A Canonical Seed Assignment Model for Key Predistribution in Wireless Sensor Networks

PATRICK TAGUE and RADHA POOVENDRAN
Network Security Lab (NSL), University of Washington

A promising solution for trust establishment in wireless sensor networks is the assignment of cryptographic seeds (keys, secrets, etc.) to sensor nodes prior to network deployment, known as *key predistribution*. In this article, we propose a canonical seed assignment model for key predistribution characterizing seed assignment in terms of the probability distribution describing the number of nodes receiving each seed and the algorithm for seed assignment. In addition, we present a sampling framework for seed assignment algorithms in the canonical model. We propose a probabilistic k-connectivity model for randomly deployed secure networks using spatial statistics and geometric random graph theory. We analyze key predistribution schemes in the canonical model in terms of network connectivity and resilience to node capture. The analytical results can be used to determine the average or worst-case connectivity or resilience to node capture for a key predistribution scheme. Furthermore, we demonstrate the design of new key predistribution schemes and the inclusion of existing schemes in the canonical model. Finally, we present a general approach to analyze the addition of nodes to an existing secure network and derive results for a well-known scheme.

Categories and Subject Descriptors: C.2.0 [**Computer-Communication Networks**]: General—*Security and protection*; C.2.1 [**Computer-Communication Networks**]: Network Architecture and Design—*Distributed networks, wireless communication*

General Terms: Algorithms, Design, Performance, Security

Additional Key Words and Phrases: Key establishment, key predistribution, network models, sensor networks

ACM Reference Format:

Tague, P. and Poovendran, R. 2007. A canonical seed assignment model for key predistribution in wireless sensor networks. ACM Trans. Sens. Netw. 3, 4, Article 19 (October 2007), 39 pages. DOI = 10.1145/1281492.1281494 http://doi.acm.org/10.1145/1281492.1281494

This work is supported in part by the following grants: ONR award, N00014-04-1-0479; ARO grant, W911NF-05-1-0491, and DOD IASP award.

A preliminary version of this article appeared in IEEE WiOpt Symposium, April 2006.

Authors' address: University of Washington, Department of Electrical Engineering, PAC AE100R, Campus Box 352500, Seattle, WA 98195-2500; email: {tague,rp3}@u.washington.edu.

1. INTRODUCTION

Advances in sensor technology suggest that large-scale wireless sensor networks (WSNs) can provide sensing and distributed processing using low-cost, resource-constrained sensor nodes [Asada et al. 1998] for commercial, industrial, and military applications such as disaster relief and recovery, medical patient monitoring, smart homes, mechanical system monitoring, and target detection and tracking. As data integrity, authentication, privacy, and confidentiality are often important concerns in such applications, secure communication protocols are required. However, the ad-hoc nature of WSNs require minimal interaction with base stations or a central authority, so trust establishment for secure communication is a critical task [Anderson 2001; Eschenauer and Gligor 2002]. Furthermore, random sensor deployment and the physical communication constraints of sensor nodes make trust establishment a very challenging problem in WSNs.

The resource constraints of sensor nodes are the limiting factor in the type of cryptographic primitives that can be implemented. There have been recent efforts to implement public-key cryptography in wireless sensor networks [Hill et al. 2000; Gura et al. 2004; Gupta et al. 2005; Gaubatz et al. 2005; Du et al. 2005]. However, such protocols cannot yet be implemented on all sensor nodes. Hence, many of the current solutions to key establishment rely on the use of symmetric key cryptography.

A promising solution for the establishment of secure communication in WSNs using symmetric keys is the use of *key predistribution* [Eschenauer and Gligor 2002; Chan and Perrig 2005; Liu et al. 2005]. A key predistribution scheme can be described in two primary phases: *seed assignment* and *link-key establishment*. In the seed assignment phase, *cryptographic seeds* (e.g., hashed master keys [Leighton and Micali 1993], cryptographic keys [Eschenauer and Gligor 2002], or polynomial shares [Liu and Ning 2003]) are assigned to sensor nodes prior to network deployment. In the link-key establishment phase, executed after network deployment, neighboring nodes compute *link-keys* as a function of assigned seeds in order to establish secure one-hop links.

We discuss some of the prominent key predistribution schemes in Section 2. While many of these schemes provide novel approaches for the link-key establishment phase of key predistribution, the scope of seed assignment techniques is limited.

In this article, we make the following contributions:

—We provide a canonical model of seed assignment for key predistribution in WSNs. In the canonical model, seed assignment schemes are characterized in terms of the discrete probability distribution of the number of nodes sharing each seed and the seed assignment algorithm. The canonical model allows the scheme designer to explicitly control the probability distribution and limit the effects of the tails of the distribution.

—We present a sampling framework for randomized seed assignment algorithms for use in the canonical seed assignment model. Seed assignment algorithms are classified according to the selection method used to realize

a given probability distribution, and a representative algorithm from each class is illustrated.

—We develop a model for probabilistic network k-connectivity for randomly deployed secure WSNs in which communication is restricted by radio range and the existence of shared seeds. The result is based on spatial statistics [Cressie 1993] and the asymptotic properties of geometric random graphs [Penrose 1999; Bettstetter 2002].

—We demonstrate how the worst-case analysis of any key predistribution scheme can be performed using the canonical model. Such analysis has not been available in the related literature. We also show that the average case analysis can be performed as in existing works.

—We use the canonical model to characterize the effect of network extension via node addition.

—Apart from describing and analyzing existing work using our canonical seed assignment model, we illustrate the design of new key predistribution schemes with desirable tail effects.

The remainder of this article is organized as follows. We briefly discuss related work in key predistribution in Section 2. Section 3 provides motivation for our work and formally states the problem addressed. In Section 4, we state our adversarial and network models and propose a model for probabilistic k-connectivity of the secure WSN. We propose a canonical model of seed assignment for key predistribution in Section 5. In Section 6, we present a general analysis for seed assignment in the canonical model in terms of the probability of sharing seeds, probabilistic network connectivity, and resilience to node capture. Section 8 illustrates the use of the canonical model in deriving existing key predistribution schemes. In Section 9, we analyze the effect of adding nodes to an existing secure WSN. Finally, we summarize our results in Section 10.

2. RELATED WORK

The problem of trust establishment in resource-constrained WSNs is an active research problem. Stajano and Anderson [1999] provide an overview of the issues involved in establishing trust in WSNs and propose a solution for mobile sensors based on establishing physical contact with a trusted device. The physical constraints of existing sensors were extensively evaluated by Carman et al. [2000]. Based on the challenges set forth by the decentralized and physically constrained nature of WSNs, a significant number of key predistribution schemes have been proposed.

Blom [1984] proposed the use of threshold secret-sharing for key predistribution. In the proposed scheme, each of the N nodes receives a row of the matrix $(DG)^T$ where D is a $K \times K$ random symmetric matrix and G is the $K \times N$ public generator matrix of a maximum-distance separable error-correcting code over a finite field $\mathbb{F}_q$. Any pair of nodes can compute a unique pairwise key as a function of the corresponding rows of $(DG)^T$ and G. The pairwise keys computed in the scheme remain secure until either K nodes collude or an adversary recovers K rows of $(DG)^T$ by capturing nodes. This approach was generalized by Blundo

et al. [1992] as a group key establishment scheme using secret symmetric polynomials over $\mathbb{F}_q$.

Mitchell and Piper [1988] proposed the use of combinatorial block designs for key predistribution, mapping a set of seeds assigned to a node to a block in the design.

Gong and Wheeler [1990] proposed a key predistribution scheme in which $N = m^2$ keys are arranged on the grid points of an m-by-m grid. Each of the N nodes is mapped to a grid point and assigned the $(2m - 1)$ keys in the row and column corresponding to the grid point.

Leighton and Micali [1993] proposed the use of a one-way hash function h to establish keys as a function of global secrets $X_1, \ldots, X_K$. Each node receives a random ID α_i, $0 \leq \alpha_i < L$, and a seed $h^{\alpha_i}(X_i)$ for $i = 1, \ldots, K$, where $h^n(x) = h(h^{n-1}(x))$ for $n > 1$ and $h^1(x) = h(x)$. A pair of nodes that publicly exchange identifiers α_i and β_i for $i = 1, \ldots, K$ can compute a pairwise key as a function of the mutually computable values $h^{max(\alpha_i, \beta_i)}(X_i)$ for $i = 1, \ldots, K$.

Dyer et al. [1995] proposed randomized and derandomized methods of subset assignment for key predistribution that guarantee the existence of schemes with certain collusion resistance properties using the probabilistic method, similar to the techniques of Erdös et al. [1982, 1985] and Alon [1991].

In their seminal work, Eschenauer and Gligor [2002] proposed the use of random key predistribution for securing WSNs. In the proposed scheme, each node is assigned a randomly selected subset of K seeds from a set of $P \gg K$ seeds. Any pair of nodes sharing at least one of the assigned seeds can establish a secure link. Ramkumar et al. [2003], Di Pietro et al. [2003], and Zhu et al. [2003] independently proposed similar methods using pseudo-random identity-based functions for key predistribution. Chan et al. [2003] proposed the q-composite scheme as a modification of random key predistribution [Eschenauer and Gligor 2002]. In the q-composite scheme, a pair of nodes are required to share at least $q \geq 1$ seeds, and all shared seeds are used in link-key establishment. In addition to the q-composite scheme, Chan et al. [2003] proposed the random pairwise scheme in which unique pairwise keys are assigned to a fraction of the node pairs in the network.

Du et al. [2003] and Liu and Ning [2003] simultaneously proposed the combination of random key predistribution [Eschenauer and Gligor 2002] with the threshold secret-sharing schemes of Blom [1984] and Blundo et al. [1992], respectively. In the scheme of Du et al. [2003], each node is assigned a share from each of a random selection of K of the P matrices $(D_i G)^T$, $i = 1, \ldots, P$ where D_i and G are similar to those of Blom [1984]. In the scheme of Liu and Ning [2003], each node is assigned a share from each of a random selection of K of the P secret symmetric polynomials similar to those of Blundo et al. [1992].

Çamtepe and Yener [2004] proposed the use of combinatorial block designs for key predistribution. In this scheme, each node is assigned a block of K seeds given by the entries of a block in a combinatorial design. Furthermore, to allow for flexibility in parameters, Çamtepe and Yener [2004] proposed a hybrid combinatorial scheme. In the proposed scheme, each node is assigned $m < K$ seeds corresponding to one of the $b < N$ blocks of a combinatorial design and a randomly selected subset of $(K - m)$ entries of one of the b blocks of the

complementary combinatorial design. Lee and Stinson [2004] further discuss the use of combinatorial block designs and propose an identity-based one-way function scheme in which seeds are assigned according to a strongly regular graph such that each graph vertex represents L sensor nodes. Each of the L nodes u_i, $i = 1, \ldots, L$, represented by a graph vertex u is assigned a common key k_u. In addition, each u_i is assigned the hashed values $h(k_v \| ID(u_i))$ for each graph vertex v joined to u by an edge.

Ramkumar and Memon [2004] proposed a combination of their identity-based scheme [Ramkumar et al. 2003] with the one-way hash function scheme of Leighton and Micali [1993]. For each node j, K global secrets $X_{j_1}, \ldots, X_{j_K}$ are selected from a pool of $P \gg K$ secrets. The node is assigned the hashed values $h^{\alpha_i}(X_{j_i})$ for $i = 1, \ldots, K$ as proposed by Leighton and Micali [1993].

Chan and Perrig [2005] proposed the PIKE protocol in which the nodes are arranged in a K-dimensional hyper-grid, extending the scheme of Gong and Wheeler [1990]. A common seed is assigned to the set of sensors sharing each coordinate in the hyper-grid. Liu et al. [2005] proposed a hyper-grid seed assignment scheme, effectively combining the methods of Chan and Perrig [2005] and Blundo et al. [1992], such that nodes which share a grid coordinate receive shares of a common secret polynomial.

3. MOTIVATION AND PROBLEM STATEMENT

Various properties of a key predistribution scheme can be analyzed in terms of the number of nodes sharing each seed as a result of seed assignment. Hence, the behavior of a key predistribution scheme is analyzed with respect to the probability that a given seed is shared by exactly λ of the N nodes in the WSN.

3.1 Motivation

The impact of the number of nodes λ sharing a given seed is investigated for the following metrics: the probability that a pair of nodes share at least one seed, the probability that no pair of nodes sharing a given seed are within radio range, and the potential number of secure links established using a given seed.

Intuitively, if the number of nodes λ that share a given seed is small, the probability that one of the λ nodes will share the seed with a neighboring node will be very small. This statement can be justified by estimating the probability that a neighboring node shares the given seed. Since exactly λ of the N nodes in the network hold the given seed, the probability that a neighboring node shares the seed is approximately $\frac{\lambda}{N}$. Given a node with K seeds shared by $\lambda_1, \ldots, \lambda_K$ nodes, the probability that a neighboring node shares at least one key can thus be estimated as

$$\Pr[\text{at least one seed shared}] = 1 - \left(1 - \frac{\lambda_1}{N}\right) \times \cdots \times \left(1 - \frac{\lambda_K}{N}\right). \quad (1)$$

Furthermore, if λ is small and the area within the radio range of a node is significantly less than the deployment area of the network, the probability that a seed shared by λ nodes will not be used to establish a secure link, referred to as the *key wastage* probability, will be large. This statement can be similarly

justified by estimating the key wastage probability as follows. Assuming the sensor nodes are randomly distributed over a region $\mathcal{A}$, the probability that a given pair of nodes are not within a distance r is given by

$$n_r = 1 - \frac{\pi r^2}{|\mathcal{A}|}. \tag{2}$$

The key wastage probability $w(\lambda)$ can be estimated as

$$w(\lambda) \approx n_r^{\binom{\lambda}{2}} = \left(1 - \frac{\pi r^2}{|\mathcal{A}|}\right)^{\binom{\lambda}{2}}, \tag{3}$$

noting that equality does not hold because the $\binom{\lambda}{2}$ events are not independent. Hence, the key wastage probability decreases exponentially in λ, and a seed shared by a small number of nodes λ will be unused with high probability.

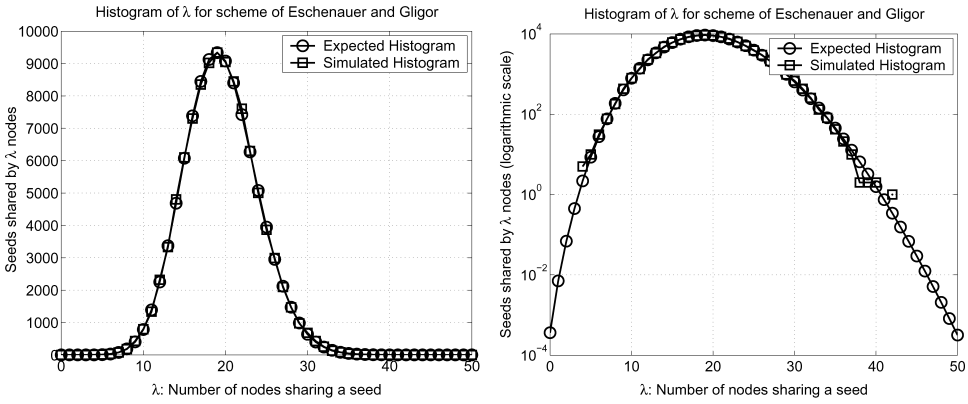
If the number of nodes λ which share a given seed is large, the number of secure links established using the seed is potentially large. An adversary with the seed can thus compromise a large number of secure links. This statement can be similarly justified by estimating the number of secure links that can be established using the given seed. Given λ nodes that share the seed, there can be as many as $\binom{\lambda}{2}$ secure links formed using the given seed, increasing quadratically in λ.

Quantifying the above metrics as a function of λ also allows for the worst-case analysis with respect to each metric. Let $\mathcal{P}(\lambda)$ denote the probability that a given seed is shared by λ nodes and $\mathcal{H}(\lambda) = P\mathcal{P}(\lambda)$ denote the expected number of seeds shared by exactly λ nodes, where P is the total number of seeds. $\mathcal{P}$ and $\mathcal{H}$ thus denote the *probability distribution* and *expected histogram* of λ, respectively. The *expected* worst case for each metric can thus be quantified as a function of the expected histogram $\mathcal{H}$.

The expected worst-case probability of sharing seeds and key wastage probability can be computed as a function of λ_{min}, defined as the minimum λ such that $\mathcal{H}(\lambda) \geq 1$. The expected worst-case number of compromised links can similarly be computed as a function of λ_{max}, defined as the maximum λ such that $\mathcal{H}(\lambda) \geq 1$. The deviation of each metric due to variation in λ can thus be quantified by comparing the values at λ_{min} and λ_{max} to that at the average value μ of the distribution $\mathcal{P}$.

As an example, the above metrics are evaluated for the random key predistribution scheme of Eschenauer and Gligor [2002]. In this scheme, each node is assigned a random subset of K seeds from a pool of $P \gg K$ seeds. When a subset of K seeds is selected for one node, a particular seed is selected with probability $\frac{K}{P}$, which can be modeled as a Bernoulli random variable. Hence, the probability distribution $\mathcal{P}(\lambda)$ is the binomial distribution $\mathcal{B}(N, \frac{K}{P})$ such that $\mathcal{P}(\lambda)$ is given by

$$\mathcal{P}(\lambda) = \binom{N}{\lambda}\left(\frac{K}{P}\right)^{\lambda}\left(1 - \frac{K}{P}\right)^{N-\lambda} \tag{4}$$



(a) (b)

Fig. 1. The expected histogram $\mathcal{H}(\lambda)$, representing the number of seeds shared by exactly λ nodes, is illustrated for Example 3.1 with vertical axis in (a) linear scale and (b) logarithmic scale.

with average value $\mu = \frac{NK}{P}$, and the values of the histogram $\mathcal{H}$ are given by

$$\mathcal{H}(\lambda) = P\binom{N}{\lambda}\left(\frac{K}{P}\right)^{\lambda}\left(1-\frac{K}{P}\right)^{N-\lambda}. \tag{5}$$

The following example illustrates the effect of this binomial distribution on the metrics of interest.

Example 3.1. Let a WSN of $N = 10,000$ nodes be assigned seeds according to the key predistribution scheme of Eschenauer and Gligor [2002] with $K = 200$ and $P = 102,881$, where P is chosen to guarantee network connectivity with probability 0.999 for an average of $d = 50$ nodes within radio range. The average number of nodes sharing a given seed is $\mu = \frac{NK}{P} = \frac{10,000\times 200}{102,881} \approx 20$. The expected histogram $\mathcal{H}$ and the simulated histogram are provided in Figure 1. For the given parameters, the condition $\mathcal{H}(\lambda) \geq 1$ is satisfied for all λ between $\lambda_{min} = 4$ and $\lambda_{max} = 40$.

The variation in the probability of sharing seeds is quantified by computing the probability given in (1) for $\lambda_1, \ldots, \lambda_K$ all equal to the values λ_{min}, μ, and λ_{max}, yielding 0.0769, 0.3224, and 0.5514, respectively. The expected worst-case probability of sharing seeds can alternatively be defined as a function of the K smallest values $\lambda_{min}^{(1)}, \ldots, \lambda_{min}^{(K)}$ which occur according to the expected histogram $\mathcal{H}$.

The variation in the key wastage probability is quantified by computing the probability given in (3). Since the network is randomly deployed, the quantity $\frac{\pi r^2}{|\mathcal{A}|}$ is approximately equal to $\frac{d}{N} = 0.005$. Hence, the key wastage probability for the values λ_{min}, μ, and λ_{max} is equal to 0.9704, 0.3858, and 0.0200, respectively.

The variation in the number of potential compromised links is similarly computed for the values λ_{min}, μ, and λ_{max}, yielding 6, 190, and 780 links, respectively.

3.2 Problem Statement

Example 3.1 shows that the use of random key predistribution [Eschenauer and Gligor 2002] induces a binomial distribution $\mathcal{B}(N, \frac{K}{P})$ on the number of nodes that share each seed. As demonstrated, the induced distribution can lead to undesirable tail-effects related to the seeds that are shared by very few or very many nodes in the WSN. The natural question which arises is whether key predistribution schemes can be designed to induce other distributions which do not suffer from the undesirable tail-effects. Moreover, the secondary question which arises is whether it is possible to design universal algorithms for seed assignment which can be used to realize a wide variety of distributions, leading to a general class of application-dependent key predistribution schemes. To the best of our knowledge, there are no existing key predistribution schemes which can address these questions. In fact, any scheme derived from random key predistribution [Eschenauer and Gligor 2002] results in the same binomial distribution and tail-effects as in Example 3.1.

Hence, we aim to characterize the distribution on the number of nodes sharing each seed and the algorithms that can be used to assign seeds to nodes in the WSN. The goal of this characterization is to decouple the distribution from the algorithm used to assign seeds, leading to a class of algorithms that can be used to realize a wide variety of distributions which avoid undesirable tail-effects, thus addressing both of the questions of interest.

4. NETWORK AND SECURITY MODELS

In this section, we state our models and assumptions about the capabilities of adversaries and the deployment of the sensor network.

4.1 Adversarial Model

We assume that adversaries are able to eavesdrop and record transmissions throughout the WSN. Furthermore, we assume that adversaries are able to physically capture sensor nodes and access all information stored within them. We are primarily concerned with adversaries attempting to capture a sufficient number of nodes to compromise a given fraction of the secure links in the WSN. Hence, we do not consider attacks on other network protocols (e.g., node replication, sleep deprivation attacks, wormhole attacks, etc.). We assume that the adversary can capture sensor nodes in any part of the network, and we further assume, as in many recently published works [Eschenauer and Gligor 2002; Liu et al. 2005] that the captured nodes are chosen randomly and independently.

4.2 Network Model

Each sensor is assumed to be equipped with an omni-directional radio with fixed communication range r.[1] Furthermore, a pair of nodes that are within distance r can establish a secure link only if sufficient assigned seeds are shared between

[1]Due to the use of spatial statistics, the area covered by the radio range of a node need not be circular. Hence, this assumption is only necessary to guarantee bidirectional communication between sensor nodes.

them. The wireless network is made up of N sensor nodes deployed randomly (uniformly) over a region $\mathcal{A} \subseteq \mathbb{R}^2$, and the resulting location of node i is given by $x_i \in \mathcal{A}$ for $i = 1, \ldots, N$. The connectivity of the resulting secure WSN is determined with respect to Definition 4.1 as follows.

Definition 4.1. The *connectivity* $\kappa(G)$ of a graph G is defined as the minimum number of vertices that leave a disconnected graph when removed. A graph G with $\kappa(G) \geq k$ is said to be *k-connected*.

A geometric random graph [Penrose 1999; Bettstetter 2002], as given by Definition 4.2, is used to model the physical radio restrictions on the nodes of the sensor network. Furthermore, the shared-seed relation between sensor nodes is modeled using a logical graph as given by Definition 4.3. The combination of the geometric random graph and the logical graph yields a graph-theoretical model for the secure WSN in the form of the restricted network graph as given by Definition 4.4.

Definition 4.2. A *(Euclidean) geometric random graph* $G_g(N, \mathcal{A}, r)$ is the result of random distribution of N vertices in the region $\mathcal{A}$ such that a pair of vertices i and j are adjacent if and only if the (Euclidean) distance between them is no more than r.

Definition 4.3. A *logical graph* $G_L(N, \mathcal{R})$ models a logical relationship between each pair of sensors such that a pair of nodes i and j are adjacent if and only if the pairwise relation $\mathcal{R}$ is satisfied.

Definition 4.4. The *restricted network graph* $G(N, \mathcal{A}, r, \mathcal{R})$ represents a WSN of N nodes deployed over a region $\mathcal{A}$ such that sensors i and j can communicate if and only if they are within distance r and the relation $\mathcal{R}$ is satisfied. The graph G is given by the edgewise intersection of a geometric random graph $G_g(N, \mathcal{A}, r)$ and a logical graph $G_L(N, \mathcal{R})$.

We provide the following results relating to the node degree and the connectivity of the restricted network graph. Theorem 4.7 provides a probabilistic connectivity model which can be used to provide parameters to yield sufficient network connectivity with a desired probability.

LEMMA 4.5. *Given a node u with degree D in the logical graph $G_L(N, \mathcal{R})$, the probability $Pr[d_u \geq k]$ that u has degree at least k in the graph $G(N, \mathcal{A}, r, \mathcal{R})$ is given by*

$$Pr[d_u \geq k] = 1 - e^{-\rho \frac{D+1}{N} \pi r^2} \sum_{i=0}^{k-1} \frac{\left(\rho \frac{D+1}{N} \pi r^2\right)^i}{i!}.$$

PROOF. The vertex density of the geometric random graph $G_g(N, \mathcal{A}, r)$ is given by $\rho = \frac{N}{|\mathcal{A}|}$. The vertices are distributed according to a two-dimensional Poisson point process with rate ρ, so the probability distribution of the number of nodes within distance r of a node is a Poisson distribution [Cressie 1993]. Hence, the probability that the degree d_g of a node is at least k in $G_g(N, \mathcal{A}, r)$

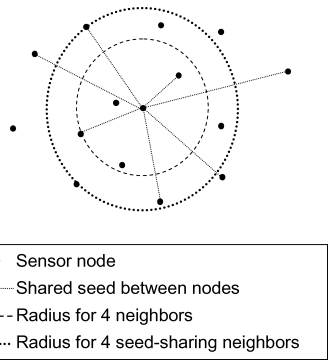


Fig. 2. The radio range of a node in the WSN required for a connected network increases when considering only neighboring nodes which share seeds.

is given by

$$Pr[d_g \geq k] = 1 - e^{-\rho\pi r^2} \sum_{i=0}^{k-1} \frac{(\rho\pi r^2)^i}{i!}. \tag{6}$$

Given that a vertex u has degree D in $G_L(N, \mathcal{R})$, d_u is at least k in $G(N, \mathcal{A}, r, \mathcal{R})$ if and only if at least k of the D neighbors in $G_L(N, \mathcal{R})$ are within distance r of u. Since the neighbors of u in $G_L(N, \mathcal{R})$ are determined independently of the neighbors of u in $G_g(N, \mathcal{A}, r)$, the neighbors of u in $G(N, \mathcal{A}, r, \mathcal{R})$ are uniformly distributed in the region $\mathcal{A}$. Hence, the neighbors of u in $G_L(N, \mathcal{R})$ form a geometric random graph $G_g^u(D+1, \mathcal{A}, r)$, represented by a Poisson point process with rate $\frac{D+1}{|\mathcal{A}|} = \rho\frac{D+1}{N}$. Hence, replacing ρ by $\rho\frac{D+1}{N}$ in (6) completes the proof. □

As suggested in the proof of Lemma 4.5, a decrease in the density of a geometric random graph requires an increase in the radio range r in order to guarantee that the degree d_u of a node u in the graph $G_L(N, \mathcal{R})$ is sufficiently high. This increase in radio range is illustrated in Figure 2. In what follows, we prove that the probability given by Lemma 4.5 is independent for every pair of nodes.

LEMMA 4.6. *In a geometric random graph $G_g(N, \mathcal{A}, r)$, the probability that each of a pair of nodes has degree at least k is independent, i.e., for nodes u and v*

$$Pr[d_u \geq k, d_v \geq k] = Pr[d_u \geq k]Pr[d_v \geq k].$$

PROOF. Let $d_{u\setminus v}$ denote the number of nodes in the region $R_{u\setminus v}$ that is within radius r of node u but not within radius r of node v. Similarly, let $d_{u,v}$ denote the number of nodes in the region $R_{u,v}$ that is within radius r of both u and v. The joint probability $Pr[d_u \geq k, d_v \geq k]$ can be decomposed as

$$Pr[d_u \geq k, d_v \geq k] = \sum_{i \geq k} Pr[d_u \geq k | d_v = i]Pr[d_v = i] \tag{7}$$

$$= \sum_{i \geq k} \left(1 - \sum_{j<k} Pr[d_u = j | d_v = i]\right) Pr[d_v = i]. \tag{8}$$

Noting that $i > j$ in (8), the probability $Pr[d_u = j\,|d_v = i]$ can be expressed as

$$Pr[d_u = j\,|d_v = i] = \sum_{n=0}^{j} Pr[d_u = j\,|d_v = i, d_{u,v} = n]Pr[d_{u,v} = n] \tag{9}$$

$$= \sum_{n=0}^{j} Pr[d_{u\setminus v} = j - n|d_v = i, d_{u,v} = n]Pr[d_{u,v} = n] \tag{10}$$

$$= \sum_{n=0}^{j} Pr[d_{u\setminus v} = j - n]Pr[d_{u,v} = n] \tag{11}$$

$$= \sum_{n=0}^{j} e^{-\rho|R_{u\setminus v}|}\frac{(\rho|R_{u\setminus v}|)^{j-n}}{(j-n)!}e^{-\rho|R_{u,v}|}\frac{(\rho|R_{u,v}|)^{n}}{n!} \tag{12}$$

$$= e^{-\rho\pi r^2}\frac{\rho^j}{j!}\sum_{n=0}^{j}\binom{j}{n}(\pi r^2 - |R_{u,v}|)^{j-n}|R_{u,v}|^n \tag{13}$$

$$= e^{-\rho\pi r^2}\frac{(\rho\pi r^2)^j}{j!} = Pr[d_u = j]. \tag{14}$$

Under the spatial Poisson point process model, the number of points that appear in disjoint regions of $\mathcal{A}$ are independently distributed. Hence, in the above formulation, (11) follows from the fact that the region $R_{u\setminus v}$ is disjoint from both the region $R_{u,v}$ and the region within radio range r of node v. The Poisson process model further allows substitution of the identically distributed probabilities in (12). Equation (13) follows by substituting $|R_{u\setminus v}| = \pi r^2 - |R_{u,v}|$ and collecting terms, and (14) is obtained by applying the binomial theorem and again using the properties of the Poisson point process. Substituting (14) into (8) completes the proof. □

THEOREM 4.7. *The restricted network graph $G(N, \mathcal{A}, r, \mathcal{R})$ resulting from the edgewise intersection of a logical graph $G_L(N, \mathcal{R})$ with average node degree D and a geometric random graph $G_g(N, \mathcal{A}, r)$ with node density $\rho = \frac{N}{|\mathcal{A}|}$ is k-connected with probability $P_G(k)$ given by*

$$P_G(k) = \left(1 - e^{-\rho\frac{D+1}{N}\pi r^2}\sum_{i=0}^{k-1}\frac{\left(\rho\frac{D+1}{N}\pi r^2\right)^i}{i!}\right)^N.$$

PROOF. Applying Lemma 4.6 to each geometric random graph on $(D + 1)$ nodes with density $\rho = \frac{D+1}{|\mathcal{A}|}$ as in Lemma 4.5, the minimum node degree d_{min} in the graph $G(N, \mathcal{A}, r, \mathcal{R})$ is given by

$$Pr[d_{min} \geq k] = Pr[d_1 \geq k, \ldots, d_N \geq k] = Pr[d \geq k]^N. \tag{15}$$

As r increases, a geometric random graph becomes k-connected, asymptotically, as soon as the minimum vertex degree is k with high probability [Penrose 1999]. Hence, the probability of connectivity is given by $P_G(k) = Pr[d_{min} \geq k] = Pr[d \geq k]^N$. □

Theorem 4.7 provides the model for probabilistic k-connectivity used throughout this paper. Several papers on key predistribution have used a

connectivity model based on the assumption that the underlying logical graph is given by a random graph with independent edge probability p. In Corollary 4.8, we show that this random graph model can be approximated by a special case of the model given by Theorem 4.7.

COROLLARY 4.8. *If $G_L(N, \mathcal{R})$ is a random graph with independent edge probability p, the probability $P_G(1)$ given by Theorem 4.7 can be approximated by the result given by Eschenauer and Gligor* [2002].

PROOF. The average vertex degree in a random graph on N vertices with independent edge probability p is given by $D = p(N-1)$, so Theorem 4.7 yields a connectivity probability of

$$P_G(1) = \left(1 - e^{-\rho \frac{p(N-1)+1}{N} \pi r^2}\right)^N \tag{16}$$

$$\approx e^{-Ne^{-\rho \frac{p(N-1)+1}{N} \pi r^2}} \tag{17}$$

$$\approx e^{-Ne^{-\rho p \pi r^2}} \tag{18}$$

where (17) follows from the approximation $1 - x \approx e^{-x}$ for $|x| \ll 1$ and (18) follows by noting that $\frac{p(N-1)+1}{N} \approx p$ for $N \gg 1$. The probability of connectivity stated by Eschenauer and Gligor [2002] using the random graph approach can be expressed as

$$P_c = e^{-Ne^{-\frac{N}{N-1} p(\rho \pi r^2 - 1)}} \tag{19}$$

$$\approx e^{-Ne^{-\rho p \pi r^2}} \tag{20}$$

where (20) follows by noting that $\frac{N}{N-1} p(\rho \pi r^2 - 1) \approx p \rho \pi r^2$ since $\frac{N}{N-1} \approx 1$ and $p \ll 1$. Hence, the connectivity probabilities $P_G(1)$ and P_c are approximately equal for all practical purposes. □

5. SEED ASSIGNMENT FOR KEY PREDISTRIBUTION

In this section, we provide a canonical seed assignment model for key predistribution. We discuss the assignment of seeds to nodes in a WSN and the properties of such seed assignment in terms of a bipartite graph process. Based on the graph-theoretic interpretation of seed assignment, we derive the canonical seed assignment model and discuss the properties of the model. Based on the graph-theoretical interpretation, we propose a sampling framework for seed assignment in the canonical model which decomposes the space of seed assignment algorithms into four classes. Finally, we propose a seed assignment algorithm for each of the four classes.

5.1 Proposed Approach

The assignment of seeds to the nodes of a WSN can be seen as a process on a bipartite graph g with vertex set $V(g) = \mathcal{N} \cup \mathcal{S}$ where the set $\mathcal{N}$ represents the set of N nodes and the set $\mathcal{S}$ represents the set of P seeds. An edge (n, s) in the edge-set $E(g) \subseteq \mathcal{N} \times \mathcal{S}$ represents the assignment of the seed s to the node n.

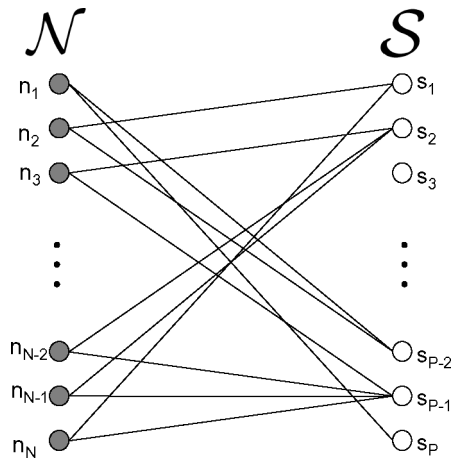


Fig. 3. Bipartite graph g representing the assignment of seeds to nodes in the WSN.

Figure 3 illustrates the use of a bipartite graph g for the assignment of seeds to nodes in the WSN.

For such a bipartite graph g, we can describe the edge-set $E(g)$ in terms of the degree $deg(n)$ of each vertex $n \in \mathcal{N}$ and the degree $deg(s)$ of each vertex $s \in \mathcal{S}$. Similarly, the assignment of seeds to nodes can be described in terms of the number of seeds assigned to each node and the number of nodes which share each seed. As in all schemes in Section 2, we assume that every node receives exactly K seeds, corresponding to $deg(n) = K$ for all $n \in \mathcal{N}$, so the number of edges in g is $|E(g)| = NK$. Hence, we can describe seed assignment in terms of the degrees $deg(s)$ for $s \in \mathcal{S}$ which result from the assignment of seeds to nodes in the WSN. Specifically, we are interested in the probability $Pr[deg(s) = \lambda]$ that a seed s is assigned to exactly λ nodes in the network.

If a desired probability distribution $Pr[deg(s) = \lambda], \lambda = 0, \ldots, N$, on the set $\mathcal{S}$ is given, a graph algorithm is required in order to construct the graph g such that the distribution is realized. However, due to the restriction that every vertex in $\mathcal{N}$ must have degree K, such algorithms may not exist for all values of N and K. An example that illustrates this fact for combinatorial design based key predistribution schemes is discussed by Çamtepe and Yener [2004].

The graph-theoretical interpretation of seed assignment in WSNs is the basis of our canonical seed assignment model. The canonical model is stated formally by the following set of definitions in terms of the bipartite graph g. Table I summarizes the notation for the canonical seed assignment model in WSNs in terms of the graph-theoretical interpretation.

5.2 Canonical Seed Assignment Model

The canonical seed assignment model is primarily concerned with the probability distribution on the degrees of the nodes in $\mathcal{S}$, corresponding to the number of nodes which share each seed. The set of nodes sharing each seed and the probability distribution on the set sizes are defined formally in Definition 5.1 and Definition 5.2.

Table I. Notation for the Canonical Seed Assignment Model in WSNs in Terms of the Graph-Theoretical Interpretation

	Bipartite Graph Process	Canonical Model
g	bipartite graph	seed assignment in WSN
$V(g)$	vertex set of g	set of nodes and seeds
$\mathcal{N}$	vertex partition set of $V(g)$	set of sensor nodes
N	number of vertices in $\mathcal{N}$	number of nodes in WSN
$\mathcal{S}$	vertex partition set of $V(g)$	set of seeds
P	number of vertices in $\mathcal{S}$	number of seeds assigned in WSN
$E(g)$	edge set of g, $E(g) \subseteq \mathcal{N} \times \mathcal{S}$	seed assignment to nodes
$deg(n) = K$	degree K of each vertex $n \in \mathcal{N}$	K seeds assigned to every node
$S(s)$	$\{n : (n, s) \in E(g)\}$	assignment set for seed s
$deg(s) = \lvert S(s)\rvert$	degree of vertex $s \in \mathcal{S}$	number of nodes in assignment set $S(s)$
$\mathcal{P}$	distribution of $deg(s), s \in \mathcal{S}$	assignment distribution
Λ	$\{deg(s) : s \in \mathcal{S}\}$	support of assignment distribution $\mathcal{P}$
μ	average vertex degree in $\mathcal{S}$	mean of distribution $\mathcal{P}$
$\mathscr{A}$	algorithm to construct g	seed assignment algorithm
$(\mathcal{P}, \mathscr{A})$	—	seed assignment scheme
$d(\lambda, \Lambda)$	—	boundary distance, $\min\{\lvert\lambda - \nu\rvert : \nu \in \Lambda\}$

Definition 5.1. The set $S(s) = \{n \in \mathcal{N} : (n, s) \in E(g)\}$ of nodes that are assigned the seed $s \in \mathcal{S}$ is the *assignment set* of seed s.

Definition 5.2. The discrete probability function $\mathcal{P}(\lambda) = Pr[|S(s)| = \lambda]$ specifying the probability that an assignment set S contains exactly λ nodes is the *assignment distribution*. The *support* of an assignment distribution $\mathcal{P}$ is given by $\Lambda = \{\lambda : \mathcal{P}(\lambda) > 0\} \subseteq \{0, \ldots, N\}$.

Given a desired assignment distribution, an algorithm must exist that can realize the given distribution on the set $\mathcal{S}$. Such an algorithm is defined formally in Definition 5.3. The degree of imperfection of a seed assignment algorithm is defined formally in Definition 5.5.

Definition 5.3. The *seed assignment algorithm* $\mathscr{A}$ is used to realize an assignment distribution $\mathcal{P}$, equivalently to construct a bipartite graph g with degree distribution $\mathcal{P}$ on $\mathcal{S}$.

Definition 5.4. A *seed assignment scheme* is given by the pair $(\mathcal{P}, \mathscr{A})$ of an assignment distribution and a seed assignment algorithm.

Definition 5.5. A *boundary set* resulting from a seed assignment scheme $(\mathcal{P}, \mathscr{A})$ is an assignment set $S(s)$ of size $\lambda \notin \Lambda$. The *boundary distance* of such a boundary set is given by $d(\lambda, \Lambda) = \min\{|\lambda - \nu| : \nu \in \Lambda\}$. Boundary sets result from either the algorithm $\mathscr{A}$ or the fact that there are only a finite number of seeds $s \in \mathcal{S}$ with degree $deg(s)$ distributed according to $\mathcal{P}$, referred to hereafter as the *finite sampling effect*.

We give a canonical seed assignment model in WSN in terms of the given definitions. A seed assignment scheme $(\mathcal{P}, \mathscr{A})$ can be characterized entirely by the assignment distribution $\mathcal{P}$ and the seed assignment algorithm $\mathscr{A}$. The performance of a seed assignment scheme $(\mathcal{P}, \mathscr{A})$ can be described in terms of the assignment distribution $\mathcal{P}$, the given set of network parameters, and the

boundary sets that result from the algorithm $\mathscr{A}$ and the finite sampling effects. The desired outcome for a seed assignment scheme $(\mathcal{P}, \mathscr{A})$ is a realization of the assignment distribution $\mathcal{P}$ with no boundary sets. In other words, the histogram representing the values $|\{s \in \mathcal{S} : deg(s) = \lambda\}|$ should be approximately equal to the scaled assignment distribution $P \cdot \mathcal{P}(\lambda)$ for all $\lambda \in \Lambda$, and every node degree $deg(s)$, $s \in \mathcal{S}$ will be a member of Λ.

As illustrated by Example 3.1, the network connectivity and resilience to node capture for a key predistribution scheme depend on the assignment distribution $\mathcal{P}$. Hence, in order to discuss desirable properties and design an assignment distribution for a given application, the effects of the assignment distribution on network connectivity and resilience to node capture must first be investigated. This detailed analysis is presented in Section 6, and the design of assignment distributions is thereafter discussed in Section 7.

As discussed in Section 3.2, we are interested in designing universal seed assignment algorithms that can be used to realize a wide variety of assignment distributions, depending on application requirements. In order to address this problem, we propose a sampling framework for seed assignment algorithms. In the sampling framework, an algorithm can realize a given assignment distribution with minimal occurrence of boundary sets through repeated sampling of the assignment distribution. Such a sampling framework ensures that the analytical characteristics of the seed assignment scheme depend only on the assignment distribution as desired. Hence, in what follows, the sampling framework for seed assignment algorithms is discussed in detail.

5.3 Sampling Framework for Seed Assignment Algorithms

In this section, we propose a sampling framework for seed assignment algorithms. In the framework, the assignment distribution is repeatedly sampled and assignment sets are constructed as a function of the samples of the assignment distribution. We consider algorithms based on random selection using the fundamental combinatorial methods of selection with and without replacement. Furthermore, we consider algorithms of two types. The first type selects an assignment set from $\mathcal{N}$ for each seed subject to the constraint that $deg(n) = K$ for all $n \in \mathcal{N}$. The second type selects a subset of K seeds from $\mathcal{S}$ for each node subject to the constraint that the values of $deg(s)$ for $s \in \mathcal{S}$ are distributed according to the assignment distribution $\mathcal{P}$. Hence, the sampling framework consists of four classes of algorithms.

We provide an example from each of the four classes of seed assignment algorithms in the sampling framework, each of which is named for the corresponding class. The **S**eed **S**election with **R**eplacement (SSR) and **S**eed **S**election with **N**o **R**eplacement (SSNR) algorithms are examples from the classes of selection with and without replacement, respectively, of subsets of $\mathcal{S}$. The **N**ode **S**election with **R**eplacement (NSR) and **N**ode **S**election with **N**o **R**eplacement (NSNR) algorithms are examples from the classes of selection with and without replacement, respectively, of subsets of $\mathcal{N}$. Table II illustrates the four classes of seed assignment algorithms in the sampling framework and classifies each of the four algorithms. In what follows, each algorithm is described in detail,

Table II. The Four Classes of Seed Assignment Algorithms in the Sampling Framework are Based on Whether the Algorithm is Based on Selection With or Without Replacement and Whether the Algorithm Selects Subsets of $\mathcal{S}$ or Subsets of $\mathcal{N}$

	Selection With Replacement	Selection Without Replacement
Subsets of $\mathcal{S}$	SSR	SSNR
Subsets of $\mathcal{N}$	NSR	NSNR

```
SSR(N, K, P)
Φ ← ∅, j ← 1
while Σ_{(s,λ)∈Φ} λ < N · K
    Φ ← Φ ∪ {(s_j, sample(P))}
    j ← j + 1
end while
for n ∈ N
    Φ_0 ← {(s, λ) ∈ Φ : λ > 0}
    E ← select (Φ_0, min(K, |Φ_0|))
    if |E| < K
        F ← select (Φ \ E, K − |E|)
        E ← E ∪ F
    end if
    assign {s : (s, λ) ∈ E} to n
    (s, λ) ← (s, λ − 1) for (s, λ) ∈ E
end for
```

Fig. 4. Code for the SSR seed assignment algorithm.

and code and an illustration are provided for each of the four algorithms. In the code for each algorithm, *select*(X, y) denotes uniform random selection of a subset of y elements from the set X, and *sample*($\mathcal{P}$) denotes the generation of a sample from an assignment distribution $\mathcal{P}$.

5.3.1 *Seed Selection with Replacement (SSR).* The SSR algorithm performs *selection with replacement* from a set Φ containing pairs (s, λ), where $s \in \mathcal{S}$ and $\lambda \in \Lambda$ is a sample of the assignment distribution $\mathcal{P}$. The number of seeds $P = |\mathcal{S}| = |\Phi|$ must be sufficient to provide a total of NK edges in the graph g. Hence, we require $\sum_{(s,\lambda)\in\Phi} \lambda \geq NK$. Once Φ is constructed, seeds are assigned to each node using random selection with replacement. For each of the N nodes, a random selection of K elements of Φ are selected, and the seed s of each selected pair (s, λ) is assigned to the node. The value λ in each selected pair (s, λ) is decremented, and the pair is replaced back into Φ if $\lambda > 0$. Thus, as the algorithm proceeds, $|\Phi|$ decreases. Near the termination of the algorithm, it is possible that $\sum_{(s,\lambda)\in\Phi} \lambda = K$ but $|\Phi| < K$, leading to a case where no set of K unique seeds can be assigned to a remaining node. Hence, if $|\Phi| = K_0 < K$, the $(K - K_0)$ remaining seeds must be selected from those which have already been removed from Φ. If any of the K_0 seeds were initially assigned a sample value of $\lambda_{min} = \min\{\lambda \in \Lambda\}$, these seeds will correspond to boundary sets of size $\lambda_{min} - 1$. Furthermore, if any of the $(K - K_0)$ seeds selected from those that were already removed from Φ were initially assigned a sample value of $\lambda_{max} = \max\{\lambda \in \Lambda\}$, these seeds will correspond to boundary sets of size $\lambda_{max}+1$. Code for the SSR algorithm is provided in Figure 4, and a graphic illustration is provided in Figure 5.

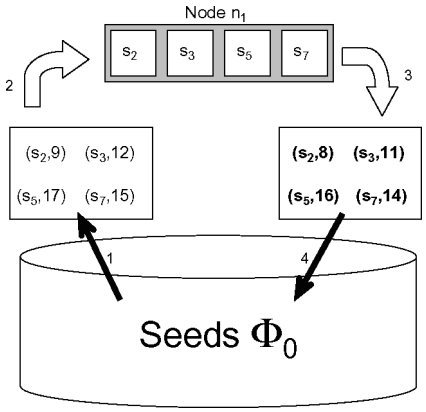


Fig. 5. SSR seed assignment algorithm with numbered steps: 1—select subset of seeds, 2—assign seeds to node, 3—decrement λ for each seed, 4—replace seeds.

5.3.2 *Seed Selection with No Replacement (SSNR).* The SSNR algorithm performs *selection without replacement* from a set Φ containing pairs (s, λ) where $s \in \mathcal{S}$ and $\lambda \in \Lambda$ is a sample of the assignment distribution $\mathcal{P}$. The number of seeds $P = |\mathcal{S}| = |\Phi|$ must be sufficient to provide a total of NK edges in the graph g. Hence, we require $\sum_{(s,\lambda)\in\Phi} \lambda \geq NK$. Once Φ is constructed, seeds are assigned to each node using random selection without replacement in a total of $\lambda_{max} = \max\{\lambda \in \Lambda\}$ rounds. In a single round, which continues as long as Φ is nonempty, a random subset of K pairs (s, λ) in Φ is selected without replacement for each subsequent node, and the value λ in each selected pair is decremented. Pairs (s, λ) such that $\lambda = 0$ are permanently removed from Φ for all subsequent rounds, so the initial size of Φ can decrease in every subsequent round. In a given round, if K is not a factor of $|\Phi|$, there will be $K_0 < K$ seeds remaining for the last node of the round. These K_0 seeds can be combined with a random selection of $(K - K_0)$ seeds which have not been permanently removed from Φ. The $(K - K_0)$ selected pairs will then be excluded from the subsequent round of the algorithm. If this occurs in the Kth round, any of the $(K - K_0)$ selected seeds which were initially assigned a sample value of $\lambda_{max} = \max\{\lambda \in \Lambda\}$ will yield a boundary set of size $\lambda_{max} + 1$. Code for the SSNR algorithm is provided in Figure 6, and a graphic illustration is provided in Figure 7.

5.3.3 *Node Selection with Replacement (NSR).* The NSR algorithm performs *selection with replacement* from a set Φ containing pairs (n, c) where $n \in \mathcal{N}$ and $c \geq 0$ counts the number of seeds assigned to node n. For each seed, a sample λ is generated from the assignment distribution $\mathcal{P}$, and a set of λ pairs (n, c) are selected from Φ. The assignment set for the given seed is composed of the n entries in the λ selected pairs. Each time a pair (n, c) is selected, the counter c is incremented, and the pair is replaced back into Φ only if $c < K$. Hence, $|\Phi|$ decreases as the algorithm proceeds. As soon as $|\Phi| < \lambda_{max} = \max\{\lambda \in \Lambda\}$, it is possible for the sampled value of λ to be less than $|\Phi|$, so the entire set Φ is selected. If $|\Phi| < \lambda_{min} = \min\{\lambda \in \Lambda\}$, this will lead to boundary sets that vary in size between 1 and $\lambda_{min} - 1$. In simulation, a

```
SSNR(N, K, P)
Φ, Φ1 ← ∅, j ← 1
while Σ_{(s,λ)∈Φ} λ < N · K
    Φ ← Φ ∪ {(s_j, sample(P))}
    j ← j + 1
end while
λ_max = max{λ : (s, λ) ∈ Φ}
for i from 1 to λ_max
    Φ0 ← Φ \ Φ1
    while |Φ0| ≥ K
        E ← select(Φ0, K)
        Φ0 ← Φ0 \ E
        assign {s : (s, λ) ∈ E} to next n ∈ N
        (s, λ) ← (s, λ − 1) for (s, λ) ∈ E
    end while
    Φ ← Φ \ {(s, λ) : λ = 0}
    if |Φ0| > 0
        Φ1 ← select (Φ \ Φ0, K − |Φ0|)
        assign {s : (s, λ) ∈ Φ0 ∪ Φ1} to
            next n ∈ N
        (s, λ) ← (s, λ − 1) for
            (s, λ) ∈ Φ0 ∪ Φ1
    end if
end for
```

Fig. 6. Code for the SSNR seed assignment algorithm.

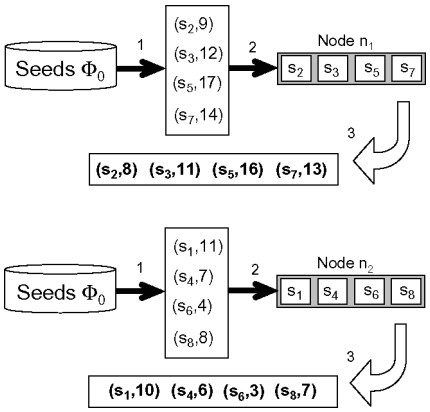


Fig. 7. SSNR seed assignment algorithm with numbered steps: 1—select subset of seeds, 2—assign seeds to node, 3—decrement λ for each seed.

majority of the boundary sets which occur have size much smaller than $\lambda_{min} - 1$. Code for the NSR algorithm is provided in Figure 8, and a graphic illustration is provided in Figure 9.

5.3.4 *Node Selection with No Replacement (NSNR).* The NSNR algorithm performs *selection without replacement* from the set Φ, initially equal to $\mathcal{N}$. Assignment sets are generated using random selection without replacement in a total of K rounds. In a single round, which continues as long as Φ is nonempty,

NSR$(N, K, \mathcal{P})$
$\Phi \leftarrow \{(n, 0) : n \in \mathcal{N}\}$
while $|\Phi| > 0$
  $\lambda \leftarrow sample(\mathcal{P})$
  $S \leftarrow select\,(\Phi, \min(\lambda, |\Phi|))$
  assign next $s \in \mathcal{S}$ to $\{n : (n, c) \in S\}$
  $(n, c) \leftarrow (n, c + 1)$ for $(n, c) \in S$
  $\Phi \leftarrow \Phi \setminus \{(n, c) \in S : c = K\}$
end while

Fig. 8. Code for the NSR seed assignment algorithm.

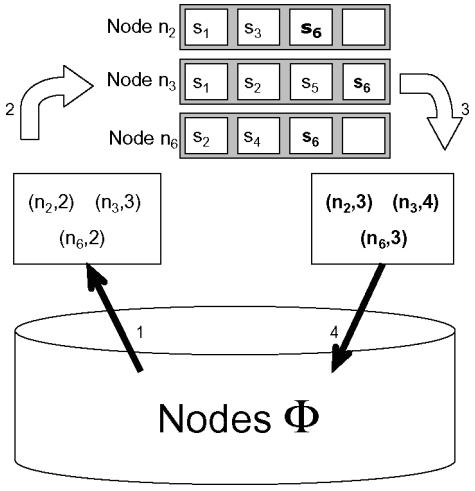


Fig. 9. NSR seed assignment algorithm with numbered steps: 1—select subset of nodes, 2—assign seed to nodes, 3—increment c for each node, 4—replace nodes.

a sample λ is generated from the assignment distribution $\mathcal{P}$, a set of λ nodes in Φ is selected for each subsequent seed, and the seed is assigned to the selected nodes. If the sample λ is such that $|\Phi| < \lambda$, the seed is assigned to the $|\Phi|$ remaining nodes and a random selection of $(\lambda - |\Phi|)$ other nodes, which are then removed from the subsequent round. In the Kth round, since we do not want to assign $(K + 1)$ seeds to any node, the final seed may be assigned to less than $\lambda_{min} = \min\{\lambda \in \Lambda\}$ nodes, resulting in a single boundary set of size between 1 and $\lambda_{min} - 1$. We note that if $\Lambda = \{\lambda\}$ and λ is a factor of N, the NSNR algorithm will not yield boundary sets, and the result of the algorithm is equivalent to a deterministic seed assignment algorithm similar to those of Çamtepe and Yener [2004] and Lee and Stinson [2004]. Code for the NSNR algorithm is provided in Figure 10, and a graphic illustration is provided in Figure 11.

The algorithms proposed herein yield a result that is essentially similar. The primary differences are the behavior of the boundary sets that result and their computational cost. Though these sets occur nondeterministically, their general behavior can be characterized. Furthermore, there tends to be a trade-off between the computational cost of an algorithm and the resulting boundary

```
NSNR(N, K, P)
Φ1 ← ∅
for i from 1 to K
    Φ ← N \ Φ1
    while |Φ| > 0
        λ ← sample(P)
        S ← select (Φ, min(λ, |Φ|))
        if |S| < λ and i < K
            Φ1 ← select (N \ S, λ − |S|)
            S ← S ∪ Φ1
        end if
        assign next s ∈ S to {n ∈ S}
        Φ ← Φ \ S
    end while
end for
```

Fig. 10. Code for the NSNR seed assignment algorithm.

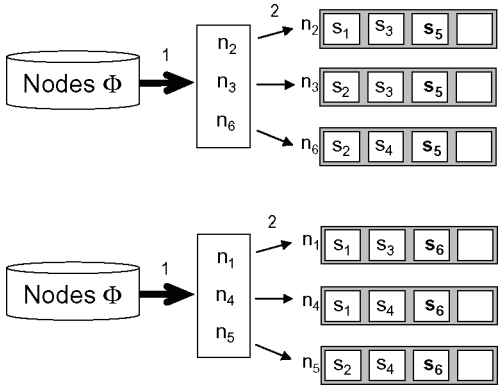


Fig. 11. NSNR seed assignment algorithm with numbered steps: 1—select subset of nodes, 2—assign seed to nodes.

distance, in that the boundary distance can be decreased at the expense of increased computation. Hence, the choice of algorithm may depend on the desired boundary distance tolerance and the allowable computational cost.

6. ANALYSIS OF SEED ASSIGNMENT SCHEMES

In this section, we provide general analysis for a seed assignment scheme $(\mathcal{P}, \mathscr{A})$ assuming the impact of any boundary sets is negligible. We compute the probability that a pair of nodes share a given number of seeds, the probability of network connectivity, and the resilience to node capture. Each of the quantities is provided in such a way that the worst case with respect to the assignment distribution $\mathcal{P}$ can be easily determined. Furthermore, the average case is computed for each quantity with respect to the assignment distribution $\mathcal{P}$. The average case is most helpful in determining sufficiency of network parameters, while the worst case is most helpful in determining whether a given set of parameters will result in undesirable tail-effects, as discussed in Section 3.1.

6.1 Probability of Sharing Seeds

The average and worst-case analysis of a key predistribution scheme can be performed with respect to the probability that a pair of nodes share any number of seeds. In addition to performance analysis, this probability is important for various applications based on local connectivity properties. For example, the q-composite scheme of Chan et al. [2003] requires a pair of nodes to share at least q keys for some $q \geq 1$. We compute the probability $p_s(i)$ that a pair of nodes share exactly i seeds as a function of the assignment set sizes λ corresponding to the seeds in each node. We then compute the average probability taken over the assignment distribution $\mathcal{P}$.

LEMMA 6.1. *A node u containing a seed s, such that $\lambda = |S(s)|$ is known, will share s with a node v with probability $p_\lambda = \frac{\lambda-1}{N-1}$.*

PROOF. Given a node u containing s, exactly $(\lambda - 1)$ of the remaining $(N - 1)$ nodes contain s. Hence, the probability that v is one of these $(\lambda - 1)$ nodes is $\frac{\lambda-1}{N-1}$. □

THEOREM 6.2. *A node u containing seeds $s_1, \ldots, s_K$, such that $\lambda_j = |S(s_j)|$ for $j = 1, \ldots, K$ are known, will share exactly i seeds with a node v with probability $p_s(i, \lambda_1, \ldots, \lambda_K)$ given by*

$$p_s(i, \lambda_1, \ldots, \lambda_K) = \frac{1}{i!(K-i)!} \sum_{\pi} \left(\prod_{j=1}^{i} \frac{\lambda_{\pi_j} - 1}{N-1} \times \prod_{j=i+1}^{K} \frac{N - \lambda_{\pi_j}}{N-1} \right)$$

where the summation is over all permutations $\pi = (\pi_1, \ldots, \pi_K)$ of $(1, \ldots, K)$.

PROOF. The event that v shares s_j with u can be modeled as a Bernoulli trial with success probability p_{λ_j} given by Lemma 6.1. Since the assignment sets are chosen independently, the K events are independent. Hence, the number of events i that occur is given by the sum of the K independent Bernoulli random variables. The probability that exactly i of the K events occur is given by the sum over all possible choices of i of the K events. For a given choice of i events, the contribution to the overall probability is the product of p_{λ_j} for the i events which occur multiplied by the product of $1 - p_{\lambda_j}$ for the $(K - i)$ events which do not occur. The term $\frac{1}{i!(K-i)!}$ is added to compensate for the $i!(K-i)!$ permutations that result in the same choice of i events. □

THEOREM 6.3. *A node u will share exactly i seeds with a node v with probability $p_s(i)$ given by*

$$p_s(i) = \binom{K}{i} \left(\frac{\mu - 1}{N-1} \right)^i \left(\frac{N - \mu}{N-1} \right)^{K-i}$$

where μ is the average assignment set size according to the assignment distribution $\mathcal{P}$.

PROOF. The probability $p_s(i)$ can be computed by taking the expected value of the probability $p_s(i, \lambda_1, \ldots, \lambda_K)$ given in Theorem 6.2 with respect to the set of samples $\lambda_1, \ldots, \lambda_K$. Hence, letting $\mathcal{E}[\cdot]$ represent this expected value, $p_s(i)$ is

given by

$$p_s(i) = \mathcal{E}\left[\frac{1}{i!(K-i)!}\sum_{\pi}\left(\prod_{j=1}^{i}\frac{\lambda_{\pi_j}-1}{N-1}\prod_{j=i+1}^{K}\frac{N-\lambda_{\pi_j}}{N-1}\right)\right]. \tag{21}$$

Since the samples λ_j are independent, this is equivalent to taking the expected value with respect to each λ_j. Moving the expected value within the summation and using the independence of the λ_j yields

$$p_s(i) = \frac{1}{i!(K-i)!}\sum_{\pi}\left(\prod_{j=1}^{i}\frac{\mathcal{E}_{\pi_j}[\lambda_{\pi_j}]-1}{N-1}\times\prod_{j=i+1}^{K}\frac{N-\mathcal{E}_{\pi_j}[\lambda_{\pi_j}]}{N-1}\right). \tag{22}$$

Identical distribution of the λ_j suggests that each $\mathcal{E}_{\pi_j}[\lambda_{\pi_j}]$ is equal to the mean μ of the assignment distribution $\mathcal{P}$. The product terms are thus independent of the index j, and the summands are independent of the permutation π, so the sum-of-products form is replaced by a single product of powers with coefficient $\frac{K!}{i!(K-i)!}$. Replacing this coefficient with $\binom{K}{i}$ completes the proof. □

Theorem 6.2 and Theorem 6.3 are useful in respectively determining the worst-case and average probability of sharing seeds. Theorem 6.2 is particularly applicable to the worst-case analysis in that it can be used to compute the worst-case probability of sharing seeds regardless of how the worst case is defined. For example, the designer of the key predistribution scheme can design an assignment distribution based on a given tolerance to one minimal λ value by bounding the probability $1-p_s(0,\lambda_{min},\mu,\ldots,\mu)$. The designer can similarly design the key predistribution scheme based on the expected worst-case probability by bounding the probability $1-p_s(0,\lambda_{min}^{(1)},\ldots,\lambda_{min}^{(K)})$ where $\lambda_{min}^{(1)},\ldots,\lambda_{min}^{(K)}$ are order statistics similar to those discussed in Section 3.1.

6.2 Network Connectivity

The probability of connectivity of the secure WSN is given by Theorem 4.7 in Section 4.2 as a function of the expected node degree D in the logical graph $G_L(N,\mathcal{R})$. For simplicity, we assume the relation $\mathcal{R}$ is true if and only if the given pair of nodes share at least one seed. Similar results can be derived for the modified relations of schemes such as the q-composite scheme [Chan et al. 2003].

We note that there are two forms of randomness present in a seed assignment algorithm. The number of nodes λ is sampled randomly from the assignment distribution $\mathcal{P}$, and the assignment set of λ nodes is selected randomly. We first compute the expected degree $d(u)$ of a node u assuming the sizes $\lambda_1,\ldots,\lambda_K$ of the K assignment sets corresponding to the seeds stored in node u are fixed and known. This computation is performed using a combinatorial occupancy problem in which each pair (u,v), for $v\in\mathcal{N}\setminus\{u\}$, is represented by a bin and a shared seed between nodes u and v is represented by a ball in the bin representing the pair (u,v). The assignment of a seed s_j to node u and (λ_j-1) of the $(N-1)$ other nodes thus corresponds to placing one ball in each of (λ_j-1) of the $(N-1)$ bins. This occupancy problem is illustrated in Figure 12. The degree $d(u)$ of node u in the graph $G_L(N,\mathcal{R})$ is given by the number of bins

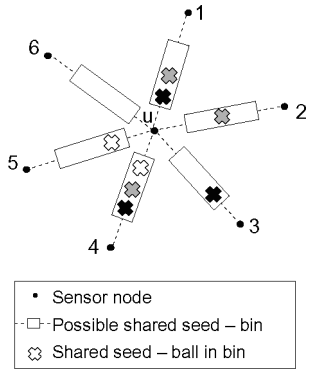


Fig. 12. Seed assignment to nodes in the WSN is represented by a combinatorial occupancy problem where each pair of nodes (u, v) is represented by a bin, and a shared seed between nodes u and v is indicated by a ball in the bin (u, v).

(u, v) which contain at least one ball. The expected node degree D is computed by taking the expected value of the node degree $d(u)$ over all possible values of $\lambda_1, \ldots, \lambda_K$ according to a given assignment distribution $\mathcal{P}$.

LEMMA 6.4. *A node u with seeds $s_1, \ldots, s_K$, such that $\lambda_j = |S(s_j)|$ for $j = 1, \ldots, K$ are known, will not share a seed with $e(u)$ nodes according to the probability $Pr[e(u) \geq E]$ given by*

$$Pr[e(u) \geq E] = \sum_{m=E}^{N-1} (-1)^{m-E} \binom{m-1}{E-1} \binom{N-1}{m} \prod_{j=1}^{K} \frac{\binom{N-1-m}{\lambda_j - 1}}{\binom{N-1}{\lambda_j - 1}}.$$

PROOF. Placing $(\lambda_j - 1)$ balls in $(N-1)$ bins such that a given set of m bins remain empty can be done in exactly $\binom{N-1-m}{\lambda_j - 1}$ ways. Thus, the number of ways to assign K seeds in such a way that a particular set of m bins remains empty is given by the product $\prod_{j=1}^{K} \binom{N-1-m}{\lambda_j - 1}$. The number of ways to select the m bins to remain empty is $\binom{N-1}{m}$. By the Inclusion-Exclusion Principle [Johnson 1980], the number of ways $M(E)$ that K subsets of bins can be chosen such that at least E bins remain empty is given by

$$M(E) = \sum_{m=E}^{N-1} (-1)^{m-E} \binom{m-1}{E-1} \binom{N-1}{m} \prod_{j=1}^{K} \binom{N-1-m}{\lambda_j - 1}. \tag{23}$$

Dividing $M(E)$ by the total number of ways to choose the K subsets given by $M(0)$ yields the probability that at least E bins remain empty. □

THEOREM 6.5. *A node u with seeds $s_1, \ldots, s_K$, such that $\lambda_j = |S(s_j)|$ for $j = 1, \ldots, K$ are known, will have expected degree $\mathcal{E}[d(u)]$ in the logical graph $G_L(N, \mathcal{R})$ given by*

$$\mathcal{E}[d(u)] = (N-1)\left(1 - \prod_{j=1}^{K} \frac{\lambda_j - 1}{N-1}\right).$$

PROOF. The expected number of empty bins $\mathcal{E}[e(u)]$ can be computed using the fact that

$$\mathcal{E}[e(u)] = \sum_{E=1}^{N-1} Pr[e(u) \geq E] \tag{24}$$

since $e(u)$ is a nonnegative discrete random variable [Feller 1957]. Substituting the result of Lemma 6.4 into (24) provides an expression for $\mathcal{E}[e(u)]$. The expected degree $\mathcal{E}[d(u)]$ is then given by

$$\mathcal{E}[d(u)] = N - 1 - \mathcal{E}[e(u)] \tag{25}$$

because each nonempty bin corresponds to an edge in the graph $G_L(N, \mathcal{R})$. Replacing $\mathcal{E}[e(u)]$ with the result from Lemma 6.4 yields

$$\mathcal{E}[d(u)] = N - 1 - \sum_{E=1}^{N-1} \sum_{m=E}^{N-1} (-1)^{m-E} \binom{N-1}{m} \binom{m-1}{E-1} \prod_{j=1}^{K} \frac{\binom{N-1-m}{\lambda_j - 1}}{\binom{N-1}{\lambda_j - 1}}. \tag{26}$$

The order of summation can be reversed by changing the limits of summation to sum over $m = 1, \ldots, N-1$ and $E = 1, \ldots, m$. Terms that are independent of E can then be moved outside of the inner summation, yielding

$$\mathcal{E}[d(u)] = N - 1 - \sum_{m=1}^{N-1} \binom{N-1}{m} \prod_{j=1}^{K} \frac{\binom{N-1-m}{\lambda_j - 1}}{\binom{N-1}{\lambda_j - 1}} \sum_{E=1}^{m} (-1)^{m-E} \binom{m-1}{E-1}. \tag{27}$$

The binomial theorem suggests that

$$\sum_{E=1}^{m} (-1)^{m-E} \binom{m-1}{E-1} = \sum_{E=0}^{m-1} (-1)^{m-1-E} \binom{m-1}{E} = 0^{m-1}. \tag{28}$$

Since $0^0 = 1$, the only nonzero term of the summation is when $m = 1$. Hence the expected degree of node u is given by

$$\mathcal{E}[d(u)] = N - 1 - \binom{N-1}{1} \prod_{j=1}^{K} \frac{\binom{N-2}{\lambda_j - 1}}{\binom{N-1}{\lambda_j - 1}} \tag{29}$$

$$= (N-1)\left(1 - \prod_{j=1}^{K} \frac{(N-2)!(\lambda_j - 1)!(N - \lambda_j)!}{(\lambda_j - 1)!(N - \lambda_j - 1)!(N-1)!}\right) \tag{30}$$

$$= (N-1)\left(1 - \prod_{j=1}^{K} \frac{N - \lambda_j}{N-1}\right). \tag{31}$$

□

THEOREM 6.6. *The expected node degree D in the logical graph $G_L(N, \mathcal{R})$ is given by*

$$D = (N-1)\left(1 - \left(\frac{N-\mu}{N-1}\right)^K\right).$$

PROOF. The expected node degree D is computed by taking the expected value of $\mathcal{E}[d(u)]$ given by Theorem 6.5 with respect to each of the set of random variables $\lambda_1, \ldots, \lambda_K$. Denoting this expected value by $\mathcal{E}[\cdot]$ yields

$$D = \mathcal{E}\left[(N-1)\left(1 - \prod_{j=1}^{K} \frac{N-\lambda_j}{N-1}\right)\right]. \tag{32}$$

Since the samples $\lambda_1, \ldots, \lambda_K$ are independent, this is equivalent to taking the expected value with respect to each of the random variables λ_j, denoted by $\mathcal{E}_j[\cdot]$. This independence yields

$$D = (N-1)\left(1 - \prod_{j=1}^{K} \frac{N-\mathcal{E}_j[\lambda_j]}{N-1}\right). \tag{33}$$

Identical distribution of the λ_j suggests that $\mathcal{E}_j[\lambda_j]$ can be replaced by the mean μ of the assignment distribution $\mathcal{P}$ completing the proof. □

The result of Theorem 6.6 can then be used in conjunction with Theorem 4.7 to yield the probability $P_G(k)$ that the restricted network graph $G(N, \mathcal{A}, r, \mathcal{R})$ is k-connected. Hence, given the number N of sensors in the network, seed storage K, desired connectivity k, deployment density ρ, and radio range r, the mean μ of the assignment distribution $\mathcal{P}$ can be chosen to guarantee k-connectivity with the desired probability.

6.3 Resilience to Attacks

Since every seed is assigned to multiple nodes, a seed may be used to establish many secure links throughout the network. Thus, an adversary who randomly captures nodes may be able to decrypt secure communication links between uncaptured nodes, referred to as *link compromise*. The average probability of link compromise $f(x)$ due to the capture of x nodes often depends on the underlying structure of the key predistribution scheme. Hence, for generality, our primary security metric is the probability $p(m, x)$ that exactly m of the x captured nodes contain a given seed. Similar to the results of Section 6.1, we first compute the results when the assignment set size λ of the given seed is fixed and known, and then compute the average probability as a function of the assignment distribution $\mathcal{P}$.

LEMMA 6.7. *Given uncaptured nodes u and v that share a seed s such that $\lambda = |S(s)|$ is known, the probability $p(m, x, \lambda)$ that exactly m of the x captured nodes contain s is given by*

$$p(m, x, \lambda) = \sum_{I} \left(\prod_{j=1}^{m} \frac{\lambda - j - 1}{N - I_j - 1} \times \prod_{i \notin I} \frac{N - \lambda - i + m_i + 1}{N - i - 1} \right)$$

where the summation is taken over all vectors $I = (I_1, \ldots, I_m)$ such that $1 \leq I_1 < \cdots < I_m \leq x$ and $m_i = \max\{h : I_h < i\}$.

PROOF. Each successive node capture can be modeled as a Bernoulli trial that is successful if an additional copy of the seed s is contained in the captured node. The success probability of the xth trial, however, depends on the number of previously successful trials because the maximum number of successful trials is fixed at λ. Hence, the Bernoulli trials are not independent. Letting $I = (I_1, \ldots, I_m)$ represent the indices of the m successful trials out of the x attempts. In trial i, given that m_i nodes containing s have been captured, the probability that one of the $\lambda - 2 - m_i$ nodes containing the seed s was selected randomly from the $(N-2)-(i-1)$ nodes remaining in the network is given by $\frac{\lambda-2-m_i}{N-i-1}$. The number of previously captured nodes m_i is given by the number of indices I_h in I with $h < i$, i.e., $m_i = \max\{h : I_h < i\}$. The contribution $p(m, x, \lambda, I)$ for a given vector I is thus equal to the product of the success probabilities for the m trials $I_1, \ldots, I_m$ and the failure probabilities for the $(x-m)$ remaining trials given by

$$p(m, x, \lambda, I) = \prod_{i \in I} \frac{\lambda - 2 - m_i}{N - i - 1} \times \prod_{i \notin I} \frac{N - \lambda - i + m_i + 1}{N - i - 1}. \tag{34}$$

For $I_j \in I$, the value of m_{I_j} is simply given by the number of prior successes $(j-1)$. Hence, the contribution $p(m, x, \lambda, I)$ for a given vector I is given by

$$p(m, x, \lambda, I) = \prod_{j=1}^{m} \frac{\lambda - j - 1}{N - I_j - 1} \times \prod_{i \notin I} \frac{N - \lambda - i + m_i + 1}{N - i - 1}. \tag{35}$$

The final result is obtained by summing over all possible I. □

LEMMA 6.8. *Given uncaptured nodes u and v that share a seed s such that* $\lambda = |S(s)|$ *is known, if* $x \ll N - 2$ *and* $m \ll \lambda - 2$*, the probability* $p(m, x, \lambda)$ *that exactly m of the x captured nodes contain s can be approximated as*

$$p(m, x, \lambda) \approx \binom{x}{m} \left(\frac{\lambda - 2}{N - 2}\right)^m \left(\frac{N - \lambda}{N - 2}\right)^{x-m}.$$

PROOF. If $x \ll N$ and $m \ll \lambda - 2$ and $m_i = \max\{h : I_h < i\}$ as in Lemma 6.7, then the approximations

$$\frac{(\lambda - 2) - (m_i - 1)}{(N - 2) - (x - 1)} \approx \frac{\lambda - 2}{N - 2} \tag{36}$$

$$\frac{(N - 2) - (\lambda - 2 - m_i) - (x - 1)}{(N - 2) - (x - 1)} \approx \frac{(N - 2) - (\lambda - 2)}{N - 2} = \frac{N - \lambda}{N - 2} \tag{37}$$

can be substituted into the result of Lemma 6.7, yielding

$$p(m, x, \lambda) = \sum_{I} \left(\prod_{j=1}^{m} \frac{\lambda - 2}{N - 2} \times \prod_{i \notin I} \frac{N - \lambda}{N - 2} \right). \tag{38}$$

Each product term is independent of the indices i and j, so the result reduces to

$$p(m, x, \lambda) = \sum_{I} \left(\frac{\lambda - 2}{N - 2}\right)^m \left(\frac{N - \lambda}{N - 2}\right)^{x-m}. \tag{39}$$

Furthermore, the summand is independent of the index I, so the summation over I can be replaced by the summand multiplied by $\binom{x}{m}$ corresponding to the number of possible vectors I. □

THEOREM 6.9. *Given uncaptured nodes u and v that share a seed s, the probability $p(m, x)$ that exactly m of the x captured nodes contain s can be approximated as*

$$p(m, x) \approx \binom{x}{m} \left(\frac{\mu - 2}{N - 2}\right)^m \left(\frac{N - \mu}{N - 2}\right)^{x-m}$$

where μ is the mean of a given assignment distribution $\mathcal{P}$.

PROOF. This result is an approximation to the result of Lemma 6.8 obtained by replacing the λ by the mean μ of the random variable λ with respect to the assignment distribution $\mathcal{P}$. □

The approximations in Lemma 6.8 and Theorem 6.9 are useful in respectively approximating the worst-case and average probability of link compromise. The average probability of link compromise $f(x)$ is dependent on the application and the link-key establishment protocol, though it is typically a function of $p(m, x)$ approximated by Theorem 6.9. Since $p(m, x)$ depends only on the mean μ of the assignment distribution $\mathcal{P}$, the network size N, and the number of captured nodes x, it can be a useful metric in designing the assignment distribution $\mathcal{P}$.

The probability of link compromise $f(x, \lambda)$ for a link secured by a seed shared by λ nodes is also application- and protocol-dependent, though it is typically a function of $p(m, x, \lambda)$ approximated by Lemma 6.8. The worst-case probability of link compromise can thus be computed as $f(x, \lambda_{max})$ where λ_{max} is similar to that discussed in Section 3.1. Since $p(m, x, \lambda_{max})$ depends only on the maximum value λ_{max} in the support of the assignment distribution $\mathcal{P}$, the network size N, and the number of captured nodes x, it can be a useful metric in designing the assignment distribution $\mathcal{P}$.

7. ASSIGNMENT DISTRIBUTIONS

Because the algorithms in the sampling framework presented in Section 5.3 can realize a given assignment distribution $\mathcal{P}$ with negligible occurrence of boundary sets, the assignment distributions can be designed independently of the seed assignment algorithms. Hence, assignment distributions can be designed with respect to the analytical results in Section 6 in terms of average and worst-case network connectivity and resilience to node capture. However, the finite sampling effects described in Definition 5.5 must still be considered in the design of an assignment distribution.

In general, the design of an assignment distribution depends highly on the application requirements and link-key establishment scheme. Furthermore, the average network connectivity and resilience to node capture depend only on the mean μ of the assignment distribution $\mathcal{P}$. Hence, the optimal assignment distribution is *application specific*.

The design of an assignment distribution $\mathcal{P}$ can be broken into two primary steps. The first step is to determine the support Λ of the assignment

distribution, and the second step is to determine the probability mass $\mathcal{P}(\lambda)$ for every $\lambda \in \Lambda$.

7.1 Assignment Distribution Support

In order to compensate for finite sampling effects, the size of the support Λ of the assignment distribution $\mathcal{P}$ should be larger than 1. In contrast, however, the size of Λ should be as small as possible to avoid the undesirable tail-effects discussed in Section 3.1. Though not a requirement, we assume the support Λ is a contiguous subset of $\{0, \ldots, N\}$, i.e., if $\lambda_1, \lambda_2 \in \Lambda$ then $\lambda \in \Lambda$ for all $\lambda \in \{0, \ldots, N\}$ such that $\lambda_1 \leq \lambda \leq \lambda_2$. Furthermore, Λ should contain the values nearest to the average value μ of the assignment distribution $\mathcal{P}$ required for sufficient network connectivity as given by Theorem 4.7 and Theorem 6.6. Hence, the design of the support Λ is equivalent to determination of $\lambda_{min} = \min\{\lambda \in \Lambda\}$ and $\lambda_{max} = \max\{\lambda \in \Lambda\}$ such that $\lambda_{min} \leq \mu \leq \lambda_{max}$.

In order to determine the value of λ_{min}, we consider the worst-case probability of sharing seeds $p_s(i, \lambda_{min}, \ldots, \lambda_{min})$ as given by Theorem 6.2. Similarly, to determine the value of λ_{max}, we consider the worst-case resilience to node capture in terms of the probability $p(m, x, \lambda_{max})$ as given by Lemma 6.7 and approximated by Lemma 6.8. Furthermore, we must consider the finite sampling effects that arise due to the choice of λ_{min}, λ_{max}, and the seed assignment algorithms. For the SSR and SSNR algorithms, only boundary sets with distance 1 can occur, so the finite sampling effects can be seen as negligible. However, for the NSR and NSNR algorithms, boundary sets with distance between 1 and $\lambda_{min} - 1$ may occur. Hence, for the NSR and NSNR algorithms, we are interested in minimizing the boundary distance of the resulting boundary sets. Through simulation, we note that as the value $|\Lambda| = \lambda_{max} - \lambda_{min} + 1$ decreases, the distance of boundary sets due to finite sampling effects tends to increase. Thus, to avoid boundary sets with large distance, λ_{max} should be increased and λ_{min} should be decreased. Hence, there exists a trade-off among improving the worst-case probability of sharing seeds, improving the worst-case resilience to node capture, and minimizing the boundary distance of boundary sets which occur due to finite sampling effects. Therefore, determining the optimal values of λ_{min} and λ_{max} is application-dependent.

7.2 Probability Mass on Λ

Once the support Λ of the assignment distribution $\mathcal{P}$ is determined, the probability mass $\mathcal{P}(\lambda)$ for each $\lambda \in \Lambda$ must be determined. However, if $|\Lambda| > 1$, there are an uncountably infinite number of possible assignment distributions for given values of μ, λ_{min}, and λ_{max}, leading to a high degree of freedom in determining the assignment distribution $\mathcal{P}$.

As worst-case probability of sharing seeds and the worst-case resilience to node capture are best mitigated by an assignment distribution with trivial support $|\Lambda| = 1$, we approximate this performance by placing more probability mass on the values of λ nearest to μ, resulting in an assignment distribution which is peaked near μ and decreases as $|\mu - \lambda|$ increases.

7.3 Illustration of Assignment Distribution Design

We provide the following example to illustrate the design of an assignment distribution for a given link-key establishment scheme and set of network parameters.

Example 7.1. Let a WSN of $N = 5,000$ nodes with $K = 100$ seeds per node and a radio range of $r = 40m$ be deployed over a region $\mathcal{A}$ of area $|\mathcal{A}| = 0.5km^2$ such that 2-connectivity is desired with probability 0.99. We assume that any nodes sharing at least one seed can establish a link-key as a function of the shared seeds and a link can be compromised as soon as the seeds used to compute the link-key are captured. Furthermore, we assume that the value λ_{min} must be such that the worst-case probability of sharing seeds is within 20% of the average probability of sharing seeds, and the value λ_{max} must be such that the worst-case probability of link compromise is within 20% of the average probability of link compromise for $x = 50$ captured nodes.

Theorem 4.7 yields a minimum average vertex degree of $D = 1,813$ in the logical graph $G_L(N, \mathcal{R})$. Theorem 6.6 yields a minimum average assignment set size of $\mu \geq 23.47$. The average probability of sharing at least 1 seed is given by Theorem 6.3 as $(1 - p_s(0)) = 0.3627$. Hence, the value of λ_{min} must result in $(1 - p_s(0, \lambda_{min}, \ldots, \lambda_{min})) \geq 0.3$. Theorem 6.2 yields $\lambda_{min} \geq 19$. The value of λ_{max} must result in $1 - p(0, 50, \lambda_{max}) \leq 0.25$. Lemma 6.8 yields $\lambda_{max} \leq 30$. Hence, we choose the support $\Lambda = \{19, \ldots, 28\}$ and the symmetric probability mass function $\mathcal{P}$ given by

$$\mathcal{P}(\lambda) = \begin{cases} \frac{\lambda - 18}{30}, & \lambda \in \{19, \ldots, 23\} \\ \frac{29 - \lambda}{30}, & \lambda \in \{24, \ldots, 28\} \\ 0, & \text{else} \end{cases} \tag{40}$$

resulting in average assignment set size $\mu = 23.5 \geq 23.47$. Figure 13 displays the boundary sets that occur as a result of finite sampling effects for the assignment distribution given in (40) when each of the four algorithms in Section 5.3 is used.

8. MODELING EXISTING SCHEMES

In this section, we demonstrate that many existing key predistribution schemes can be modeled and analyzed using the canonical model of seed assignment. The schemes addressed in this section are those that were addressed in Section 2 and are briefly summarized in Table III in terms of the canonical model.

The threshold secret-sharing schemes of Blom [1984] and Blundo et al. [1992] can be described in the canonical model by the seed assignment scheme $(P_{tss}, \mathscr{A}_{tss})$ where the assignment distribution has support $\Lambda_{tss} = \{N\}$. The algorithm $\mathscr{A}_{tss}$ is the trivial algorithm which assigns a matrix row or polynomial share to each node.

The one-way hash function scheme of Leighton and Micali [1993] can similarly be modeled by the seed assignment scheme $(P_{ohf}, \mathscr{A}_{ohf})$ where the assignment distribution has support $\Lambda_{ohf} = \{N\}$. The algorithm $\mathscr{A}_{ohf}$ randomly

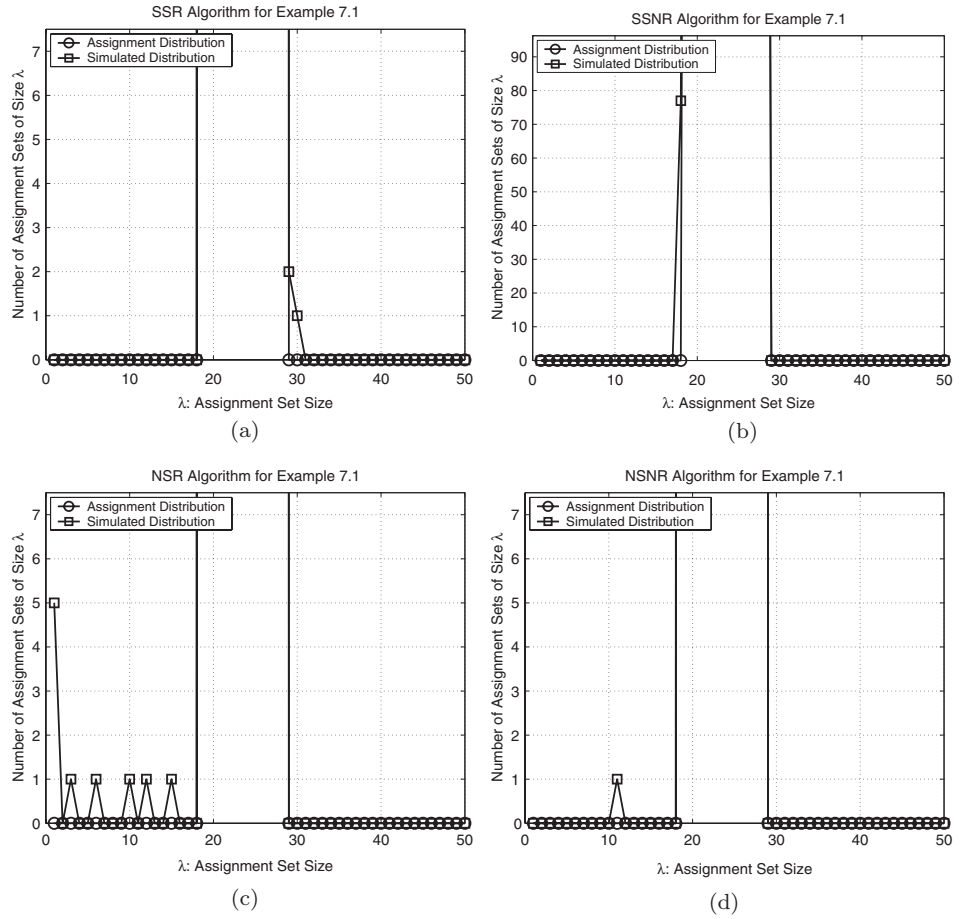


Fig. 13. Occurrence of boundary sets for Example 7.1 using (a) SSR algorithm (b) SSNR algorithm (c) NSR algorithm (d) NSNR algorithm.

generates identifiers α_i and assigns the identifiers and hash values $h^{\alpha_i}(X_i)$ for each node.

The random key predistribution scheme of Eschenauer and Gligor [2002] can be described in the canonical model by the seed assignment scheme $(P_{rkp}, \mathscr{A}_{rkp})$ where P_{rkp} is the binomial distribution given by (4) in Section 3.1. The assignment algorithm $\mathscr{A}_{rkp}$ assigns a random subset of K keys to each node from a total of $P \gg K$ keys. The published result for resilience to node capture [Chan et al. 2003] can be obtained from Theorem 6.9 by noting that the average of a binomial distribution is $\mu = \frac{NK}{P}$.

The schemes of Chan et al. [2003], Ramkumar et al. [2003], Ramkumar and Memon [2004], Di Pietro et al. [2003], Zhu et al. [2003], Du et al. [2003], Liu and Ning [2003], and Liu et al. [2005] can be described by the same seed assignment scheme $(P_{rkp}, \mathscr{A}_{rkp})$ because the modifications to the schemes do not affect the seed assignment. The resilience of these schemes can be computed as a function of the result of Theorem 6.9 using the specifics of each scheme.

Table III. Existing Key Predistribution Schemes are Described in the Canonical Model (A single value for the assignment distribution corresponds to the fixed size of assignment sets resulting from the scheme, and $\mathcal{B}(n, p)$ corresponds to a binomial distribution with parameters n and p)

Reference	Assignment Distribution	Algorithm
[Blom 1984]	N	secret share
[Blundo et al. 1992]	N	secret share
[Leighton and Micali 1993]	N	hashed secret
[Eschenauer and Gligor 2002]	$\mathcal{B}(N, \frac{K}{P})$	subset of seeds
[Ramkumar et al. 2003]	$\mathcal{B}(N, \frac{K}{P})$	subset of seeds
[Chan et al. 2003]	$\mathcal{B}(N, \frac{K}{P})$	subset of seeds
[Chan et al. 2003]	2	seeds to random pairs
[Du et al. 2003]	$\mathcal{B}(N, \frac{K}{P})$	subset of secret shares
[Liu and Ning 2003; Liu et al. 2005]	$\mathcal{B}(N, \frac{K}{P})$	subset of secret shares
[Çamtepe and Yener 2004]	λ	combinatorial design
[Çamtepe and Yener 2004]	$m + \mathcal{B}(N - b - m, \frac{m}{b})$	combinatorial design
[Lee and Stinson 2004]	λ	combinatorial design
[Lee and Stinson 2004]	$rL + 1$	strongly regular graph
[Ramkumar and Memon 2004]	$\mathcal{B}(N, \frac{K}{P})$	subset of seeds
[Chan and Perrig 2005]	$\sqrt[K]{N}$	hyper-grid arrangement
[Liu et al. 2005]	$\sqrt[K]{N}$	hyper-grid arrangement

The random pairwise scheme of Chan et al. [2003] can be described in the canonical model by the seed assignment scheme $(P_{rp}, \mathscr{A}_{rp})$ where the assignment distribution has support $\Lambda_{rp} = \{2\}$. The assignment algorithm $\mathscr{A}_{rp}$ assigns a total of K unique pairwise keys to each node. The perfect resilience of the random pairwise scheme is reflected in the result of Theorem 6.9.

The deterministic seed assignment schemes of Çamtepe and Yener [2004] and Lee and Stinson [2004] based on combinatorial block designs can be described in the canonical model by the seed assignment scheme $(P_{cbd}, \mathscr{A}_{cbd})$ where the assignment distribution has support $\Lambda_{cbd} = \{\lambda\}$. The assignment algorithm $\mathscr{A}_{cbd}$ assigns a common seed to each node in a block of the corresponding combinatorial block design. Such schemes assume that $\frac{N}{L}$ is an integer.

Hybrid symmetric block designs of Çamtepe and Yener [2004] can be described in the canonical model by the seed assignment scheme $(P_{hs}, \mathscr{A}_{hs})$ where P_{hs} is the distribution corresponding to the sum of a constant m and a binomial random variable with parameters $N - b - m$ and $\frac{m}{b}$. This is derived by noting that each seed is contained in exactly m of the b blocks of the combinatorial design and $(b - m)$ of the b blocks of the complementary combinatorial design. The algorithm $\mathscr{A}_{hs}$ assigns a block of the design and a subset of a randomly selected block from the complementary design.

The ID-based one-way scheme of Lee and Stinson [2004] can be described in the canonical model by the seed assignment scheme $(P_{ios}, \mathscr{A}_{ios})$ where the assignment distribution has support $\Lambda_{ios} = \{rL+1\}$. The assignment algorithm $\mathscr{A}_{ios}$ assigns to each node u_i, $i = 1, \ldots, L$ represented by the graph vertex u in the strongly regular graph a key k_u and the values $h(k_v \| ID(u_i))$ for each vertex v adjacent to u in the strongly regular graph. This scheme assumes that $\frac{N}{L}$ is an integer.

Hyper-grid seed assignment schemes such as those of Chan and Perrig [2005] and Liu et al. [2005] can be described in the canonical model by the seed

assignment scheme $(P_{hg}, \mathscr{A}_{hg})$ where the assignment distribution has support $\Lambda_{hg} = \{\sqrt[K]{N}\}$. The assignment algorithm $\mathscr{A}_{hg}$ arranges the N nodes on the grid-points of a K-dimensional hyper-grid and assigns a common seed or share of a common secret to the $\sqrt[K]{N}$ nodes which share a hyper-grid coordinate. Such schemes assume that $\sqrt[K]{N}$ is an integer.

8.1 Analysis of Selected Existing Schemes

In what follows, the worst-case resilience to random node capture is analyzed for selected existing key predistribution schemes using the results of Section 6. The schemes of interest are the random key predistribution scheme of Eschenauer and Gligor [2002], the q-composite scheme of Chan et al. [2003], and the random polynomial-pool scheme of Liu and Ning [2003].

The worst-case resilience is computed in this example for a network of $N = 5,000$ nodes with storage for 100 key-length quantities per node, a radio range of $r = 40m$, a region $\mathcal{A}$ of size $|\mathcal{A}| = 0.5km^2$, and a desired probability of 2-connectivity of $P_G(2) \geq 0.99$. According to Theorem 4.7, the sensor network in each of the following examples will be 2-connected using only secure single-hop links with probability at least 0.99 if the average vertex degree D in the logical graph satisfies $D \geq 1,813$.

8.1.1 *Random Key Predistribution.* Theorem 6.6 with $K = 100$ thus implies that the mean μ of the assignment distribution must satisfy $\mu \geq 23.47$. Due to the binomial assignment distribution, this condition is equivalent to $\frac{NK}{P} \geq 23.47$ or $P \leq \frac{5,000 \times 100}{23.47} = 21,303.79$. Hence, to achieve the desired probability of connectivity with maximum average resilience to node capture, the total number of keys P is chosen as $P = 21,303$.

The average probability of link compromise due to random node capture can then be approximated using Theorem 6.9 as $f(x) = 1 - p(0, x) = 1 - (\frac{4,976.53}{4,998})^x$. The expected worst-case probability of link compromise due to random node capture can be approximated using Lemma 6.8 by computing the value of λ_{max} as given in Section 3.1, yielding $\lambda_{max} = 43$. Hence, the expected worst-case probability of link compromise due to random node capture can be approximated as $f(x, 43) = 1 - p(0, x, 43) = 1 - (\frac{4,957}{4,998})^x$.

The impact of this worst-case analysis can be seen by comparing $f(x)$ and $f(x, 43)$ as given in Figure 14(a).

8.1.2 *q-Composite Scheme.* In this case, a pair of nodes must share at least q keys. The expected node degree D given by Theorem 6.6 can be generalized using a binomial assumption, leading to the result that

$$D = (N-1)\left(1 - \sum_{i=0}^{q-1} \binom{K}{i} \left(\frac{\mu - 1}{N-1}\right)^i \left(\frac{N-\mu}{N-1}\right)^{K-i}\right) \tag{41}$$

for any $q \geq 1$. Letting $q = 3$ for this example, the above result yields the condition $\mu \geq 108.18$ to guarantee the desired probability of connectivity. Due to the binomial assignment distribution, this condition is equivalent to $\frac{NK}{P} \geq 108.18$ or $P \leq \frac{5,000 \times 100}{108.18} = 4,621.92$. Hence, to achieve the desired probability

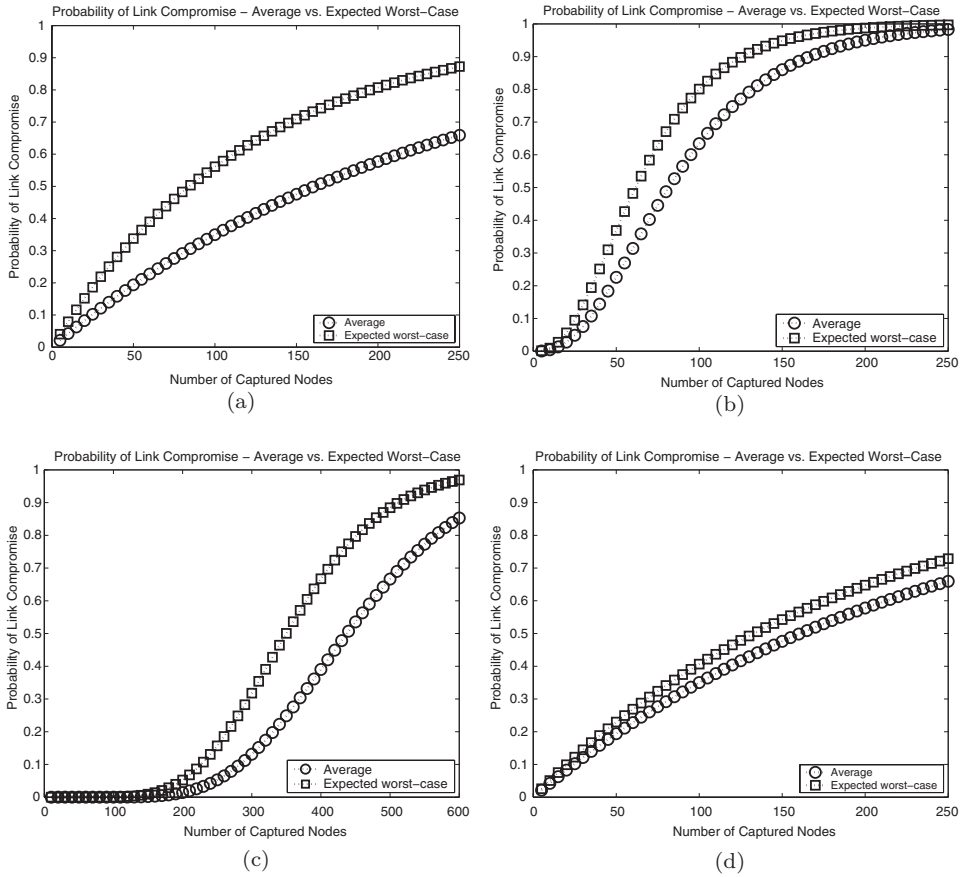


Fig. 14. The probability of link compromise is plotted for average and expected worst case for (a) random key predistribution in Section 8.1.1, (b) q-composite scheme in Section 8.1.2, (c) random polynomial-pool scheme in Section 8.1.3, and (d) Example 7.1 revisited in Section 8.2.

of connectivity with maximum average resilience to node capture, the total number of keys P is chosen as $P = 4,621$.

The average probability of link compromise due to random node capture can then be approximated using Theorem 6.9 and methods of Chan et al. [2003] as

$$f(x) = \sum_{i=q}^{K} (1 - p(0,x))^i \frac{p_s(i)}{p_s(j \geq q)}, \tag{42}$$

where $p_s(j \geq q)$ is the probability that a pair of nodes share at least q keys. Substituting the values of $p_s(i)$ using Theorem 6.3 and inserting all given parameters yields the average probability of link compromise due to random node capture.

The expected worst-case probability of link compromise due to random node capture can be approximated similarly using Lemma 6.8 by computing the value of λ_{max} as given in Section 3.1, yielding $\lambda_{max} = 142$. Hence, the expected worst-case probability of link compromise due to random node capture can be

approximated as $f(x, 142)$ by replacing $1 - p(0, x)$ with $1 - p(0, x, 142)$. Note that the expected worst-case probability $p_s(i, \lambda_1, \ldots, \lambda_K)$ according to Theorem 6.2, where $\lambda_1, \ldots, \lambda_K$ are as given in Section 3.1, can be further substituted for the average probability $p_s(i)$ given by Theorem 6.3.

The impact of this worst-case analysis can be seen by comparing $f(x)$ and $f(x, 142)$ as given in Figure 14(b).

8.1.3 *Random Polynomial-Pool Scheme.* For the polynomial based scheme of Liu and Ning [2003], the value K is equal to the number of $(t - 1)$-degree polynomial shares assigned to each node. The requirement for 100 key-length quantities assigned to each node thus corresponds to the requirement that $tK \leq 100$. Letting $t = 5$ for this example yields $K = 20$.

Theorem 6.6 with $K = 20$ thus implies that the mean μ of the assignment distribution must satisfy $\mu \geq 112.34$. Due to the binomial assignment distribution, this condition is equivalent to $\frac{NK}{P} \geq 112.34$ or $P \leq \frac{5{,}000 \times 20}{112.34} = 890.1$. Hence, to achieve the desired probability of connectivity with maximum average resilience to node capture, the total number of polynomials P is chosen as $P = 890$.

The average probability of link compromise due to random node capture can then be approximated using Theorem 6.9 as

$$f(x) = 1 - \sum_{m=0}^{t-1} p(m, x). \tag{43}$$

The expected worst-case probability of link compromise due to random node capture can be approximated using Lemma 6.8 by computing the value of λ_{max} as given in Section 3.1, yielding $\lambda_{max} = 140$. Hence, the expected worst-case probability of link compromise due to random node capture can be approximated as

$$f(x, 140) = 1 - \sum_{m=0}^{t-1} p(m, x, 140). \tag{44}$$

The impact of this worst-case analysis can be seen by comparing $f(x)$ and $f(x, 140)$ as given in Figure 14(c).

8.2 Analysis of Scheme in Example 7.1

In what follows, the random key predistribution scheme of Eschenauer and Gligor [2002] as analyzed in Section 8.1.1 is compared to that developed in Example 7.1 using the model proposed in this article.

Under the assignment distribution provided in (40), the mean μ of the assignment distribution is given by $\mu = 23.5 \geq 23.47$, thus yielding the desired probability of 2-connectivity. The average probability of link compromise due to random node capture can be approximated using Theorem 6.9 as $f(x) = 1 - p(0, x) = 1 - (\frac{4{,}976.5}{4{,}998})^x$. The expected worst-case probability of link compromise due to random node capture can be approximated using Lemma 6.8

using $\lambda_{max} = 28$ given by (40). Hence, the worst-case probability of link compromise can be approximated as $f(x, 28) = 1 - p(0, x, 28) = 1 - (\frac{4,972}{4,998})^x$.

The impact of this worst-case analysis can be seen by comparing $f(x)$ and $f(x, 28)$ as given in Figure 14(d). The impact of the proposed model can thus be seen by comparing the expected worst-case probability in Figure 14(a) to that in Figure 14(d).

9. DEPLOYMENT OF ADDITIONAL NODES IN THE WSN

In many applications, it may be necessary to deploy additional sensor nodes to replace those that have a depleted energy supply or to increase the coverage of an existing WSN. If the link-key establishment method is such that addition of nodes to the WSN does not require a prohibitive amount of communication overhead, the incorporation of the additional sensor nodes into the secure WSN can be described in terms of the canonical model. However, if a sufficient number of nodes are to be deployed into an existing WSN, the seed assignment scheme for the subsequent deployment might be very different from that of the original deployment. We investigate such scenarios using the canonical model of seed assignment schemes assuming that N nodes have been deployed using the seed assignment scheme $(\mathcal{P}, \mathscr{A})$ and M additional nodes are to be deployed into the existing WSN. We provide a general approach for deployment of additional nodes and give an example which yields a well-known result.

In deploying M additional nodes into an existing WSN, it is highly desirable for the $N + M$ nodes to act as a single secure WSN, so the M additional nodes must be assigned seeds which can be used to establish link-keys with any of the $N + M$ nodes. Furthermore, if M is sufficiently large, a subset of the seeds assigned to the M additional nodes can be fresh. The exact proportion of fresh and existing seeds used in seed assignment for the additional nodes is application dependent, though it is computed as a function of N and M. For simplicity, we assume that a fraction f of the seeds assigned to the M additional nodes are fresh, and the remaining fraction $(1 - f)$ of the seeds are chosen randomly from the set of existing seeds.

The seed assignment scheme $(\mathcal{P}', \mathscr{A}')$ used to assign seeds to the M additional nodes can be designed as a function of the seed assignment scheme $(\mathcal{P}, \mathscr{A})$, the parameters N, M, and f, the total total number of seeds P assigned to the N nodes in the existing WSN, and the total number of seeds P' to be assigned to the M additional nodes. The overall assignment distribution $\mathcal{Q}$ for the network of $N + M$ nodes assigned seeds using assignment distributions $\mathcal{P}$ and $\mathcal{P}'$ is given by the following theorem.

THEOREM 9.1. *Given $N + M$ nodes such that N nodes are assigned a total of P seeds using the assignment distribution $\mathcal{P}$ and M nodes are assigned a total of P' seeds using the assignment distribution $\mathcal{P}'$ where a fraction f of the P' seeds are fresh, the overall assignment distribution $\mathcal{Q}$ is given by*

$$\mathcal{Q}(\lambda) = \frac{P - (1 - f)P'}{P + fP'}\mathcal{P}(\lambda) + \frac{(1 - f)P'}{P + fP'}(\mathcal{P} * \mathcal{P}')(\lambda) + \frac{fP'}{P + fP'}\mathcal{P}'(\lambda),$$

where $\mathcal{P} * \mathcal{P}'$ is the discrete convolution of the assignment distributions $\mathcal{P}$ and $\mathcal{P}'$ given by

$$(\mathcal{P} * \mathcal{P}')(\lambda) = \sum_{\nu \in \Lambda} \mathcal{P}(\nu)\mathcal{P}'(\lambda - \nu).$$

PROOF. The probability that a seed s is assigned only to the M additional nodes is equal to the number of fresh seeds divided by total seeds, $\frac{fP'}{P+fP'}$. The probability that s is assigned only to the N existing nodes is equal to the number of existing seeds divided by total seeds, $\frac{P-(1-f)P'}{P+fP'}$. The probability that s is assigned to both existing and additional nodes is then $\frac{(1-f)P'}{P+fP'}$. The probability that s is assigned to λ_1 existing nodes and λ_2 additional nodes can then be written in terms of $\mathcal{P}$, $\mathcal{P}'$, and $\mathcal{P} * \mathcal{P}'$ using the above probabilities as weights. □

The parameters of the assignment distribution $\mathcal{P}'$ can be chosen in a similar way to the methods described in Section 7 and the relationship between the expected value μ of $\mathcal{P}$, μ' of $\mathcal{P}'$, and γ of $\mathcal{Q}$ given by

$$\gamma = \frac{P-(1-f)P'}{P+fP'}\mu + \frac{(1-f)P'}{P+fP'}\left(\mu+\mu'\right) + \frac{fP'}{P+fP'}\mu' \tag{45}$$

$$= \frac{P}{P+fP'}\mu + \frac{P'}{P+fP'}\mu'. \tag{46}$$

We note that, if $0 < f < 1$, the support of the distribution $\mathcal{Q}$ is necessarily larger than the support of either distribution $\mathcal{P}$ or $\mathcal{P}'$. Hence, the addition of nodes to the network with $0 < f < 1$ causes the assignment distribution to spread. This tends to increase the value of λ_{max} of the assignment distribution $\mathcal{Q}$ compared to that of either $\mathcal{P}$ or $\mathcal{P}'$, thus increasing the worst-case probability estimated by Lemma 6.8.

In both Theorem 9.1 and (46), the number of seeds P' assigned to the M additional nodes may be unknown prior to seed assignment, but it can be approximated by its expected value $P' = \frac{MK}{\mu'}$. We consider the following examples of deployment of additional nodes into a WSN.

Example 9.2. Consider an existing network of N nodes, each of which is assigned K randomly selected seeds from a set of P seeds using the random key predistribution scheme of Eschenauer and Gligor [2002]. The assignment distribution $\mathcal{P}$ is thus given by the binomial distribution $\mathcal{B}(N, \frac{K}{P})$ with mean $\mu = \frac{NK}{P}$. In order to replace depleted nodes and reinforce the WSN, M additional nodes with K seeds per node are to be added to the existing network. To paraphrase Eschenauer and Gligor [2002], since a sufficient number of the K seeds stored in each node are not used to establish link-keys in the existing network, the same set of P seeds can be used in the random key predistribution scheme for the M additional nodes. Hence, $P' = P$ and the distribution $\mathcal{P}'$ is given by the binomial distribution $\mathcal{B}(M, \frac{K}{P})$ with mean $\mu' = \frac{MK}{P}$. Since the same P seeds are used, and the trial probability for the binomial distributions of $\mathcal{P}$ and $\mathcal{P}'$ are the same, this situation is equivalent to deploying a network of $N + M$ nodes with assignment distribution $\mathcal{Q}$ given by the binomial distribution $\mathcal{B}(N + M, \frac{K}{P})$ with mean $\gamma = \mu + \mu' = \frac{(N+M)K}{P}$. This result corresponds to

the result of Theorem 9.1 with the given distributions $\mathcal{P}$ and $\mathcal{P}'$, $P = P'$, and $f = 0$.

The result of Theorem 9.1 is far more general, however, than is illustrated by Example 9.2. The following example demonstrates the generality of Theorem 9.1.

Example 9.3. Similar to Example 9.2, consider an existing network of N nodes, each of which is assigned K randomly selected seeds from a set of P seeds using the random key predistribution scheme of Eschenauer and Gligor [2002] with assignment distribution $\mathcal{P}$ given by the binomial distribution $\mathcal{B}(N, \frac{K}{P})$. The additional M nodes are assigned K seeds each from a total of $P' = P$ seeds, such that a given fraction f of the P' seeds are fresh, and a fraction $(1 - f)$ are randomly selected from the initial set of P seeds. For this example, we assume the assignment distribution $\mathcal{P}'$ is given by the a symmetric peaked distribution similar to that of (40) with $\Lambda = \{\lambda \in \{0, \ldots, N\} : |\lambda - \mu'| \leq 5\}$ where μ' is a given value for the mean of the assignment distribution $\mathcal{P}'$. Hence, the overall assignment distribution $\mathcal{Q}$ corresponding to the $N + M$ nodes is given by Theorem 9.1. The mean of the distribution $\mathcal{Q}$ is given by (46) as

$$\gamma = \frac{P}{P + fP'}\mu + \frac{P'}{P + fP'}\mu' \tag{47}$$

$$= \frac{1}{1 + f}(\mu + \mu'). \tag{48}$$

Since $\frac{1}{2} \leq \frac{1}{1+f} \leq 1$, the maximum value of γ is $\mu + \mu'$, and the minimum value of γ is $\frac{\mu+\mu'}{2}$. Hence, choosing $f = 0$ (as in Example 9.2) maximizes the connectivity of the resulting network, as given by Theorem 4.7, but also maximizes the probability $p(m, x)$ approximated by Theorem 6.9. Since the original network parameters were specified to guarantee network connectivity, the maximum value of γ given by $f = 0$ is a secondary concern to the significant reduction in resilience to node capture. Hence, choosing $f \gg 0$ in this example can maintain the connectivity of the network without sacrificing resilience to node capture. Furthermore, the choice of $\mathcal{P}'$ with support of size at most $|\Lambda| = 11$ yields an overall support set of size at most $N + 11$. If $M > 11$, this choice of $\mathcal{P}'$ results in a significant improvement in the worst-case probability of sharing seeds as given by Theorem 6.2 and the worst-case resilience to node capture given by Lemma 6.7 or Lemma 6.8 compared to the resulting assignment distribution in Example 9.2.

10. CONCLUSION

We proposed a canonical seed assignment model for key predistribution schemes in WSNs based on the probability that a seed is shared by a given number of nodes and the algorithm for assignment of seeds to nodes. We proposed a sampling framework for seed assignment algorithms for use in the canonical model and a model for probabilistic connectivity of secure WSNs restricted by radio range and the existence of shared seeds. We analyzed key predistribution schemes in the canonical model in terms of network connectivity

and resilience to node capture, reflecting the worst-case probabilities for each metric. We demonstrated the design of new key predistribution schemes using the canonical model while paying particular attention to the worst-case seed-sharing probability and resilience to node capture. We have shown that many existing key predistribution schemes can be modeled and analyzed using the canonical model. Finally, we presented an approach to analyze the effect of adding nodes to an existing secure WSN. This approach enables the design of assignment distributions that can tightly match the security requirements of a given application.

REFERENCES

ALON, N. 1991. Probabilistic methods in extremal finite set theory. In *Extremal Problems for Finite Sets*, 39–57.

ANDERSON, R. 2001. *Security Engineering: A Guide to Building Dependable Distributed Systems*. John Wiley & Sons, Inc., New York.

ASADA, G., DONG, M., LIN, T., NEWBERG, F., POTTIE, G., MARCY, H., AND KAISER, W. 1998. Wireless integrated network sensors: Low-power systems on a chip. In *Proceedings of the 24th IEEE European Solid-State Circuits Conference (ESSCIRC'98)*. The Hague, The Netherlands, 9–12.

BETTSTETTER, C. 2002. On the minimum node degree and connectivity of a wireless multihop network. In *Proceedings of the 3rd ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc'02)*. Lausanne, Switzerland, 80–91.

BLOM, R. 1984. An optimal class of symmetric key generation systems. In *Advances in Cryptology: Proceedings of the EUROCRYPT'84*. Paris, France. Lecture Notes in Computer Science, vol. 209, 335–338.

BLUNDO, C., SANTIS, A. D., HERZBERG, A., KUTTEN, S., VACCARO, U., AND YUNG, M. 1992. Perfectly-secure key distribution for dynamic conferences. In *Advances in Cryptology: Proceedings of the CRYPTO'92*. Santa Barbara, CA, Lecture Notes in Computer Science, vol. 740, 471–486.

CARMAN, D. W., KRUUS, P. S., AND MATT, B. J. 2000. Constraints and approaches for distributed sensor network security. Tech. rep. #00-010, NAI Labs.

ÇAMTEPE, S. A. AND YENER, B. 2004. Combinatorial design of key distribution mechanisms for distributed sensor networks. In *Proceedings of the 9th European Symposium on Research in Computer Security (ESORICS'04)*. Sophia Antipolis, France. Lecture Notes in Computer Science, vol. 3193. 293–308.

CHAN, H. AND PERRIG, A. 2005. PIKE: Peer intermediaries for key establishment in sensor networks. In *Proceedings of the 24th Conference of the IEEE Communications Society (INFOCOM'05)*. Miami, FL, 524–535.

CHAN, H., PERRIG, A., AND SONG, D. 2003. Random key predistribution schemes for sensor networks. In *Proceedings of the IEEE Symposium on Security and Privacy*. Oakland, CA, 197–213.

CRESSIE, N. 1993. *Statistics for Spatial Data*. John Wiley and Sons, Inc., New York.

DI PIETRO, R., MANCINI, L. V., AND MEI, A. 2003. Random key assignment for secure wireless sensor networks. In *Proceedings of the ACM Workshop on Security of Ad Hoc and Sensor Networks (SASN'03)*. Fairfax, VA, 62–71.

DU, W., DENG, J., HAN, Y. S., AND VARSHNEY, P. K. 2003. A pairwise key pre-distribution scheme for wireless sensor networks. In *Proceedings of the 10th ACM Conference on Computer and Communications Security (CCS'03)*. Washington, DC, 42–51.

DU, W., WANG, R., AND NING, P. 2005. An efficient scheme for authenticating public keys in sensor networks. In *Proceedings of the 6th ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc'05)*. Urbana-Champaign, IL, 58–67.

DYER, M., FENNER, T., FRIEZE, A., AND THOMASON, A. 1995. On key storage in secure networks. *J. Cryptol. 8*, 189–200.

ERDÖS, P., FRANKL, P., AND FÜREDI, Z. 1982. Families of finite sets in which no set is covered by the union of two others. *J. Combinat. Theory*, Series A *33*, 158–166.

ERDÖS, P., FRANKL, P., AND FÜREDI, Z. 1985. Families of finite sets in which no set is covered by the union of r others. *Israel J. Math. 51*, 1–2.

ESCHENAUER, L. AND GLIGOR, V. D. 2002. A key-management scheme for distributed sensor networks. In *Proceedings of the 9th ACM Conference on Computer and Communications Security (CCS'02)*. Washington, DC, 41–47.

FELLER, W. 1957. *An Introduction to Probability Theory and Its Applications*. vol. 1. John Wiley and Sons, Inc., New York, NY.

GAUBATZ, G., KAPS, J. P., OZTURK, E., AND SUNAR, B. 2005. State of the art in ultra-low power public key cryptography for wireless sensor networks. In *Proceedings of the 3rd IEEE Conference on Pervasive Computing and Communications (PerCom'05)*. Kauai, HI, 146–150.

GONG, L. AND WHEELER, D. J. 1990. A matrix key distribution scheme. *J. Cryptol. 2*, 1, 51–59.

GUPTA, V., MILLARD, M., FUNG, S., ZHU, Y., GURA, N., EBERLE, H., AND SHANTZ, S. C. 2005. Sizzle: A standards-based end-to-end security architecture for the embedded internet. In *Proceedings of the 3rd IEEE Conference on Pervasive Computing and Communications (PerCom'05)*. Kauai, HI, 247–256.

GURA, N., PATEL, A., WANDER, A., EBERLE, H., AND SHANTZ, S. C. 2004. Comparing elliptic curve cryptography and RSA on 8-bit CPUs. In *Proceedings of the 6th Workshop on Cryptographic Hardware and Embedded Systems (CHES'04)*. Cambridge, MA, 119–132.

HILL, J., SZEWCZYK, R., WOO, A., HOLLAR, S., CULLER, D., AND PISTER, K. 2000. System architecture directions for networked sensors. In *Proceedings of the International Conference on Architectural Support for Programming Languages and Operating Systems (ASPLOS-IX)*. Cambridge, MA, 93–104.

JOHNSON, B. R. 1980. An elementary proof of inclusion-exclusion formulas. *Amer. Math. Month. 87*, 9, 750–751.

LEE, J. AND STINSON, D. R. 2004. Deterministic key predistribution schemes for distributed sensor networks. In *Selected Areas in Cryptography (SAC2004)*. Waterloo, Ontario, Canada. Lecture Notes in Computer Science, vol. 3357, 294–307.

LEIGHTON, T. AND MICALI, S. 1993. Secret-key agreement without public-key cryptography. In *Advances in Cryptology: Proceedings of (CRYPTO'93)*. Santa Barbara, CA. Lecture Notes in Computer Science, vol. 773, 456–479.

LIU, D. AND NING, P. 2003. Establishing pairwise keys in distributed sensor networks. In *Proceedings of the 10th ACM Conference on Computer and Communications Security (CCS'03)*. Washington, DC, 52–61.

LIU, D., NING, P., AND LI, R. 2005. Establishing pairwise keys in distributed sensor networks. *ACM Trans. Inform. Syst. Secur. 8*, 1, 41–77.

MITCHELL, C. J. AND PIPER, F. C. 1988. Key storage in secure networks. *Discrete Appl. Math.* 21, 215–228.

PENROSE, M. 1999. On k-connectivity for a geometric random graph. *W. Rand. Struct. Algor. 15*, 2, 145–164.

RAMKUMAR, M. AND MEMON, N. 2004. An efficient random key pre-distribution scheme. In *Proceedings of the IEEE Conference on Global Communications (GLOBECOM'04)*. Dallas, TX, 2218–2223.

RAMKUMAR, M., MEMON, N., AND SIMHA, R. 2003. Pre-loaded key based multicast and broadcast authentication in mobile ad-hoc networks. In *Proceedings of the IEEE Conference on Global Communications (GLOBECOM'03)*. San Francisco, CA, 1405–1409.

STAJANO, F. AND ANDERSON, R. 1999. The resurrecting ducking: Security issues for ad-hoc wireless networks. In *Proceedings of the 7th Annual International Workshop on Security Protocols*. Cambridge, UK, 172–194.

ZHU, S., XU, S., SETIA, S., AND JAJODIA, S. 2003. Establishing pair-wise keys for secure communication in ad hoc networks: A probabilistic approach. In *Proceedings of the 11th IEEE International Conference on Network Protocols (ICNP'03)*. Atlanta, GA, 326–335.

Received December 2005; revised October 2006; accepted January 2007