Ning and Li [7] proposed a polynomial-based RKP scheme: it is a random subset assignment scheme, in which a polynomial pool is used, instead of using a key pool as in the previous approaches. The random subset assignment scheme assigns to each sensor node the secrets generated from a random subset of polynomials in the polynomial pool.

[1]This study is partly supported by Grant-in-Aid for Scientific Research (A) (21240001) and IT Specialist Program of Ministry of Education, Culture, Sports, Science and Technology, Japan (MEXT).

Castelluccia and Spognardi [2] have proposed the resilient (robust) RKP scheme (RoK) for multiphase WSNs, in which the network resiliency increases without reducing secure connectivity. The RoK scheme improves the security of the RKP scheme by limiting the lifetime of the key pools and by refreshing the pool sub-keys.

Some recent schemes improve the resiliency of the RoK. Yilmaz et al. [8] proposed a more resilient scheme than the RoK to speed up the self-healing process. Kalkan et al. [6] proposed a zone-based RKP (Zo-RoK) scheme which combines the best parts of Du et al.'s scheme [4] and the RoK, and improves the resiliency of the RoK. The ratio of compromised links in these schemes is not too low, although they are more resilient than the RoK.

III. Preliminaries

A. Notation

P	:	Key pool size
m	:	Key ring size of sub-keys
n	:	Total number of nodes (i.e., Size of network)
G	:	Last generation of the network
g_X	:	Deployed generation of node X
c	:	Average number of captured nodes during a generation
ML	:	Maximum life generation of a node
FKP^j	:	Forward key pool at generation j
BKP^j	:	Backward key pool at generation j
PLP^j	:	Polynomial pool at generation j
FKR_X^j	:	Forward key ring of X at generation j
BKR_X^j	:	Backward key ring of X at generation j
PLR_X^j	:	Polynomial ring of X at generation j
fk_s^j	:	s-th forward key $\in FKP^j$ at generation j
bk_s^j	:	s-th backward key $\in BKP^j$ at generation j
ID_X	:	Index of node X
q	:	Large prime number
H	:	Secure hash function $H:\{0,1\}^* \rightarrow \{0,1\}^q$
F	:	Hash function $F:\{0,1\}^* \rightarrow \{0,1\}^{log_2(P)}$
$f_s^j(x,y)$	:	s-th bivariate t-degree polynomial at generation j over a finite field $\mathbb{F}_q$

A generation is a regular time epoch divided into fixed-length time slots.

B. Requirements

The following requirements need to be considered when designing a resilient RKP scheme in WSNs. Although pre-distribution of more keys into sensor nodes increases secure connectivity, more keys can be revealed to the adversary.

High secure connectivity. After the deployment, two nodes share at least one common key with a certain probability to establish a link. This probability is called secure connectivity. High secure connectivity is required in the RKP scheme. The connectivity depends on P and m.

High resiliency. Sensor nodes may be deployed in public or hostile locations in many applications. We assume that the adversary can mount a physical attack on a sensor node after it is deployed, and read secret information from its memory. Resiliency is estimated by the ratio of links that are compromised by the capture of nodes.

Restricted resources. It is required that the WSNs consist of small, battery-operated devices with limited memory and limited computational power.

C. Attacker Model

We assume two different types of attackers (the constant attackers and the temporary attackers) in order to consider the different environments.

Constant attacker model. This type of attackers regularly corrupts nodes of the network without interrupting. In other words, the constant attacker keeps compromising nodes at a constant rate, from the deployment of the first generation of sensors to the end of the life of the network.

Temporary attacker model. This type of attackers is active only during a limited amount of time. The temporary attacker compromises nodes within the specific period.

IV. The RoK Scheme

Castelluccia and Spognardi have proposed a resilient (robust) RKP scheme (RoK) for multiphase WSNs, in which the network resiliency increases without reducing secure connectivity [2]. In this scheme all the keys are identified with the generation, hence all the valid sub-keys are updated by the end of each generation. In other words, sub-keys have limited lifetimes and are refreshed periodically. Furthermore, a security mechanism should be able to guarantee that the key ring of any node is bound to a given amount of time. After exceeding this time, a node should no more establish a secure communication between new deployed nodes. This maximum life generation is set to 10 generations ($ML = 10$), which is almost the maximum battery life of a node. A sensor deployed at generation j will run out of power before generation $j+ML$. This binding is provided by the backward and forward hash chains. As a result of this binding, the keys obtained from captured nodes get old by this time and new established links remain safe.

A. Protocol Description

1. Key pools generation. The forward and the backward key pool are initiated with $P/2$ random keys. In the case of the forward key pool, each key is updated by hashing the current key with H at each generation. More precisely, the forward key pool at generation j is defined as: $FKP^j = \left\{ fk_1^j, fk_2^j, \ldots, fk_{P/2}^j \right\}$, where $fk_s^j = H(fk_s^{j-1})$ ($j = 1,\ldots,G$, $s = 1,\ldots,P/2$). On the other hand, the backward key pool is first generated for generation G. The backward key pool at generation j is defined as: $BKP^j = \left\{ bk_1^j, bk_2^j, \ldots, bk_{P/2}^j \right\}$, where $bk_s^j = H(fk_s^{j+1})$ ($j = G-1,\ldots,0$, $s = 1,\ldots,P/2$).

2. Key rings assignment. Each node is configured with $m/2$ sub-keys from the backward and forward key pools. More formally, node A is configured with key rings, defined as:

$FKR_A^j = \left\{ fk_s^j \right\}$ and $BKR_A^j = \left\{ bk_s^j \right\}$, such that $s = F(ID_A \parallel i \parallel g_A)$ ($i = 1,2,\ldots,m/2$). Note that $g_A = j$ when a node A is deployed at generation j.

3. Establishing a secure link. After deployment, a node A initiates the neighbors discovery procedure by broadcasting a message that includes ID_A and g_A. A receiver node B, at first, decides if their generations are close enough. This is done by testing if $|g_A - g_B| < ML$. In addition to this, if $g_A < g_B$ and the above holds, then they can share keys starting from generation g_B up to generation "$g_A + ML - 1$". Secondly, the node B calculates $F(ID_A \parallel i_1 \parallel g_A)$ and compares them with its indices, $F(ID_B \parallel i_2 \parallel g_B)$ for all $i_1, i_2 \in 1,2,\ldots,m/2$. If there are collisions such that $F(ID_B \parallel y \parallel g_B) = F(ID_A \parallel x \parallel g_A)$, where $x,y \in \{1,2,\ldots m/2\}$, then it is known that they both have the forward key $fk^{g_B}_{F(ID_B \parallel y \parallel g_B)}$ and the backward key $bk^{g_A+ML-1}_{F(ID_B \parallel y \parallel g_B)}$ in their memory. In this way, all colluding local indices $a,b,\ldots,z \in \{1,2,\ldots m/2\}$ are found and the following becomes their pairwise symmetric key:

$$K_{AB}^{RoK} = H\left(fk^{g_B}_{F(ID_B \parallel a \parallel g_B)} \parallel bk^{g_A+ML-1}_{F(ID_B \parallel a \parallel g_B)} \parallel \cdots \parallel fk^{g_B}_{F(ID_B \parallel z \parallel g_B)} \parallel bk^{g_A+ML-1}_{F(ID_B \parallel z \parallel g_B)} \right)$$

Note that K_{AB}^{RoK} satisfies both *forward security* and *backward security*, i.e., sub-keys composing K_{AB}^{RoK} is useful only between two generations of g_B and $g_A + ML - 1$ for attackers.

B. Analytical Model

As explained in [2], the average probability P_{RoK} that a link is indirectly compromised at generation j against constant attackers is given by:

$$P_{RoK} = \sum_{i=1}^{m} \left(1 - \left(1 - \frac{m}{P} \right)^{c \cdot E_c} \right)^i \frac{p_i}{1 - p_0} \tag{1}$$

where p_i is the probability that two nodes share i sub-keys, given by $p_i = \left({}_P C_i \cdot {}_{P-i} C_{2(m-i)} \cdot {}_{2(m-i)} C_{m-i} \right) / {}_P C_m{}^2$, and E_c is the average span over which a link can be captured, given by:

$$E_c = \sum_{j=0}^{ML} j \left[p(j) \cdot \sum_{k=0}^{j} p(k) + \sum_{k=0}^{j-1} p(k) \cdot p(j) \right] \tag{2}$$

where $p(j)$ is the probability that a node picked at random from the network is in age j which is described in [2].

V. The Proposed Scheme (RPoK)

The primary aim of our scheme is to not only increase secure connectivity between nodes, but also decrease the compromised ratio of nodes against node-capture attacks in multiphase WSNs. Practically, a private sub-key is not directly stored in each sensor node by applying the t-degree polynomial-based scheme to the RoK scheme. As a result, an attacker has to capture $(t+1)$ sub-keys during a limited period of time in order to corrupt a link. Furthermore, we achieve the forward and backward security of the polynomial by linear transformations using forward and backward keys. Therefore, our scheme can dramatically improve the ratio of compromised links compared with the RoK scheme.

A. Protocol Description

In this section, we mainly focus on the parts that are different from the RoK scheme.

1. Pools generation. Our scheme uses three kinds of pools, i.e., FKP^j, BKP^j and PLP^j, where FKP^j and BKP^j are the same as RoK. PLP^j is defined as $PLP^j = \left\{ f_1^j(x,y), f_2^j(x,y), \ldots, f_{P/2}^j(x,y) \right\}$, where $f_s^j(x,y) = \alpha_{j-1} f_s^{j-1}(x,y) + \beta_{j-1}$, $\alpha_{j-1} = H(fk_s^{j-1} \parallel bk_s^{j-1})$ and $\beta_{j-1} = H(bk_s^{j-1} \parallel fk_s^{j-1})$ ($j = 1,\ldots,N$, $s = 1,\ldots,P/2$).

2. Rings assignment. Node A is configured with key rings, defined as: $FKR_A^j = \left\{ fk_s^j \right\}$, $BKR_A^j = \left\{ bk_s^j \right\}$ and $PLR_A^j = \left\{ f_s^j(x,y) \right\}$, such that $s = F(ID_A \parallel i \parallel g_A)$ ($i = 1,2,\ldots,m/2$). Note that $g_A = j$ when the node A is deployed at generation j.

3. Establishing a secure link. After deployment, a node A initiates neighbors discovery procedure with node B and both nodes calculate indices, similar to RoK. If there are collisions such that $F(ID_B \parallel y \parallel g_B) = F(ID_A \parallel x \parallel g_A)$, where $x,y \in \{1,2,\ldots m/2\}$, then it is known that they both have $fk^{g_B}_{F(ID_B \parallel y \parallel g_B)}$, $bk^{g_A+ML-1}_{F(ID_B \parallel y \parallel g_B)}$ and $f^{g_B}_{F(ID_B \parallel y \parallel g_B)}(ID_A, ID_B)$ in their memory. In this way, all colluding local indices $a,b,\ldots,z \in \{1,2,\ldots m/2\}$ are found and the following becomes their pairwise symmetric key:

$$K_{AB}^{RPoK} = H\left(f^{g_B}_{F(ID_B \parallel a \parallel g_B)}(ID_A, ID_B) \parallel \cdots \parallel f^{g_B}_{F(ID_B \parallel z \parallel g_B)}(ID_A, ID_B) \right)$$

Note that $f_s^{g_B}(ID_A, ID_B)$ satisfies both forward and backward security because of linear transformations, as mentioned in the pools generation phase. Furthermore, in our scheme, K_{AB}^{RPoK} is a session key in each time-slot, while K_{AB}^{RoK} in RoK is a common key in the overlapping generations. We assume that K_{AB}^{RPoK} is updated in each time slot.

B. Security Evaluation

The goal of this section is to evaluate the resiliency of our proposal, and compare it with the resiliency of the RoK scheme. For a fair comparison, the same size of memory is assumed among three schemes: RKP, RoK and RPoK. When the length of each ring is $m/2$, the number of sub-keys of RoK is just m, while the number of sub-keys and coefficients in our scheme is in total $m(t+3)/2$. Thus, we set $2m/(t+3)$ as the length of each ring in our scheme from the standpoint of fairness.

1) Evaluation by Simulation: We followed a similar simulation procedure as in [2], hence we evaluate the ratio of compromised links against constant attackers to show the improvement of resiliency in our scheme, and we also evaluate the ratio of compromised links against temporary attackers to show the faster self-healing capabilities of our scheme. For ease of exposition and without loss of generality, we assume that the time slots of node compromising have the same duration and are synchronized similar to RoK. The R_S is defined as (active-compromised links) / (active links).

Simulation Setup: The simulations were implemented in C on Windows XP SP3 . All the simulations were repeated 25 times, and the results report the average values.

- *Parameters*: To simplify the security analysis, we modeled the network as a grid of sensors of size $n = 400$. The maximum life generation of a node is set to 10 ($ML = 10$). Note that P and m in our scheme are decided not only by the degree t but also by the secure connectivity. For instance, we set $(P,m) = (1660,100)$ for $t = 2$ and $(P,m) = (1158,83)$ for $t = 3$. A generation consists of 10 time slots. The attacker corrupts one active node at each time slot ($c = 10$).
- *Network*: We assume that the number of neighbors of each sensor is constant and equal to four. We also assume that the network topology does not change over time: At each generation, expired nodes are replaced with new ones, configured with fresh keys. The new nodes establish secure links with their four neighbors, using session keys.

Simulation Details: We evaluate the security of these three schemes by the number of links that get **indirectly** corrupted when the nodes are compromised. A link, between nodes A and B, is said to be indirectly corrupted when neither A nor B have been corrupted, but when the adversary has collected all the backward and forward sub-keys that A and B have in common. These sub-keys have been collected by compromising other nodes.

At generation 0, n nodes are deployed. We simulated nodes expiration by assigning to each node a random expiration date, chosen according to a Gaussian distribution with mean $ML/2$ and with standard deviation $ML/6$. In other words, sub-keys have limited lifetimes (i.e., the mean life generation is 5) and are refreshed periodically.

The attacker may create a table of keys that belong to various generations. He corrupts one active node at each time slot and updates such a table. He then uses this table to corrupt links. We counted, at each generation, the number of compromised links and computed the ratio R_S. Note that an attacker does not capture a node which has already been corrupted in this simulation.

Simulation Results: Figure 1 displays the ratio R_S against a constant attacker. It can be observed that the R_S of RKP reaches 1 in a really short time. This means RKP is not a resilient scheme against node-capture attacks in multiphase WSNs. On the other hand, the R_S of RPoK is suppressed to 0.0081 in $t = 2$ while the R_S of RoK is suppressed to 0.047. Figure 3 extends the y-axis of Figure 1. Of course, R_S of RPoK comes close to zero by the more degree t.

The results for the temporary attacker are collected in Figure 2. The action interval of the attacker (from generation 5 to generation 14) is denoted with the label "Adv. activity". We simulated a network with the same settings as the network used for the constant attacker. The RKP scheme keeps a ratio of compromised links greater than 0, even when the adversary stops its activity. Figure 2 illustrates the self-healing property of RPoK and RoK: as soon as the adversary stops its activity, the ratio of the compromised links starts decreasing

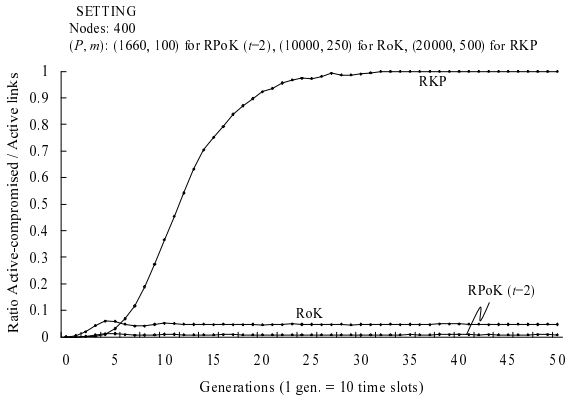

Fig. 1. R_S: Simulation results against constant attackers.

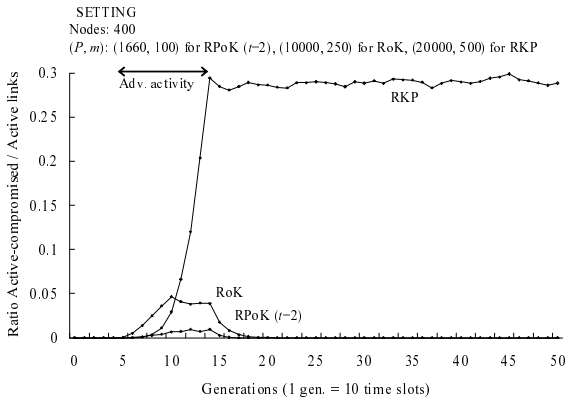

Fig. 2. R_S: Simulation results against temporary attackers.

as new generations of nodes are deployed. Consequently, these simulation results show that our scheme outperforms the RoK scheme in resilient multiphase WSNs. Of course, R_S of RPoK comes close to zero by the more degree t.

2) Analytical Model: We focus on the analytical model against constant attackers (the more powerful attackers) in order to compare with the simulation results, similar to the RoK. For simplicity, we assume that the attacker corrupts all the nodes at once, i.e., at the beginning of each generation. First of all, we revise Equation (2) of the RoK, since the approximation of this expression is slightly loose. Although $p(j-1)$ is necessary at the point of $p(j)$ in Equation (2), the expression $\sum_{k=0}^{j} p(k)$ is the probability of $p(0)$ or $p(1)$ or $\cdots$ or $p(j-1)$ or $p(j)$. This means that $p(j-1)$ is not alway included in $\sum_{k=0}^{j} p(k)$. Thus, the revised E_c' is given by:

$$E_c' = \sum_{j=0}^{ML} j \left(p(j) \cdot \sum_{k=0}^{j} p(k) + \sum_{k=0}^{j-1} p(k) \cdot p(j) - p(j) \left(\sum_{k=0}^{Max(0,\ j-2)} p(k) \right)^{j-1} \right) \quad (3)$$

Secondly, we can obtain the following analytical model by extending the polynomial-based scheme [7] using Equation (3). The probability that an active link is computed at generation

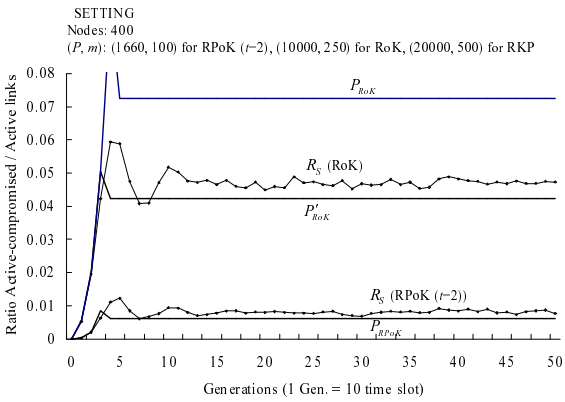

Fig. 3. Comparison between the analytical results and simulation results against constant attackers.

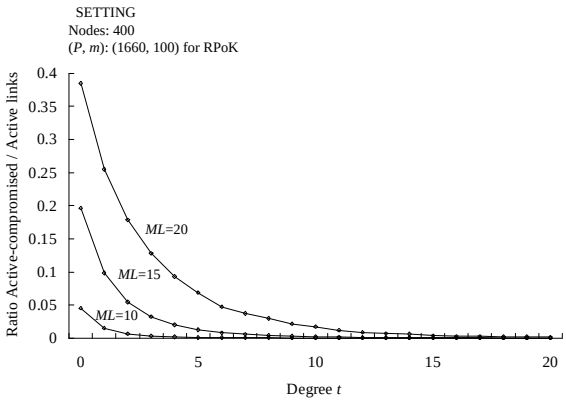

Fig. 4. Analysis of R_{RPoK} against constant attackers.

j in RPoK is defined as:

$$P_{RPoK} = 1 - \sum_{i=0}^{t} c \cdot E_c' C_i \left(\frac{m}{P}\right)^i \left(1 - \frac{m}{P}\right)^{c \cdot E_c' - i} \quad (4)$$

Although this probability is constant similar to the RoK since it does not depend on j, it follows the results of RKP in the early generations in Figure 3 because the node is hardly redeployed. Using Equation (1) and (4), we obtained $R_{RoK} = 0.0725$ and $R_{RPoK} = 0.00612$ ($t = 2$).

VI. Discussion

A. Comparison

Figure 3 shows a comparison of between the analytical results (P_{RPoK} and P_{RoK}) with simulation results (R_S). Note that the simulation results are the same as in Figure 1. The P'_{RoK} is new results using E'_c in Equation 3, hence we can obtain the $P'_{RoK} = 0.0422$. We found that the P'_{RoK} matched the simulation results better than the P_{RoK}. We also found that our results of P_{RPoK} well matched the simulation results.

B. Analysis of R_{RPoK}

The more the degree t increases, the more resilient the RPoK becomes, but, on the other hand, the higher the computational costs become. In this section, we consider how to decide the polynomial degree t in our scheme according to ML, assuming constant attacker model. In Figure 4, we evaluated the transition of P_{RPoK} by changing t against constant attackers when j grew sufficiently. If the security goal is $P_{RPoK} < 0.0001$, then we have to set $t = 10$ for $ML = 10$, $t = 20$ for $ML = 15$, and $t = 30$ for $ML = 20$. These results show that the ratio of compromised links can come close to zero. If ML is set to a higher value, then the refreshing period of the sub-keys becomes long on average, that is, the R_{RPoK} becomes high.

C. Computational and Communication Costs

The computational cost of the RPoK is a little larger than the one of the RoK. As for the computational cost of link establishment, that for RPoK is $H + \frac{m^2}{4}F + tM$, while that for RoK is just H, where M is the multiple operation over a finite field $\mathbb{F}_q$. Note that the computational cost of F is much lower than H. As for the computational cost of key update, that for RPoK is $2mH + \frac{m(t+1)}{2}M$, while that for RoK is mH. On the other hand, the communication costs of our scheme is the same as that of the RoK scheme, since the communication in both schemes is required in only the neighbor discovery procedure of establishing a secure link.

VII. Conclusion

We proposed a strongly resilient polynomial-based RKP scheme for multiphase WSNs (RPoK), in which the ratio of compromised links approaches to zero (e.g., such a ratio can be analytically suppressed to less than 0.01% by setting a certain polynomial degree, as described in Figure 4.). Our simulation shows that a RPoK-based network that is constantly attacked is much less affected than a RoK-based network. Our simulation also shows that a network that is temporarily attacked automatically self-heals faster than the RoK scheme.

References

[1] C. Blundo, A.D. Santis, A. Herzberg, S. Kutten, U. Vaccaro, and M. Yung. Perfectly-secure key distribution for dynamic conferences. In *CRYPTO'92*, LNCS, 740, pages 471–486, 1993.

[2] C. Castelluccia and A. Spognardi. Rok: A robust key pre-distribution protocol for multi-phase wireless sensor networks. In *SecureComm2007*, pages 351–360, September 2007.

[3] H. Chan, A. Perrig, and D. Song. Random key predistribution schemes for sensor networks. In *SP'03*, pages 197–213, May 2003.

[4] W. Du, J. Deng, Y.S. Han, S. Chen, and P.K. Varshney. A key management scheme for wireless sensor networks using deployment knowledge. In *INFOCOM'04*, pages 586–597, March 2004.

[5] L. Eschenauer and V.D. Gligor. A key-management scheme for distributed sensor networks. In *CCS'02*, pages 41–47, November 2002.

[6] K. Kalkan, S. Yilmaz, O.Z. Yilmaz, and A. Levi. A highly resilient and zone-based key predistribution protocol for multiphase wireless sensor networks. In *Q2SWinet'09*, pages 29–36, October 2009.

[7] D. Liu, P. Ning, and R. Li. Establishing pairwise keys in distributed sensor networks. *ACM Trans. Inf. Syst. Secur.*, 8(1):41–77, 2005.

[8] O.Z. Yilmaz, A. Levi, and E. Savas. Multiphase deployment models for fast self healing in wireless sensor networks. In *SECRYPT*, pages 136–144, July 2008.