

Key Pre-Distributions From Graph-Based Block Designs

Jie Ding, Abdelmadjid Bouabdallah, and Vahid Tarokh

Abstract—With the development of wireless communication technologies which considerably contributed to the development of wireless sensor networks (WSNs), we have witnessed ever-increasing WSN-based applications which induced a host of research activities in both academia and industry. Since most of the target WSN applications are very sensitive, security issue is one of the major challenges in the deployment of WSN. One of the important building blocks in securing WSN is key management. Traditional key management solutions developed for other networks are not suitable for WSN, since WSN networks are resource (e.g., memory, computation, and energy) limited. Key pre-distribution algorithms have recently evolved as efficient alternatives of key management in these networks. Secure communication is achieved between a pair of nodes either by the existence of a key allowing for direct communication or by a chain of keys forming a key path between the pair. In this paper, we consider prior knowledge of network characteristics and application constraints in terms of communication needs between sensor nodes, and we propose methods to design key pre-distribution schemes, in order to provide better security and connectivity while requiring less resources. Our methods are based on casting the prior information as a graph. Motivated by this idea, we also propose a class of quasi-symmetric designs referred here to as *g*-designs. Our proposed key pre-distribution schemes significantly improve upon the existing constructions based on the unital designs. We give some examples and point out open problems for future research.

Index Terms—Balanced incomplete block design, graph, key pre-distribution, quasi-symmetric design, sensor networks.

I. INTRODUCTION

WIRELESS sensor networks (WSN) typically consist of a large number of sensor nodes with limited memory, computing, and power. These networks are used in both military and civilian applications. In military applications, sensor nodes may be deployed in battlefield surveillance, while in environmental applications, distributed sensors could monitor physical or environmental conditions such

Manuscript received August 18, 2015; revised November 14, 2015; accepted November 15, 2015. Date of publication November 18, 2015; date of current version February 8, 2016. This work was supported by the Laboratories of Excellence Control of Technological Systems-of-Systems (LABEX MS2T) within the Program Investments for the Future managed by the National Agency for Research through the French Government under Grant ANR-11-IDEX-0004-02. The associate editor coordinating the review of this paper and approving it for publication was Prof. Elena Gaura.

J. Ding and V. Tarokh are with John A. Paulson School of Engineering and Applied Sciences, Harvard University, Cambridge, MA 02138 USA (e-mail: jieding@g.harvard.edu; vahid@seas.harvard.edu).

A. Bouabdallah is with the Heudiasyc Laboratory, Centre National de la Recherche Scientifique, Sorbonne Universités, Université de Technologie de Compiègne, Compiègne 60203, France (e-mail: madjid.bouabdallah@hds.utc.fr).

Digital Object Identifier 10.1109/JSEN.2015.2501429

as temperature, sound, pressure, etc. and cooperatively pass their data through the network to a main location [1], [2]. Because of the sensitivity of most WSN applications, security issue is one of the major challenges in the deployment of WSN.

Security is an essential question for many sensor network applications, especially for military applications. Providing security to small sensor nodes is challenging because of the limited resources of storage, computations, communications, and energy. One of the important building blocks for the development of security solutions for WSN is the key management. The key management scheme design is more complicated due to the characteristics of the WSN such as: (1) The vulnerability of nodes to physical attacks, where the deployment in a hostile area enables the attacker to simply compromise any node and to reveal its security materials (e.g. keys, functions); (2) The nature of wireless communication, where the radio links are insecure and an attacker can eavesdrop on the radio transmissions, inject bits in the channel, and replay previously overheard packets; (3) The density and the large size of the network which make it difficult to control all the nodes.

For security or privacy reasons, it is often critical to build encrypted communications between two sensor nodes using a common secret key. Key pre-distribution scheme (KPS) is a classical way to set up secret keys among sensor nodes before the deployment phase. Compared with online key exchange protocols, key pre-distribution is more attractive for networks consisting of a large number of nodes with limited communication/computation resources [3].

Over the last decade, a host of research on key pre-distribution issue for WSN have been conducted and many solutions have been proposed in literature. Existing KPS fall into two categories: probabilistic and deterministic schemes. In probabilistic schemes, a direct connection between each two nodes is established with certain probability (i.e. probability that these two nodes share a common key). In deterministic schemes, however, each pair of nodes are known to be directly connected or not.

Eschenauer and Gligor [3] proposed a random key pre-distribution (RKP) scheme. Later on, there have been some improvements on RKP, e.g. improvements on its resilience by increasing the “intersection threshold” (the least number of common keys for two nodes to establish a direct connection) [4]. Schemes called “multiple key spaces” are proposed that combine different KPS to achieve better performance [5]–[7]. There are other probabilistic schemes

that use deployment knowledge to improve the performance of sensor networks. Du *et al.* [8] assumed a group-based deployment model where sensor nodes are divided into equal sized groups, and proposed a scheme such that groups farther away from each other share less keys. Later on, Yu and Guan [9] suggested a partition of a sensor field into hexagonal grids to enhance the security; Ito *et al.* [10] proposed an approach that uses a node probability density function to achieve higher connectivity; Liu *et al.* [11] proposed a deployment model that does not require the knowledge of the expected locations of sensors. Martin *et al.* [12] considered logical connections between nodes and proposed a general scheme based on the structure of a resolvable transversal design. We also refer to [13] that utilizes a special type of knowledge about location of nodes to improve the performance, and to [14] for a formalization of KPS from other perspectives.

We note that most existing key pre-distribution solutions that organize sensors in groups or hierarchy do not consider the interactions among them. It may happen that two nodes which are in the same group or cluster do not need to communicate while the developed key pre-distribution protocol assigns them a shared key. On the contrary, a node of a cluster C_i may need to frequently communicate with another node which is assigned to a different cluster C_j , but they don't share a key because they are not in the same cluster. Thus, there is a loss of efficiency in the above cases.

In the category of deterministic schemes, the simplest way is to assign a distinct key to each link, and $b - 1$ pairwise keys to each node, where b is the number of nodes. Choi *et al.* [15] proposed an improved method where each node only needs to store $(b + 1)/2$ keys. However, these methods suffer from scalability problems. Several deterministic key management solutions that have been developed in the literature are based on combinatorial design. Indeed, Çamtepe and Yener [16], [17] proposed a novel method that uses block design for key pre-distribution. They proposed a deterministic key pre-distribution scheme that maps a symmetric balanced incomplete block design (SBIBD) or generalized quadrangles (GQ) to key pre-distribution.

There are some other works that use the design theory to construct effective KPS. Lee and Stinson [18] introduced the common intersection designs. Chakrabarti *et al.* [19] used transversal designs and merging block techniques. Ruj and Roy [20] proposed a KPS that is based on partially balanced incomplete block designs (PBIBD). Later on, Bose *et al.* [21] proposed an improved construction that combines several PBIBDs. Bechkit *et al.* [22] proposed the unital-based key pre-distribution scheme (UKP), a deterministic scheme that improves the scalability of a network while maintaining a good key sharing probability. The appropriateness of combinatorial designs as a tool for KPS has been studied in [23]. More detailed surveys can be found in [23]–[27].

A. Motivations for Graph-Based KPS

Usually, the main goal of KPS is to develop a design solution that provides more connectivity coverage while requiring less memory (or using as few keys as possible). Existing designs are often evaluated under criteria such as

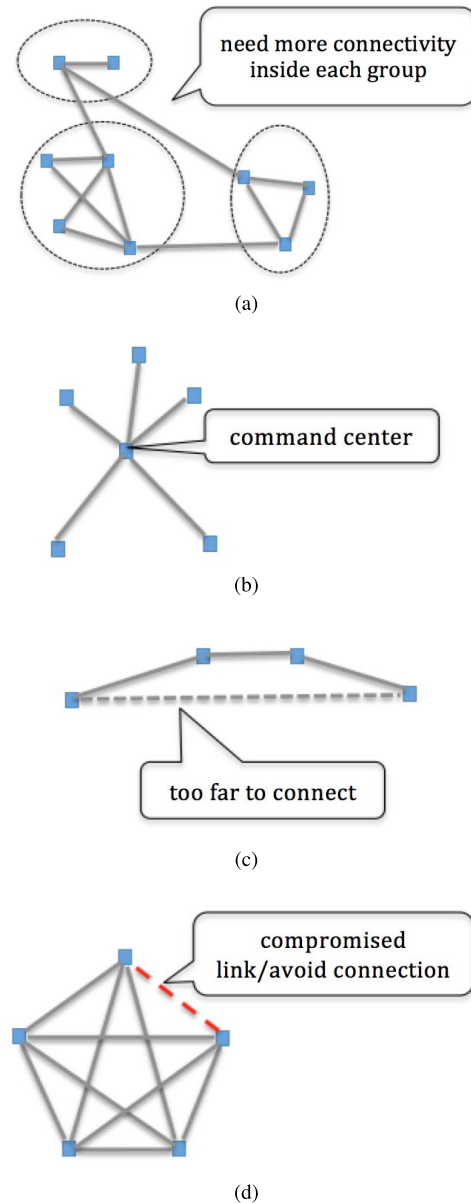


Fig. 1. Observations. (a) A hierarchical sensor network. (b) A master-slave sensor network. (c) Sensors beyond the radio frequency range. (d) Links susceptible to eavesdropping.

network connectivity, average path length, network resiliency, storage overhead, and network scalability. We will review them in Section III.

Our contributions are motivated by the following observations:

- **Observation 1:** In many applications, there are several intended deployment locations, and typically a number of sensor nodes are deployed in each of these locations. In hierarchical scenarios, each group of sensors placed in a location must pass their data to sensor nodes of higher ranks or levels. This means that for a hierarchical sensor network, nodes in the same group need more connectivity than those across groups (Figure 1a). This scheme can be used to reduce the number of required keys.
- **Observation 2:** In some scenarios, a group of sensor nodes are naturally set in a master-slave architecture.

For instance, the commander of a military needs more communications with his lower rank staff. This means that one or some nodes are in charge of collecting data or sending command signals to the remaining nodes (Figure 1b). So there are some important connections which we need to establish with higher efficiency and security.

- **Observation 3:** Two sensor nodes can communicate with each other only in a certain distance referred to as the radio frequency (RF) range (Figure 1c). If it is known that certain pair of nodes never connect directly because they are outside of each other's radio frequency range, then it would be a waste of resources to assign common keys to them.
- **Observation 4:** If it is known in advance that certain connections are more likely to be eavesdropped, the corresponding nodes should not share common keys (Figure 1d).

It is very important to consider all these observations in order to improve the efficiency of our key pre-distribution solution. This motivates us to formalize a framework that models the prior knowledge as graphs and apply them to the design of KPS while taking into consideration all these observations.

B. Contribution and Organization of This Work

We propose an efficient graph-based key pre-distribution (graph-based block design) solution that incorporate prior knowledge of network characteristics and application requirements. This provides better security and connectivity while requiring less resources once properly used. We elaborate on two practical scenarios, and explain why the graph-based design is preferable or even required. Some previous work that used deployment knowledge to optimize KPS may look similar, but is different in concept. For example, the scope of "deployment" was usually in a geographic sense. However, the model that communication should not pass through some links of potential danger was rarely studied, to the best of our knowledge.

We first briefly review the basics of block design theory in Section II. Especially, we propose g -designs, a class of the quasi-symmetric designs, and initiate the study of their applications to KPS. We propose the concept of graph-based KPS in Section III. In Section IV, we demonstrate the improvements provided by graph-based KPS and g -designs in a specific scenario. In Section V, we study another scenario. We further provide an algorithmic framework (called MAR) for KPS design for the second scenario in Section VI. Finally, we give our conclusions in Section VII.

II. BACKGROUND

A. Block Design Theory

Block design theory deals with the properties, existence, and construction of systems of finite sets whose intersections have specified numerical properties. A block design (BD) is a system of sets (V, \mathfrak{B}) . Each element in V is called a point (or treatment), and each element in \mathfrak{B} is called a block.

Definition 1: A **Balanced Incomplete Block Design (BIBD)** with parameters λ , k , r , v , and b is a block design in which v points are arranged in b blocks, such that each block contains k points, each point appears in r blocks, and each pair of points appear in exactly λ blocks. It is denoted by (λ, k, r, v, b) -BIBD. It may also be denoted by (λ, k, v) -BIBD, as r and b are given by [28]

$$r = \frac{\lambda(v-1)}{k-1}, \quad b = \frac{\lambda(v-1)v}{(k-1)k}. \quad (1)$$

Definition 2: A BIBD is a g -design if any two blocks intersect in either zero or a fixed number of points $g > 0$.

A quasi-symmetric design is a BIBD with two possible intersection numbers for any pair of blocks. A g -design clearly belongs to the class of quasi-symmetric designs when one intersection number equals to zero. But the term g -design is defined here for convenience. Some properties and constructions of g -designs have been studied in [29]–[33].

Clearly, any (λ, k, r, v, b) -BIBD with $\lambda = 1$ (also referred to as a Steiner system) is a g -design with $g = 1$.

Also, any unital design is a $(1, m+1, m^2, m^3+1, m^2(m^2-m+1))$ -BIBD for some m , and is also a g -design.

Definition 3: Let G be a graph. Let $V(G)$ and $E(G)$ be respectively the set of nodes and edges of G . G is **regular** with **degree** d if every node of G has incidence degree d . A **clique** in G is a subset of its vertices such that every two vertices in the subset are connected by an edge; in other words, it is a subgraph of G and it is complete.

Definition 4: A **strongly regular graph (SRG)** with parameter set (b, d, t, u) is defined as a regular graph of size b and degree d , such that every two adjacent nodes have t common neighbors, and every two non-adjacent nodes have u common neighbors. The SRG is denoted by $\text{srg}(b, d, t, u)$.

B. Some Properties of g -Designs

The following results are helpful for the future analysis.

Theorem 1: If there exists a (λ, k, v) -BIBD which is also a g -design with $g = 2$, then there exists a regular graph G of size b such that its edge set $E(G)$ is a disjoint union of $v(v-1)/2$ subsets, where each subset forms a clique of size λ . Also, b satisfies the equation

$$k(k-1) = \frac{\lambda v(v-1)}{b}. \quad (2)$$

Proof: The proof is given in Appendix A, as it uses a definition to be introduced in Section III. \square

Theorem 1 provides a necessary condition for the $g = 2$ case. The following result gives the equivalence between g -designs with $g = 1$ and a class of graphs.

Theorem 2: The existence of a (λ, k, v) -BIBD when $\lambda = 1$ is equivalent to the existence of a regular graph G of size b such that its edge set $E(G)$ is a disjoint union of v subsets each of which forms a clique of size r , where

$$\frac{rv}{b} = \frac{v-1}{r} + 1, \quad (3)$$

and $rv/b \in \mathbb{N}$ (the set of positive integers).

Proof: The proof is given in Appendix B, as it uses a technique to be introduced in Section VI. \square

Lemma 1 ([34], Lemma 2.1): If a (λ, k, v) -BIBD is a g -design, then its design graph is a strongly regular graph, denoted by $\text{srg}(b, d, t, u)$. When $\lambda = 1$, we have

$$d = \frac{v-k}{k-1}k, \quad t = \frac{v-1}{k-1} - 2 + (k-1)^2, \quad u = k^2. \quad (4)$$

III. GRAPH-BASED KPS AND EVALUATION METRICS

Block designs are intimately related to key pre-distribution schemes. In a KPS system designed for a WSN, to each sensor node is assigned a set of keys, called “key ring”. We let a key correspond to a point, and a key ring correspond to a block. For example, A unital design-based KPS gives $m^2(m^2 - m + 1)$ key rings from a key pool of $m^3 + 1$ keys, such that each key ring contains $m + 1$ keys.

In sensor network applications, if two key rings share at least one common key, the corresponding two nodes can be directly connected to each other, i.e. can communicate directly in a secure way since they share at least one common key. The direct connections form a graph, whose nodes correspond to the sensor nodes, and whose edges correspond to direct connections. This graph is referred to as the “design graph”. In mathematical terms, we use the following definitions and terminology:

Definition 5: A **design graph** for a specific block design is a graph G_D whose nodes $V(G_D)$ correspond to the blocks. Two nodes are connected in G_D if and only if the corresponding two blocks share at least one point.

The prior structural information of a network can also be modeled as a graph called “target graph”:

Definition 6: A **target graph** for a specific WSN is a triplet of graphs $G_T = (G_T^c, G_T^u, G_T^r)$ that satisfies

- 1) each node of G_T^c , G_T^u , or G_T^r corresponds to a node in the WSN,
- 2) two nodes are connected in G_T^c if and only if the corresponding nodes in the WSN must directly communicate,
- 3) two nodes are connected in G_T^u if and only if the corresponding nodes in the WSN are required not to directly communicate, and
- 4) for any pair of nodes not covered by the above two cases, they are not connected in G_T^r if and only if the corresponding nodes in the WSN may communicate via a path but not necessarily communicate directly.

We note that design graphs and target graphs are undirected and unweighted. Also in the classical case considered in all existing research work of the literature, G_T^c, G_T^u are null graphs (denoted by \emptyset) and G_T^r is a complete graph, i.e. $G_T = (\emptyset, \emptyset, G_{\text{complete}})$ where G_{complete} denotes the complete graph.

Given a WSN, we assume that G_T is the target graph which models the available prior information. A natural question to be asked is how to use classical performance metrics in the context of our key pre-distribution designs (Observations 1-4, Section I-A) and the graph G_T . To answer this question, we first briefly review the classical performance metrics in terms of a design graph G_D , and then we define new metrics according to a target graph G_T .

The classical performance metrics known in KPS designs are the following:

- **Direct connectivity coverage** is the fraction of the direct links to all the possible links in the network, i.e. the probability that a given pair of nodes can be directly connected, i.e. share at least one common key.
- **Average path length** is the expectation of the length of the shortest path between two nodes drawn uniformly from the network. It can be calculated as the average length of the shortest paths between pairs of nodes in G_D . It is defined to be infinity (∞) if there exist two nodes that cannot establish a connection path.
- **Network resiliency** NR_x measures the fraction of the uncompromised external links when x sensor nodes are captured. It can be calculated as the fraction of the edges that do not contain keys employed in the key pools of the compromised nodes.
- **Storage overhead** measures the memory required to store the keys in each node, often calculated as the size of each block.
- **Network scalability** is the total number of keys needed, for a given number of nodes.

In what follows, we define the new performance metrics:

- **Direct connectivity coverage (DCC) and average path length (APL)**

If two nodes do not communicate (or are not connected in $G_T^c \cup G_T^r$), whether they share keys or not should not be considered into the evaluation of a KPS (Observation 1 and Observation 3). We therefore restrict the calculations of the two metrics to the edge set $E(G_D) \cap E(G_T^c \cup G_T^r)$, e.g. only consider the edges in G_D that also appear in $G_T^c \cup G_T^r$.

- **Direct important connectivity coverage (DICCC)**

Direct important connectivity coverage can be calculated as the direct connectivity coverage restricted to G_T^c , i.e. $|E(G_T^c) \cap E(G_D)| / |E(G_T^c)|$ with $|\cdot|$ representing the cardinality of a set. This metric is meaningful only when G_T^c is not empty.

- **Network resiliency (NR)**

Observation 4 provides a scenario where certain nodes are required not to communicate, which is represented by the edges of G_T^u of the target graph G_T . In terms of the metric of network resiliency, if two compromised nodes are connected in both G_D and G_T^u , the common keys they share are regarded as being captured. Thus, NR_x can be calculated as the fraction of the edges that do not contain keys that are employed by the edges in $E(G_D) \cap E(G_T^u)$ or the x compromised nodes. It reduces to the classical case when G_T^u is a null graph.

Definition 7: A KPS is **graph-based** if it is designed based on the target graph. Its performance is evaluated based on the above metrics.

Since it is not easy to provide a universal design that is suitable for any situation, we focus on the following two different scenarios of graph-based KPS.

Scenario 1: In this scenario, we consider the case where $G_T = (\emptyset, \emptyset, G_T^r)$, i.e. every two nodes may or may not communicate. In this scenario, we need to use the new

performance metrics “direct connectivity coverage” and “average path length”.

Scenario 2: In this scenario, we consider the case $G_T = (G_T^c, G_T^u, \emptyset)$, i.e. $G_T^c \cup G_T^u$ is a complete graph, and two nodes either must communicate or are required not to communicate. In this scenario, we need to use the new performance metrics “network resiliency” and “direct important connectivity coverage”.

IV. THE $G_T = (\emptyset, \emptyset, G_T^r)$ SCENARIO

In this scenario, G_T^r contains the information about pairs of nodes that do not need direct connections. If G_T^r is non-trivial, i.e. G_T^r is not a complete graph, we may improve the performance by employing the extra information provided by G_T^r .

For comparison, we first consider the trivial case in which G_T^r is a complete graph.

(1) When G_T^r is a complete graph, Bechkit *et al.* [22] propose a highly scalable KPS using the unital design. This was shown to outperform other KPS in many aspects. Here, we examine a more general case. We use a (λ, k, r, v, b) -BIBD with $\lambda = 1$ for KPS design, and evaluate the KPS performance as follows (in terms of v and k).

- *Direct connectivity coverage / Direct important connectivity coverage:*

$$\begin{aligned} DCC &= \frac{bd/2}{b(b-1)/2} = \frac{d}{b-1} = \frac{\frac{v-k}{k-1}k}{\frac{v(v-1)}{k(k-1)} - 1} \\ &= \frac{(v-k)k^2}{v(v-1) - k(k-1)}. \end{aligned} \quad (5)$$

- *Average path length:* If two nodes are not connected, there are $u > 1$ nodes connecting both of them, so the minimum path length between these two nodes is equal to two. The average path length is thus

$$APL = DCC + 2(1 - DCC) = 2 - DCC. \quad (6)$$

- *Network resiliency:* For approximate analysis, we assume that the captured nodes are uniformly distributed among all the nodes. Since each key occurs in r blocks among the total number of b blocks, the probability that a key is not compromised when x nodes are captured is $\binom{b-r}{x} / \binom{b}{x}$, where $b = \frac{v(v-1)}{k(k-1)}$. Further, the probability that a given link is not compromised is

$$NR_x = \frac{\binom{b-r}{x}}{\binom{b}{x}}. \quad (7)$$

- *Storage overhead:* It is the size of each block, i.e. $SO = k$.
- *Network scalability:* It is the total number of keys, i.e. $NS = v$.

(2) When G_T^r is not a complete graph, in order to exploit the graph information, let us consider a network in which there are s groups and each of the group contains b_0 sensor nodes. For each group, there are τ_0 ($0 \leq \tau_0 \leq b_0/s$) “central nodes” (or nodes of higher rank) who are responsible for collecting information from all the other nodes in the same group.

Besides this, between any two groups only the central nodes could establish connections; in other words, a “non-central node” can only communicate with the nodes within the same group. In terms of a target graph, G_T^r is isomorphic to the following matrix $[J_{mn}]_{sb_0 \times sb_0}$ (two graphs are isomorphic if their vertices are the same up to a relabeling):

$$J_{mn} = 1, \quad \forall 0 \leq (m \bmod b_0), (n \bmod b_0) < \tau_0; \quad (8)$$

$$J_{mn} = 1, \quad \forall \left\lfloor \frac{m}{b_0} \right\rfloor = \left\lfloor \frac{n}{b_0} \right\rfloor; \quad (9)$$

$$J_{mn} = 0, \quad \text{otherwise.} \quad (10)$$

Here, $\lfloor x \rfloor$ denotes the largest integer that is no more than the real number x . Equations (8) and (9) represent the possible connections among all the central nodes and among the nodes in each group, respectively.

Our goal is to design a KPS such that any (central or non-central) node could transfer its information to any other node in the network, while satisfying the required performance metrics. A possible graph-based KPS design is to assign keys, e.g. via BIBD with $\lambda = 1$, for each of the g groups, together with the group of all central nodes. We now evaluate the network performance of this graph-based design, and compare it with the classical way.

Let v_0 and $(s+1)v_0$ be respectively the size of the key pool for each group of the graph-based KPS and for the classical KPS. We define $b = sb_0$, $v = (s+1)v_0$, and then we compute the following performance metrics:

- *Direct connectivity coverage / Direct important connectivity coverage:*

For classical KPS, from $b = \frac{v(v-1)}{k(k-1)}$ we obtain $k = O(\sqrt{\frac{v(v-1)}{b}}) = O(vb^{-\frac{1}{2}})$,¹ and from Equation (5) we further obtain

$$DCC = O\left(\frac{vk^2}{v^2}\right) = O(vb^{-1}) = O(v_0b_0^{-1}). \quad (11)$$

For approximate analysis, we assume that the edges of the unital design are uniformly distributed in $E(G_T^r)$ and $E'(G_T^r)$ (the complement of $E(G_T^r)$). This implies that

$$\frac{|E(G_T^r) \cap E(G_D)|}{|E(G_T^r)|} = \frac{|E(G_D)|}{\binom{b}{2}}.$$

For graph-based KPS, the direct connectivity coverage within one group is $O(v_0b_0^{-1})$, and within the central nodes is $O(v_0(s\tau_0)^{-1}) \geq O(v_0b_0^{-1})$ using the same reasoning. So the overall DCC is at least

$$DCC_G = O(v_0b_0^{-1}), \quad (12)$$

which is as good as Equation (11).

- *Average path length:* The given target graph requires that any path is composed of two types of connections: normal node to central node, and central node to central node. For classical KPS, within a group or among central nodes, the average path length is less or equal to

$$APL = DCC + 2(1 - DCC) = 2 - DCC. \quad (13)$$

¹ $O(\cdot)$ Notation: $f = O(g)$ means there exists a positive constant c such that $c^{-1}g < f < cg$.

For graph-based KPS, connections within any group form a SRG, so the average path length is

$$APL_G = DCC_G + 2(1 - DCC_G) \approx APL. \quad (14)$$

- *Network resiliency*: We assume that the captured nodes are uniformly distributed among all the nodes. For classical KPS, each key occurs in r blocks (from the total number of b blocks), and thus the probability that a key is not compromised when x nodes are captured is

$$NR_x = \frac{\binom{b-r}{x}}{\binom{b}{x}} = \frac{\binom{b-O(b^{\frac{1}{2}})}{x}}{\binom{b}{x}}, \quad (15)$$

where we have applied $r = \frac{v-1}{k-1} = O(vk^{-1}) = O(b^{\frac{1}{2}})$. This is also the probability that a given link is not compromised.

For graph-based KPS, each key occurs in $r_0 = O(b_0^{\frac{1}{2}})$ blocks (from the total number of b blocks), and thus the probability that a key is not compromised when x nodes are captured is $\binom{b-r_0}{x}/\binom{b}{x}$. Therefore, the probability that a given link within any group is not compromised is

$$NR_{Gx} = \frac{\binom{b-r_0}{x}}{\binom{b}{x}} = \frac{\binom{b-s^{-\frac{1}{2}}O(b^{\frac{1}{2}})}{x}}{\binom{b}{x}}, \quad (16)$$

which is greater than that in (15). Furthermore, the resiliency for the central nodes is greater than or equal to NR_{Gx} because $s\tau_0 \leq b_0$. As a result, the resiliency is improved.

- *Storage overhead*: For classical KPS, storage overhead is given by

$$SO = k = O(vb^{-\frac{1}{2}}) = s^{\frac{1}{2}}O(v_0b_0^{-\frac{1}{2}}). \quad (17)$$

For graph-based KPS, this is equal to $k_0 = O(v_0b_0^{-\frac{1}{2}})$ for a normal node, and $2k_0$ for a central node, so in average:

$$SO_G = O(v_0b_0^{-\frac{1}{2}}) = s^{-\frac{1}{2}}SO. \quad (18)$$

In summary, under the same network scalability, the direct connectivity coverage and average path length of graph-based KPS are no worse than those of the classical ones, while the network resiliency and storage overhead are comparatively improved. Thus, the overall performance is improved.

V. THE $G_T = (G_T^c, G_T^u, \emptyset)$ SCENARIO

In this scenario, we want to have a KPS whose design graph is exactly G_T^c . We first consider the case where G_T^c is (by coincidence) the design graph of certain g -design with $g = 2$ and parameters (λ, k, r, v, b) . We start with such simple and “ideal” case for two main reasons: first, it provides a benchmark for a more general-purposed approach to be introduced in the next section; second, designs for more complex target graphs may be derived based upon the ideal ones.

We immediately obtain a satisfying KPS by the natural mapping between blocks of the g -design and key rings. We refer to it as the “natural” method.

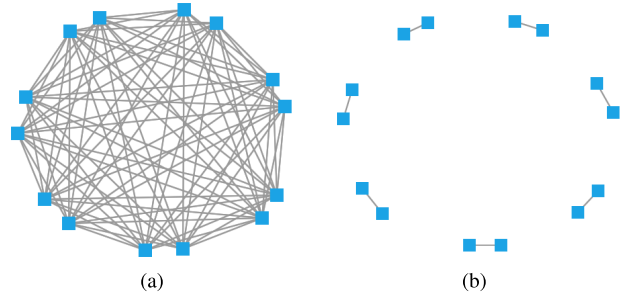


Fig. 2. Graph G_T^c and its complement. (a) G_T^c . (b) The complement graph of G_T^c .

Clearly, the storage overhead is k , and the scalability is v . As for the network resiliency, if one node (block) is captured, there will be $\binom{\lambda}{2}\binom{k}{2}$ compromised connections. To observe this, we first notice that any pair of points in the block appear in λ blocks, and there are $\binom{\lambda}{2}$ such pairs, so $\binom{\lambda}{2}\binom{k}{2}$ connections are compromised. Moreover, any other connection is secure, since the corresponding two blocks share at least one key that is not captured.

Example 1: Let G_T^c be the graph shown in Figure 2(a). As we can see, the nodes of G_T^c could be grouped into seven disconnected pairs with all other edges connected in the graph. For clarity reasons, we present in Figure 2(b) the complement of G_T^c . Suppose that we would like to construct a KPS whose design graph is G_T^c .

We construct the following g -design with $g = 2$ and parameters $(\lambda, k, r, v, b) = (3, 4, 7, 8, 14)$:

$$\begin{aligned} V &= \{a_1a_2a_3a_4a_5a_6a_7a_8\}, & (19) \\ \mathfrak{B} &= \{\{a_1a_2a_3a_4\}, \{a_1a_2a_5a_6\}, \{a_1a_2a_7a_8\}, \{a_1a_2a_6a_8\}, \\ &\quad \{a_1a_2a_5a_7\}, \{a_1a_4a_5a_8\}, \{a_1a_4a_6a_7\}, \\ &\quad \{a_5a_6a_7a_8\}, \{a_3a_4a_7a_8\}, \{a_3a_4a_5a_6\}, \{a_2a_4a_5a_7\}, \\ &\quad \{a_2a_4a_6a_8\}, \{a_2a_3a_6a_7\}, \{a_2a_3a_5a_8\}\}. & (20) \end{aligned}$$

Obviously (V, \mathfrak{B}) forms a g -design ($g = 2$) whose design graph is G_T^c . In other words, we have obtained a satisfying KPS with a_i ($i = 1, \dots, 8$) representing the keys.

Next we evaluate the performance of this construction by computing resiliency, storage overhead, and scalability measures.

We consider the simple case when one node is captured, say $V_1 = \{a_1a_2a_3a_4\}$. In this case, there are 18 connections compromised. To observe this, we first notice that any two points of V_1 appear in exactly 3 blocks; thus, there are $\binom{3}{2}\binom{4}{2} = 18$ compromised connections in total. Besides this, if both points of the intersection of two blocks do not belong to V_1 , the two blocks are able to communicate in a secure way.

The storage overhead is the size of each block, i.e. $k = 4$. The network scalability is the total number of points, i.e. $v = 8$.

Now, the question to be asked is: for any given target graph $G_T = (G_T^c, G_T^u, \emptyset)$, does there exist a KPS whose design graph G_D is exactly G_T^c ? In fact, we have a positive answer that is guaranteed by the following algorithm.

TABLE I
MATCHING AND REDUCING ALGORITHM

Input:	$G_T^c = [V(G_T^c), E(G_T^c)]$, integer $c_0 > 2$. Let $b = V(G_T) $, $e = E(G_T) $.
Initialization:	$l = 0$, $G^l = G_T^c$. Generate e different keys $K = \{k_1, \dots, k_e\}$. Assign K to $E(G^0)$, s.t. each edge is assigned a unique key. Define $\mathfrak{B} = \{B_1, \dots, B_b\}$, where $B_n = \{k_t \mid k_t \text{ is assigned to an edge which is incident to node } n\}$.
Repeat (Clique reduction procedure)	$l = l + 1$; Find a clique C^l in G^{l-1} whose size is no larger than c_0 ; Denote $K_{C^{l-1}}$ as the set of keys assigned to $E(C^l)$; Update the blocks: $B_m = B_m - K_{C^{l-1}}, \forall m \in V(C^l)$; Arbitrarily choose a key k^l from $K_{C^{l-1}}$; $B_m = B_m \cup \{k^l\}, \forall m \in V(C^l)$; Update from G^{l-1} to G^l : $E(G^l) = E(G^{l-1}) - E(C^l)$;
Until	no clique of size greater than 2 can be found.
Output:	$\mathfrak{B}, V = \cup_{m=1}^b B_m$.

VI. MATCHING AND REDUCING ALGORITHM (MAR)

A schematic diagram of the Matching and Reducing Algorithm (MAR) is depicted in Table I.

In the initialization step, a unique key is assigned to each edge, i.e. two nodes of that edge contain the key. Thus, the key ring of each node has been determined. However, our objective is to reduce the size of the key pool and key rings. Notice that for any clique C in G_T^c , it will not change the design graph if the distinct keys assigned to $E(C)$ are replaced by only one key. In other words, if all the nodes of C share a common key, they still directly connect with one another, and edges outside C are not affected. Therefore, MAR looks for cliques in the current target graph, reduces the number of keys within each clique to one, and then updates the target graph by removing the clique. The size of the clique, however, is upper-bounded by a constant integer $c_0 > 2$ to ensure a reasonable network resiliency. Consider, for example, if G_T^c is a complete graph, one key is enough for full connectivity; however, the the whole network is compromised as long as any one node is captured.

An illustrating example is given in Figure 3. We elaborate on the other aspects of this algorithm:

- 1) We did not give the details of how to detect cliques, or the optimality criteria for clique selection. Indeed, MAR encompasses many more algorithms. For example, one may propose a specific algorithm that deviates from MAR, in order to minimize the total number of keys ($\min |V|$), etc.
- 2) It is interesting that MAR is applied as a technical tool to the proof of Theorem 2 (in Appendix B).
- 3) The upper bound of the clique size in MAR is designed to ensure network resiliency NR_x . But is there any theoretical lower bound of NR_x if the clique size is bounded by c_0 ? The following theorem gives a positive answer.

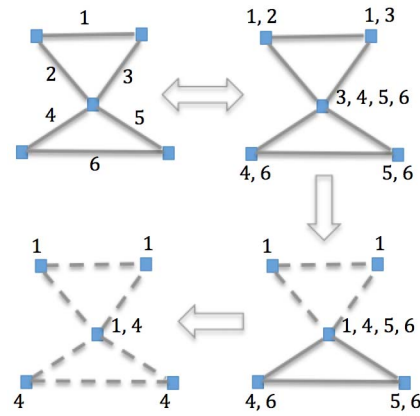


Fig. 3. An illustrating example where MAR with $c_0 = 3$ is applied to a simple graph.

A. Network Resiliency for MAR

Theorem 3: For a given graph G_T^c of degree d (i.e. the degree of any node is no larger than d), the network resiliency NR_x of the KPS determined by the Matching and Reducing Algorithm satisfies

$$NR_x \geq 1 - \frac{x \binom{c_0}{2} \lceil \frac{d}{c_0-1} \rceil}{|E(G_T^c)|}, \quad (21)$$

where $\lceil d/(c_0-1) \rceil$ denotes the smallest integer that is no less than $d/(c_0-1)$.

Proof: Let us consider an arbitrary node x . Each key that is assigned to x is associated with one clique in G_T^c , due to the clique reduction procedure of the MAR algorithm (an edge can be regarded as a clique of size 2). Assume that x is associated with M cliques, and that clique m is of size λ_m , $m = 1, \dots, M$. Let $y_m = \lambda_m - 1$. It is clear that

$$\sum_{m=1}^M y_m \leq d, \quad 1 \leq y_m \leq c_0 - 1 \quad (c_0 \text{ is given in Table I}). \quad (22)$$

Moreover, the number of compromised links when node x is captured is: $F(y_1, \dots, y_m) = \sum_{m=1}^M \binom{y_m+1}{2}$. We now evaluate the maximum for F under constraint (22).

Let $f(y) = \binom{y+1}{2}$. Given positive numbers a, b, s such that $s - a > 0$ and $s - b > 0$, it is easy to observe that $f(a) + f(s-a) < f(b) + f(s-b)$ if and only if $|a-s/2| < |b-s/2|$.

Given two positive variables y_1 and y_2 satisfying $y_1 + y_2 \leq c_0 - 1$, we have

$$\left| y_1 - \frac{y_1 + y_2}{2} \right| < \left| (y_1 + y_2) - \frac{y_1 + y_2}{2} \right|.$$

Thus,

$$F(y_1, y_2, \dots, y_m) < F(y_1 + y_2, \dots, y_m). \quad (23)$$

Moreover, given two positive variables y_1 and y_2 satisfying $y_1 + y_2 > c_0 - 1$, $y_1 \leq y_2 < c_0 - 1$, we have

$$\left| y_2 - \frac{y_1 + y_2}{2} \right| < \left| c_0 - 1 - \frac{y_1 + y_2}{2} \right|.$$

We conclude that

$$F(y_1, y_2, \dots, y_m) < F(c_0 - 1, y_1 + y_2 - (c_0 - 1), \dots, y_m).$$

By continuous application of (23) and the above inequality, F is maximized when $y_1 = \dots = y_{M-1} = c_0 - 1$, $y_M = d - (M - 1)(c_0 - 1)$, and $M = \lceil d/(c_0 - 1) \rceil$. Thus,

$$F(\cdot) \leq \binom{c_0}{2} M = \binom{c_0}{2} \left\lceil \frac{d}{c_0 - 1} \right\rceil = F_0. \quad (24)$$

If x nodes are captured, the worst case is that they do not share keys and $x F_0$ connections are compromised, which implies the result in (21). \square

B. Example

In this section, we revisit the special case discussed in Section V, i.e. G_T^c in $G_T = (G_T^c, G_T^u, \emptyset)$ is the design graph of a g -design with $g = 2$ and parameters (λ, k, r, v, b) . Now we apply the Matching and Reducing Algorithm and choose the parameter c_0 to be λ , so that the network resiliency is not worse than the “natural” method (Section V). Here are the reasons:

- Due to Theorem 1, the edge set of G_T^c is a disjoint union of $\binom{v}{2}$ subsets, each of which forms a clique of size λ . Further, when applying MAR with $c_0 = \lambda$, it is clear that the minimal number of keys is obtained when the C^l in each step is one of the $\binom{v}{2}$ cliques. Therefore, there are $\binom{v}{2}$ keys in total, and the number of keys required by each node is

$$\frac{d}{\lambda - 1} = \frac{2|E(G_T^c)|}{b} \frac{1}{\lambda - 1} = \frac{2\binom{v}{2}\binom{\lambda}{2}}{b} \frac{1}{\lambda - 1} = \binom{k}{2},$$

where the last equality is due to Equation (2).

- If one node is captured, there will be $\binom{k}{2}\binom{\lambda}{2}$ connections compromised, which is the same as the “natural” method.
- Finally, if c_0 is chosen to be larger, the network resiliency obviously decreases.

Let us go back to Example 1. By choosing the parameter c_0 to be 3, there are $\binom{v}{2} = 28$ keys in total, and each node requires $\binom{k}{2} = 6$ keys. The keys/key rings can be realized as:

$$\begin{aligned} \mathfrak{B} = & \{\{12, 13, 14, 23, 24, 34\}, \{12, 15, 16, 25, 26, 56\}, \\ & \{12, 17, 18, 27, 28, 78\}, \{13, 16, 18, 36, 38, 68\}, \\ & \{13, 15, 17, 35, 37, 57\}, \{14, 15, 18, 45, 48, 58\}, \\ & \{14, 16, 17, 46, 47, 67\}, \{56, 57, 58, 67, 68, 78\}, \\ & \{34, 37, 38, 47, 48, 78\}, \{34, 35, 36, 45, 46, 56\}, \\ & \{24, 25, 27, 45, 47, 57\}, \{24, 26, 28, 46, 48, 68\}, \\ & \{23, 26, 27, 36, 37, 67\}, \{23, 25, 28, 35, 38, 58\}\}. \end{aligned}$$

Here, each key is uniquely denoted by a two-digit integer. For example, 12 represents a key, and 13 represents another key.

Clearly, if one node is captured, $6 \cdot \binom{c_0}{2} = 18$ connections are compromised, which is the same as the “natural” approach.

Moreover, if c_0 is chosen to be 2, the network resiliency is improved. This is because every connection is secured by a unique key, and if one node is captured, only $d = 12 < 18$ connections are compromised. However, the storage overhead increases and network scalability decreases.

If c_0 is chosen to be 4, the network resiliency decreases. To observe this, consider the following case:

Label the the 14 nodes to be n_1, \dots, n_{14} , and let two nodes n_i, n_j be disconnected if and only if $|i - j| = 7$ (Figure 2). If we apply MAR with $c_0 = 4$, then it can be assumed that the following four cliques of size 4 appear in the “clique reduction procedure” (by possible relabeling):

$$\begin{aligned} & \{n_1, n_2, n_3, n_4\}, \quad \{n_1, n_5, n_6, n_7\}, \\ & \{n_1, n_9, n_{10}, n_{11}\}, \quad \{n_1, n_{12}, n_{13}, n_{14}\}. \end{aligned}$$

This means that if node n_1 is captured, the four keys, along with the $4 \cdot \binom{4}{2} = 24 > 18$ connections among the above four cliques, are compromised.

In summary, the Matching and Reducing Algorithm provides a general solution for KPS design given an arbitrary target graph. Nevertheless, the previous example reveals that for specific target graphs (G_T^c), there is a potential advantage of using g -designs ($g > 1$) based KPS in terms of storage overhead and scalability. We believe that g -design is a promising design tool for KPS and leave that for future work here.

VII. CONCLUSION

We proposed a new approach for key pre-distribution in WSN which takes into consideration the application needs in terms of communication motivated by several realistic observations. We redefined the classical performance metrics in order to evaluate our graph-based key pre-distribution scheme. We introduced the g -designs, studied some of their connections with graph theory, and applied them to KPS constructions. Two specific target graphs were considered. Especially, we introduced an algorithm framework called the Matching and Reducing Algorithm. Examples were provided to demonstrate the performance of the proposed schemes.

APPENDIX A

PROOF OF THEOREM 1

Consider a g -design (V, \mathfrak{B}) with $g = 2$ and design graph G . Equation (1) implies that G has $b = \frac{\lambda v(v-1)}{k(k-1)}$ nodes. Because two connected blocks share exactly $g = 2$ keys, all edges in G are induced by a pair of keys in V . Besides this, any pair of keys induces a clique of size λ in G . Two cliques do not share an edge, otherwise there exists two blocks that share at least three keys. Finally, every block intersect with other $\frac{k(k-1)}{2}(\lambda - 1)$ blocks, because each block contains $\frac{k(k-1)}{2}$ pairs and each pair belongs to $\lambda - 1$ other blocks. This implies that G is regular.

APPENDIX B

PROOF OF THEOREM 2

Sufficiency: Suppose that there exists a $(\lambda = 1, k, v)$ -BIBD. Let G be its design graph. Any point in V induces r nodes that are connected to one another, forming a clique of size r . Besides this, any two cliques induced by two different points do not share an edge, because otherwise these two points appear in two different blocks contradicting $\lambda = 1$. Finally, every block intersects with other $(r - 1)k$ blocks, because each block contains k points and each point belongs to $r - 1$ other blocks. This implies that G is regular.

Necessity: Let $k = rv/b = (v - 1)/r + 1$. Applying MAR to the given graph G for v iterations, we obtain (V, \mathfrak{B}) together

with an empty graph G^l . Due to MAR, $|V| = v$, each key appears in r blocks, and each block in \mathfrak{B} contains

$$\frac{d}{r-1} = \frac{2e}{b} \frac{1}{r-1} = 2 \frac{vr(r-1)}{2} \frac{1}{b(r-1)} = k$$

keys. Next we prove that any pair of keys appear in exactly one block. To that end, we let b_{ij} be the number of blocks which keys i and j both belong to, and count the value of $B = \sum_{1 \leq i, j \leq v} b_{ij}$ in two different ways: on one hand, because each block contains k points and there are b blocks, B can be calculated as

$$B = \frac{k(k-1)}{2} b = \frac{k(k-1)}{2} \frac{v(v-1)}{k(k-1)} = \frac{v(v-1)}{2}.$$

On the other hand, any pair of keys appear in no more than one block, because G is decomposed into v cliques any two of which share no edges. Moreover, B is no more than the total number of pairs $v(v-1)/2$. Therefore, any pair must exactly appear in one block.

REFERENCES

- [1] J. Yick, B. Mukherjee, and D. Ghosal, "Wireless sensor network survey," *Comput. Netw.*, vol. 52, no. 12, pp. 2292–2330, 2008.
- [2] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "Wireless sensor networks: A survey," *Comput. Netw.*, vol. 38, no. 4, pp. 393–422, 2002.
- [3] L. Eschenauer and V. D. Gligor, "A key-management scheme for distributed sensor networks," in *Proc. 9th ACM Conf. Comput. Commun. Secur.*, 2002, pp. 41–47.
- [4] H. Chan, A. Perrig, and D. Song, "Random key predistribution schemes for sensor networks," in *Proc. Symp. Secur. Privacy*, May 2003, pp. 197–213.
- [5] W. Du, J. Deng, Y. S. Han, P. K. Varshney, J. Katz, and A. Khalili, "A pairwise key predistribution scheme for wireless sensor networks," *ACM Trans. Inf. Syst. Secur.*, vol. 8, no. 2, pp. 228–258, 2005.
- [6] J. Lee and D. R. Stinson, "Deterministic key predistribution schemes for distributed sensor networks," in *Selected Areas in Cryptography*. New York, NY, USA: Springer, 2005, pp. 294–307.
- [7] D. Liu, P. Ning, and R. Li, "Establishing pairwise keys in distributed sensor networks," *ACM Trans. Inf. Syst. Secur.*, vol. 8, no. 1, pp. 41–77, 2005.
- [8] W. Du, J. Deng, Y. S. Han, S. Chen, and P. K. Varshney, "A key management scheme for wireless sensor networks using deployment knowledge," in *Proc. 23rd Annu. Joint Conf. IEEE Comput. Commun. Soc. (INFOCOM)*, vol. 1, Mar. 2004, pp. 1–12.
- [9] Z. Yu and Y. Guan, "A robust group-based key management scheme for wireless sensor networks," in *Proc. IEEE Wireless Commun. Netw. Conf.*, vol. 4, Mar. 2005, pp. 1915–1920.
- [10] T. Ito, H. Ohta, N. Matsuda, and T. Yoneda, "A key pre-distribution scheme for secure sensor networks using probability density function of node deployment," in *Proc. 3rd ACM Workshop Secur. Ad Hoc Sensor Netw.*, 2005, pp. 69–75.
- [11] D. Liu, P. Ning, and W. Du, "Group-based key predistribution for wireless sensor networks," *ACM Trans. Sensor Netw.*, vol. 4, no. 2, p. 11, 2008.
- [12] K. M. Martin, M. B. Paterson, and D. R. Stinson, "Key predistribution for homogeneous wireless sensor networks with group deployment of nodes," *ACM Trans. Sensor Netw.*, vol. 7, no. 2, p. 11, 2010.
- [13] S. R. Blackburn, T. Etzion, K. M. Martin, and M. B. Paterson, "Distinct difference configurations: Multihop paths and key predistribution in sensor networks," *IEEE Trans. Inf. Theory*, vol. 56, no. 8, pp. 3961–3972, Aug. 2010.
- [14] M. Kendall, K. M. Martin, S.-L. Ng, M. B. Paterson, and D. R. Stinson, "Broadcast-enhanced key predistribution schemes," *ACM Trans. Sensor Netw.*, vol. 11, no. 1, p. 6, 2014.
- [15] T. Choi, H. B. Acharya, and M. G. Gouda, "The best keying protocol for sensor networks," *Pervasive Mobile Comput.*, vol. 9, no. 4, pp. 564–571, 2013.
- [16] S. A. Çamtepe and B. Yener, "Combinatorial design of key distribution mechanisms for wireless sensor networks," in *Computer Security—ESORICS*. New York, NY, USA: Springer, 2004, pp. 293–308.
- [17] S. A. Çamtepe and B. Yener, "Combinatorial design of key distribution mechanisms for wireless sensor networks," *IEEE/ACM Trans. Netw.*, vol. 15, no. 2, pp. 346–358, Apr. 2007.
- [18] J. Lee and D. R. Stinson, "A combinatorial approach to key predistribution for distributed sensor networks," in *Proc. IEEE Wireless Commun. Netw. Conf.*, vol. 2, 2005, pp. 1200–1205.
- [19] D. Chakrabarti, S. Maitra, and B. Roy, "A key pre-distribution scheme for wireless sensor networks: Merging blocks in combinatorial design," *Int. J. Inf. Secur.*, vol. 5, no. 2, pp. 105–114, 2006.
- [20] S. Ruj and B. Roy, "Key predistribution using partially balanced designs in wireless sensor networks," in *Parallel and Distributed Processing and Applications*. New York, NY, USA: Springer, 2007, pp. 431–445.
- [21] M. Bose, A. Dey, and R. Mukerjee, "Key predistribution schemes for distributed sensor networks via block designs," *Designs, Codes Cryptogr.*, vol. 67, no. 1, pp. 111–136, 2013.
- [22] W. Bechkit, Y. Challal, A. Bouabdallah, and V. Tarokh, "A highly scalable key pre-distribution scheme for wireless sensor networks," *IEEE Trans. Wireless Commun.*, vol. 12, no. 2, pp. 948–959, Feb. 2013.
- [23] K. M. Martin, "On the applicability of combinatorial designs to key predistribution for wireless sensor networks," in *Coding and Cryptology*. New York, NY, USA: Springer, 2009, pp. 124–145.
- [24] J. Hwang and Y. Kim, "Revisiting random key pre-distribution schemes for wireless sensor networks," in *Proc. 2nd ACM Workshop Secur. Ad Hoc Sensor Netw.*, 2004, pp. 43–52.
- [25] S. A. Çamtepe and B. Yener, "Key distribution mechanisms for wireless sensor networks: A survey," Dept. Comput. Sci., Rensselaer Polytech. Inst., Troy, NY, USA, Tech. Rep. TR-05-07, 2005.
- [26] R. Wei and J. Wu, "Product construction of key distribution schemes for sensor networks," in *Selected Areas in Cryptography*. New York, NY, USA: Springer, 2005, pp. 280–293.
- [27] J. Lee and D. R. Stinson, "On the construction of practical key predistribution schemes for distributed sensor networks using combinatorial designs," *ACM Trans. Inf. Syst. Secur.*, vol. 11, no. 2, p. 1, 2008.
- [28] A. Dey, *Theory of Block Designs*. New York, NY, USA: Wiley, 1986.
- [29] M. Shrikhande, "A survey of some problems in combinatorial designs—A matrix approach," *Linear Algebra Appl.*, vol. 79, pp. 215–247, Jul. 1986.
- [30] S. S. Sane and M. S. Shrikhande, "Finiteness questions in quasi-symmetric designs," *J. Combinat. Theory A*, vol. 42, no. 2, pp. 252–258, 1986.
- [31] V. C. Mavron and M. Shrikhande, "On designs with intersection numbers 0 and 2," *Arch. Math.*, vol. 52, no. 4, pp. 407–412, 1989.
- [32] M. S. Shrikhande and S. S. Sane, *Quasi-Symmetric Designs*. Cambridge, U.K.: Cambridge Univ. Press, 1991, no. 164.
- [33] C. J. Colbourn and J. H. Dinitz, Eds., *Handbook of Combinatorial Designs*. Boca Raton, FL, USA: CRC Press, 2010.
- [34] S. S. Sane and M. S. Shrikhande, "Quasi-symmetric 2, 3, 4-designs," *Combinatorica*, vol. 7, no. 3, pp. 291–301, 1987.

Jie Ding is currently pursuing the Ph.D. degree with the School of Engineering and Applied Sciences, Harvard University. His current research interests include statistical inference, machine learning, sequence design, and information criteria.

Abdelmadjid Bouabdallah received the Engineering degree from the University of Science and Technology Houari Boumediene Algeria, and the master's and Ph.D. degrees from the University of Paris-Sud, Orsay, France, in 1988 and 1991, respectively. From 1992 to 1996, he was an Assistant Professor with the University of Evry-Val-d'Essonne, France, and he has been a Professor with the University of Technology of Compiegne, since 1996, where he is leading the Networking and Security Research Group and the Interaction and Cooperation Research of the Excellence Research Center LABEX MS2T. He conducted several large-scale research projects founded by well known companies (Motorola Laboratories, Orange Laboratories, CEA, etc.) and academy (ANR-RNRT, CNRS, and ANR-Carnot). His research interest includes Internet QoS, security, unicast/multicast communication, wireless sensor networks, and fault tolerance in wired/wireless networks.

Vahid Tarokh is currently a Professor of Applied Mathematics with the School of Engineering and Applied Sciences, Harvard University. His current research interests are data analysis, network security, optical surveillance, and radar theory.