

Combinatorial Design of Key Distribution Mechanisms for Wireless Sensor Networks

Seyit A. Çamtepe¹, Bülent Yener¹

Technical Report 04-10

Department of Computer Science, Rensselaer Polytechnic Institute
Troy, NY 12180, USA

{camtes,yener}@cs.rpi.edu

April 12, 2004

Abstract. Key distribution is one of the most challenging security issues in wireless sensor networks where sensor nodes are randomly scattered over a hostile territory. In such a sensor deployment scenario, there will be no prior knowledge of post deployment configuration. For security solutions requiring pairwise keys, it is impossible to decide how to distribute key pairs to sensor nodes before the deployment. Existing approaches to this problem are to assign more than one key, namely a key-chain, to each node. Key-chains are randomly drawn from a key-pool. Either two neighboring nodes have a key in common in their key-chains, or there is a path, called key-path, among these two nodes where each pair of neighboring nodes on this path has a key in common. Problem in such a solution is to decide on the key-chain size and key-pool size so that every pair of nodes can establish a session key directly or through a path with high probability. The size of the key-path is the key factor for the efficiency of the design. This paper presents novel, *deterministic* and *hybrid* approaches based on *Combinatorial Design* for key distribution. In particular, several *block design* techniques are considered for generating the key-chains and the key-pools.

Comparison to probabilistic schemes shows that our combinatorial approach produces better connectivity with smaller key-chain sizes.

1 Introduction and Problem Definition

In this work, we consider a sensor network in which sensor nodes need to communicate with each other for data processing and routing. We assume that the sensor nodes are distributed to the target area in large numbers and their location within this area is determined randomly. These type of sensor networks are typically deployed in adversarial environments such as military applications where large number of sensors may be dropped from airplanes.

In this application, secure communication among sensor nodes requires authentication, privacy and integrity. In order to establish this, there must be a *secret key* shared between a pair of communicating sensor nodes. Because the network topology is unknown prior to deployment, a key pre-distribution scheme

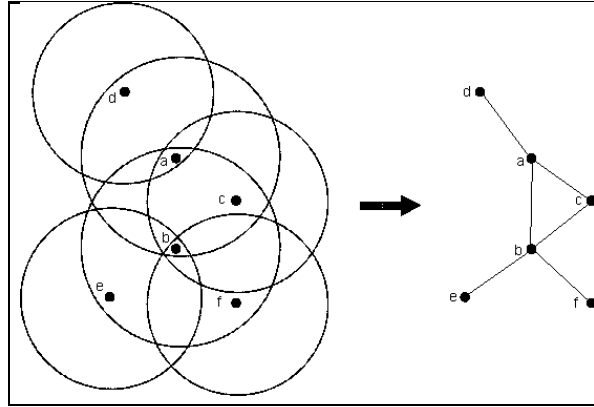


Fig. 1. A Wireless Sensor Network

is required where keys are stored into ROMs of sensors before the deployment. The keys stored must be carefully selected so to increase the probability that two neighboring sensor nodes have at least one key in common. Nodes that do not share a key directly may use a path where each pair of nodes on the path shares a key. The length of this path is called *key-path* length. Average *key-path* length, is an important performance metric and design consideration. Consider sample sensor network given in Figure-1. Assume that only sensor nodes *a* and *b* don't share a key. Nodes *a* and *c* can establish secure communication where the key-path length is one. Node *c* and *b* also have key-path length of one. However, nodes *a* and *b* can only use the path *a-c-b* to communicate securely with key-path length of two.

The common approach is to assign each sensor node multiple keys, *randomly* drawn from a *key-pool*, to construct a *key-chain* to ensure that either two neighboring nodes have a key in common in their key-chain, or there is a key-path. Thus the challenge is to decide on the key-chain size and key-pool size so that every pair of nodes can establish a session key directly or through a path. Key-chain size is limited by the storage capacity of the sensor nodes. Moreover, very small key-pool increases the probability of key share between any pair of sensor nodes by decreasing the security in that, the number of the keys needed to be discovered by the adversary decreases. Similarly, very large key-pool decreases the probability of key share by increasing the security.

Eschenauer *et al.* in [13] propose a *random key pre-distribution scheme* where tens to hundreds of keys are uploaded to sensors before the deployment. In their solution, initially a large key pool of P and their identities are generated. For each sensor, k keys are randomly drawn from the key-pool P without replacement. These k keys and their identities form a *key-chain* which is loaded to the memory of the sensor node. Two neighboring nodes compare the list of identities of keys in their key-chains. Since only the identities are exchanged, this process can take

place without any privacy mechanism. Eschenauer *et al.* also propose to employ *Merkle Puzzle* [19] similar approach to secure key identities which requires too much processing and storage for a resource limited sensor node. After key identity exchange, common keys are used to secure the link in between two sensor nodes. It may be the case that some of the neighboring nodes may not be able to find a key in common. These nodes may communicate securely through other nodes, through other secured links. Eschenauer *et al.* shows that, for the key-pool size of 10,000 keys, only 75 keys are needed in key-chains so that probability of key share between any two nodes is 0.5. Later in results section, we will show that our deterministic algorithms provides better probability of key share than *random key pre-distribution scheme* for the same key-chains size. Chan *et al.* in [5] propose a modification to the basic scheme of Eschenauer *et al.* They increase the amount of key overlap required for key-setup. That is, q common keys are needed instead of one to be able to increase the security of the communication between two neighboring nodes. Their proposal requires larger key-chains and smaller key-pools than the original proposal of Eschenauer *et al.* In [32], common keys in the key-chains are used to establish multiple logical paths over which costly *threshold key sharing scheme* is used to agree on a new secret.

Random-pairwise key scheme in [5] is a modification of the pairwise key scheme. It is based on *Erdoes and Renyi's* work; to achieve probability p of any two nodes are connected, in a network of n nodes, each node needs to store only a random set of np pairwise keys instead of $n - 1$. This scheme provides perfect resilience since each one of the pairwise keys are distinct, but it can not support large networks because key-chain size is linearly dependent to network size. Slijepcevic *et al.* in [23] propose that each sensor node shares a list of master keys, a random function and a seed. Every sensor uses shared random function and shared seed to select a network-wise or group-wise master key. In [3, 17], polynomial-based key pre-distribution protocol proposed for group key pre-distribution. In [18], polynomial pool-based key pre-distribution is used for pairwise key establishment. For each sensor, random or a grid based pre-distribution scheme is used to select set of polynomials from a pool.

In [2], Blom proposes a k -secure key pre-distribution system where a public Vandermonde matrix P and a private symmetric matrix S over a finite field $GF(q)$ is used. The idea is to provide each node relatively small secret and public data from which they can derive a key for their neighbors. Thus, rows of the matrix $A = (S.P)^T$ and columns of matrix P are distributed to the nodes (i.e. node i stores row_i of matrix A and col_i of matrix P). Since matrix P is Vandermonde matrix, it is sufficient to store column generator instead of whole column. Symmetric matrix $K = A.P$ is the key matrix. A pair of nodes, say nodes i and j , first exchanges their col_i and col_j , then multiplies row_i of matrix A and col_j of P to calculate key $K_{i,j} = K_{j,i}$ (partial matrix multiplication $K = A.P$). Blom's scheme is a deterministic scheme where any pair of nodes can calculate common secret key, that is, probability of key share and key path length are both one. Blom's scheme stores small amount data to nodes but requires nodes to calculate the keys whenever required. That is, Blom's scheme requires

resource limited sensor nodes to perform costly multiplication operations on large numbers. Blom’s scheme can resist capture of at most k nodes, credentials stored in $k + 1$ nodes are enough to recover all the keys used in the network. Increase in k (size of the stored row of matrix A) increases the resilience but it means more storage and more computation for a sensor node. For the same key-chain size k , our symmetric block design algorithm provides the same probability of key share and better resilience with the same storage requirements, but without any costly multiplication operations. Du *et al.* in [8] uses Blom’s scheme with ω spaces to increase resilience. They use more than one key space, that is ω of the matrix S and generate $A_\delta = (S_\delta.P)^T$ for $1 \leq \delta \leq \omega$. Each node is randomly assigned rows from τ spaces out of ω spaces. It may be the case that two neighboring nodes do not share a key space. Therefore, as τ/ω ratio decreases, probability of key share decreases rapidly. Unlike Du *et al.*, we use smaller key chains and generalized quadrangles block design to improve the resilience. In [9], Du *et al.* first model a node deployment knowledge in a wireless sensor network and then develop a key pre-distribution scheme based on this model.

In [22, 29, 10, 11, 6, 4, 33] a network architecture where there are one or more base-stations is considered. These base-stations are considered as powerful in resource and sensor nodes are clustered around them. Each sensor node shares a key with each base-station to secure sensor node to base-station and base-station to sensor node unicast communication. Authentication mechanism for the broadcasts from base-station to sensor nodes is addressed in [22, 10, 11, 16, 6]. They propose modified versions of *TESLA* where a verifiable key, which is used to encrypt a message, is disclosed later than the message was broadcasted.

1.1 Our Contributions and Organization of this Work

The main contribution of this work is the *deterministic* and *hybrid* approaches to the key distribution problem. In particular, we bring in a novel construction methodology from *Combinatorial Design Theory* to address this problem. Although there are some applications of Combinatorial Designs in cryptography [25–27], and in network design [31, 28], best to our knowledge this work is the first to apply design theory to key distribution. Our analysis indicates that deterministic approach has strong advantages over the randomized one since it (i) increases the probability that two nodes share a key, and (ii) decreases the key-path length.

This paper is organized as follows: In Section 2 we provide a brief background to the combinatorial designs used in this work without exceeding the scope of this paper. In Section 3 we introduce our key distribution construction and explain the mapping from design theory to this practical problem. In Section 4 we address scalability issues. In Section 5, we present our analysis and comparison with randomized methods. Finally, in Section 6 we conclude.

2 Background on Combinatorial Designs

A *Balanced Incomplete Block Design (BIBD)* is an arrangement of v distinct objects into b blocks such that each block contains exactly k distinct objects, each object occurs in exactly r different blocks, and every pair of distinct objects occurs together in exactly λ blocks. The design can be expressed as (v, k, λ) , or equivalently (v, b, r, k, λ) , where: $\lambda(v-1) = r(k-1)$ and $bk = vr$.

2.1 Symmetric BIBD

A *BIBD* is called *Symmetric BIBD* or *Symmetric Design* when $b = v$ and therefore $r = k$ [7, 1, 14, 30]. A *Symmetric Design* has four properties: every block contains $k = r$ elements, every element occurs in $r = k$ blocks, every pair of elements occurs in λ blocks and every pair of blocks intersects in λ elements.

Example 1: Consider $(v, k, \lambda) = (7, 3, 1)$, or equivalently $(v, b, r, k, \lambda) = (7, 7, 3, 3, 1)$, *Symmetric Design*. Let $S = \{1, 2, 3, 4, 5, 6, 7\}$ be the set of $|S| = v = 7$ objects. There are $b = 7$ blocks and each block contains $k = 3$ objects. Every object occurs in $r = 3$ blocks. Every pair of distinct objects occurs in $\lambda = 1$ blocks and every pair of blocks intersects in $\lambda = 1$ objects. The blocks of the *Symmetric Design* are:

$$\{1, 2, 3\} \{1, 4, 5\} \{1, 6, 7\} \{2, 4, 6\} \{2, 5, 7\} \{3, 4, 7\} \{3, 5, 6\} .$$

In this paper, we are interested in a subset of Symmetric Designs, called a *Finite Projective Plane*. A *Finite Projective Plane* consists of a finite set P of points and a set of subsets of P , called lines. For an integer n where $n \geq 2$, *Finite Projective Plane* of order n has four properties: (i) every line contains exactly $n+1$ points, (ii) every point occurs on exactly $n+1$ lines, (iii) there are exactly n^2+n+1 points, and (iv) there are exactly n^2+n+1 lines. If we consider lines as blocks and points as objects, then a *Finite Projective Plane* of order n is a *Symmetric Design* with parameters $(n^2+n+1, n+1, 1)$ [7, 1].

Given a block design $D = (v, k, \lambda)$ with a set S of $|S| = v$ objects and $B = \{B_1, B_2, \dots, B_b\}$ of $|B| = b$ blocks where each block includes exactly k objects, *Complementary Design* \overline{D} has the complement blocks $\overline{B}_i = S - B_i$ as its blocks for $1 \leq i \leq b$. \overline{D} is a block design with parameters $(v, b, b-r, v-k, b-2r+\lambda)$ where $(b-2r+\lambda > 0)$ [1, Theorem 1.1.6]. If $D = (v, k, \lambda)$ is a *Symmetric Design*, then $\overline{D} = (v, v-k, v-2r+\lambda)$ is also a *Symmetric Design* [1, Corollary 1.1.7].

Example 2: Consider Symmetric Design $D = (v, k, \lambda) = (7, 3, 1)$ of *Example-1*. Complementary Design of this design is $\overline{D} = (v, v-k, b-2r+\lambda) = (7, 4, 2)$. Given the same set S of $|S| = v = 7$ objects, there are $b = 7$ blocks and each block contains $v-k = 4$ objects. Every object occurs in $b-r = 4$ blocks. Every pair of distinct objects occurs in $b-2r+\lambda = 2$ blocks and every pair of blocks intersects in $b-2r+\lambda = 2$ objects. The blocks of the *Complementary Design* are:

$$\{4, 5, 6, 7\} \{2, 3, 6, 7\} \{2, 3, 4, 5\} \{1, 3, 5, 7\} \{1, 3, 4, 6\} \{1, 2, 5, 6\} \{1, 2, 4, 7\} .$$

2.2 Finite Generalized Quadrangle

A *Finite Generalized Quadrangle (GQ)* is an incidence structure $S = (P, B, I)$ where P and B are disjoint and nonempty sets of points and lines respectively, and for which I is a symmetric point-line incidence relation satisfying the following axiom:

1. Each point is incident with $t + 1$ lines ($t \geq 1$) and two distinct points are incident with at most one line,
2. Each line is incident with $s + 1$ points ($s \geq 1$) and two distinct lines are incident with at most one point,
3. If x is a point and L is a line not incident (I) with x , then there is a unique pair $(y, M) \in PXB$ for which $x I M I y I L$.

In this work, we are interested in three known GQ's as defined in [20, 12, 15, 21]: two GQs are from the *Projective Space* $PG(4, q)$ and $PG(5, q)$ of order q , third one is from $PG(4, q^2)$ of order q^2 . Let function f be an *irreducible binary quadratic*, then the three GQs can be defined as follows:

1. $GQ(s, t) = GQ(q, q)$ from $PG(4, q)$ with canonical equation $x_0^2 + x_1x_2 + x_3x_4 = 0$:
 $GQ(q, q) \Rightarrow s = t = q, v = b = (q + 1)(q^2 + 1)$.
2. $GQ(s, t) = GQ(q, q^2)$ from $PG(5, q)$ with canonical equation $f(x_0, x_1) + x_2x_3 + x_4x_5 = 0$:
 $GQ(q, q^2) \Rightarrow s = q, t = q^2, v = (q + 1)(q^3 + 1), b = (q^2 + 1)(q^3 + 1)$.
3. $GQ(s, t) = GQ(q^2, q^3)$ from $PG(4, q^2)$ with canonical equation $x_0^{q+1} + x_1^{q+1} + \dots + x_d^{q+1} = 0$:
 $GQ(q^2, q^3) \Rightarrow s = q^2, t = q^3, v = (q^2 + 1)(q^5 + 1), b = (q^3 + 1)(q^5 + 1)$.

Consider $GQ(s, t) = GQ(q, q)$ in which lines are mapped to blocks and points to objects. Thus, there are $v = b = (q + 1)(q^2 + 1)$ blocks and objects where each block contains $s + 1 = q + 1$ objects, and where each object is contained in $t + 1 = q + 1$ blocks.

Example 3: Consider $GQ(s, t) = GQ(2, 2)$ for $q = 2$. There are $v = b = 15$ blocks and objects where each block contains $s + 1 = 3$ objects, and where each object is contained in $t + 1 = 3$ blocks. Assume the set of objects $S = \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14\}$, then the blocks are:

$$\{0, 3, 4\} \{0, 7, 8\} \{0, 11, 12\} \{1, 3, 5\} \{1, 7, 9\} \{1, 11, 13\} \{2, 3, 6\} \{2, 7, 10\} \{2, 11, 14\} \\ \{4, 9, 14\} \{4, 10, 13\} \{5, 8, 14\} \{5, 10, 12\} \{6, 8, 13\} \{6, 9, 12\} .$$

Blocks $\{0, 3, 4\}$ and $\{1, 7, 9\}$ don't share an object, but there are three other blocks that share an object with both: (i) block $\{0, 7, 8\}$ by sharing objects 0 and 7 , (ii) block $\{1, 3, 5\}$ by sharing objects 3 and 1 , and (iii) block $\{4, 9, 14\}$ by sharing objects 4 and 9 .

Example 4: *Complementary Design* $GQ(s, t) = GQ(2, 2)$ of *Example-3* will have $b = 15$ blocks where each block has $v - s - 1 = 12$ objects.

Symmetric Design	Key Distribution
Object Set (S)	→ Key-Pool (P)
Object Set Size ($ S = v = n^2 + n + 1$)	→ Key-Pool Size ($ P $)
Blocks	→ Key-Chains
# Blocks ($b = n^2 + n + 1$)	→ # Key-Chains (N)
# Blocks ($b = n^2 + n + 1$)	→ # Sensor Nodes (N)
# Objects in a Block ($k = n + 1$)	→ # Keys in a Key-Chain (K)
# Blocks that an Object is in ($r = n + 1$)	→ # Key-Chains that a Key is in
Two Blocks share ($\lambda = 1$) Objects	→ Two Key-Chains share (χ) Keys

Table 1. Mapping from Symmetric Design to Key Distribution

$\{1, 2, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14\}$ $\{1, 2, 3, 4, 5, 6, 9, 10, 11, 12, 13, 14\}$
 $\{1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 13, 14\}$ $\{0, 2, 4, 6, 7, 8, 9, 10, 11, 12, 13, 14\}$
 $\{0, 2, 3, 4, 5, 6, 8, 10, 11, 12, 13, 14\}$ $\{0, 2, 3, 4, 5, 6, 7, 8, 9, 10, 12, 14\}$
 $\{0, 1, 4, 5, 7, 8, 9, 10, 11, 12, 13, 14\}$ $\{0, 1, 3, 4, 5, 6, 8, 9, 11, 12, 13, 14\}$
 $\{0, 1, 3, 4, 5, 6, 7, 8, 9, 10, 12, 13\}$ $\{0, 1, 2, 3, 5, 6, 7, 8, 10, 11, 12, 13\}$
 $\{0, 1, 2, 3, 5, 6, 7, 8, 9, 11, 12, 14\}$ $\{0, 1, 2, 3, 4, 6, 7, 9, 10, 11, 12, 13\}$
 $\{0, 1, 2, 3, 4, 6, 7, 8, 9, 11, 13, 14\}$ $\{0, 1, 2, 3, 4, 5, 7, 9, 10, 11, 12, 14\}$
 $\{0, 1, 2, 3, 4, 5, 7, 8, 10, 11, 13, 14\}$.

3 Combinatorial Design to Key Distribution

In the following two sections, we describe how *Symmetric Designs* and *Generalized Quadrangles* are used to generate key-chains for the sensors in a sensor network.

3.1 Mapping from Symmetric Design to Key Distribution

In this work, we are interested in *Finite Projective Plane* of order n which is a *Symmetric Design (Symmetric BIBD)* with parameters $(n^2 + n + 1, n + 1, 1)$.

Mapping: We assume a distributed sensor network where there are N sensor nodes. Sensor nodes communicate with each other and require pairwise keys to secure their communication. Each sensor has a *key-chain* of K keys which is stored to its ROM before the deployment. Keys are selected from a set P of *key-pool*. To secure the communication between them, a pair of sensor nodes need to have χ keys in common in their key-chains. Based on this, we define mapping given in Table-1

For a sensor network of N nodes, with total of N key-chains, a Symmetric Design with $b \geq N$ blocks needs to be constructed by using set S with $|S| = v = b$ objects. That means, $b = v = n^2 + n + 1 \geq N$ for a prime power n [7, 1]. Each object in S can be associated with a distinct random key, and each block can be used as a key-chain. That provides $b \geq N$ key-chains each having $K = k = n + 1$

keys. Symmetric Design guarantees that any pair of blocks has λ objects in common, meaning that any pair of key-chains, or equivalently sensor nodes, has $\chi = \lambda$ keys in common.

Construction: There are several methods to construct *Symmetric Designs* of the form $(n^2 + n + 1, n + 1, 1)$. In this project, we use a *complete set* of $(n - 1)$ *Mutually Orthogonal Latin Squares (MOLS)*. A *Latin Square* on n symbols is an $n \times n$ array such that each of the n symbols occurs exactly once in each row and in each column. The number n is called the *order of the square*. If $A = (a_{ij})$ and $B = (b_{ij})$ are any two $n \times n$ arrays, the *join* of A and B is a $n \times n$ array whose $(i, j)^{th}$ element is the pair (a_{ij}, b_{ij}) . The Latin Squares A and B of order n are *Orthogonal* if all entries of the join of A and B are distinct. Latin Squares A_1, A_2, \dots, A_r are *Mutually Orthogonal Latin Squares (MOLS)* if they are orthogonal in pairs. For prime power n , a set of $(n - 1)$ MOLS of order n is called a *Complete Set* [7, 1]. A complete set of $(n - 1)$ MOLS can be used to construct *Affine Plane* of order n which is an $(n^2, n, 1)$ design. *Affine Plane* of order n can be converted to *Projective Plane* of order n which is a $(n^2 + n + 1, n + 1, 1)$ *Symmetric Design*. The construction algorithm can be summarized as follows:

1. Given a network size of N , find a prime power n where $n^2 + n + 1 \geq N$,
2. Generate a complete set of $(n - 1)$ MOLS of order n [1, Theorem 5.1.1],
3. Construct the Affine Plane of order n from the MOLS [1, Theorem 1.3.5],
4. Construct the Projective Plane of order n from the Affine Plane [1, Theorem 1.2.5].

Analysis: Symmetric Design has a very nice property that, any pair of blocks shares exactly one object. Probability of key share between any pair of nodes is $P_{SYM} = 1$, so that *Average Key Path Length* is 1.

Resilience is an important metric for the security. Attackers might be capturing nodes selectively or randomly. In the case of selective capture, we may simply assume that attacker has ability to monitor whole network and selects the nodes wisely. Since key-chain size is $n + 1$ for a symmetric design with $n^2 + n + 1$ nodes and keys, attacker will need at least $n + 1$ key-chains to be able to recover all the keys. A wise attacker may select to capture the nodes which have the same *specific key* in their key-chains. From the properties of the symmetric design, we know that there are $n + 1$ such key-chains. Since every pair of keys can occur in exactly one key-chain, then every $n^2 + n$ keys must be pairing with the *specific key* in these $n + 1$ key-chains. But an unlucky attacker who selects the nodes randomly might be capturing n^2 key-chains which do not include the *specific key*. Therefore an unlucky attacker will need to capture $n^2 + 1$ key-chains to be able to recover all the keys.

Symmetric Design of the form $(n^2 + n + 1, n + 1, 1)$ is not a scalable solution itself. Given a fixed key-chain size $k = n + 1$, it can support network sizes of N where $N \leq n^2 + n + 1$. For networks of size $N < n^2 + n + 1$, simply some of blocks may not be used still preserving key sharing probability $P_{SYM} = 1$. For the networks where $N > n^2 + n + 1$, key-chain size must be increased, that is, n must be increased to next prime power. Due to the memory limitations of a

$GQ(s,t)$	s	t	b	v
$GQ(q, q)$	q	q	$q^3 + q^2 + q + 1$	$q^3 + q^2 + q + 1$
$GQ(q, q^2)$	q	q^2	$q^5 + q^3 + q^2 + 1$	$q^4 + q^3 + q + 1$
$GQ(q^2, q^3)$	q^2	q^3	$q^8 + q^5 + q^3 + 1$	$q^7 + q^5 + q^2 + 1$

Table 2. The $GQ(s, t)$ parameters.

Generalized Quadrangle $GQ(s, t)$	Key Distribution
Point Set (P)	→ Key-Pool (P)
Point Set Size ($ S = v = (s + 1)(st + 1)$)	→ Key-Pool Size ($ P $)
Line Set (B)	→ Key-Chains
# Lines ($ B = b = (t + 1)(st + 1)$)	→ # Key-Chains (N)
# Lines ($ B = b = (t + 1)(st + 1)$)	→ # Sensor Nodes (N)
# Points on a Line ($s + 1$)	→ # Keys in a Key-Chain (K)
# Lines that a Point is incident ($t + 1$)	→ # Key-Chains that a Key is in
Two Lines share (≤ 1) points	→ Two Key-Chains share (χ) Keys

Table 3. Mapping from GQ to Key Distribution

sensor node, this may not be a good solution. Moreover, such an increase in n may produce designs which can support much bigger networks than required. In probabilistic key distribution schemes, it is always possible to increase size of key-pool for a fixed key-chain size to increase the possible number of distinct key-chains. But, such an approach sacrifices the key share probability and requires better connectivity at underlying physical network. It is possible to merge deterministic and probabilistic designs to inherit advantages of both. Later in Section-4, we propose *Hybrid* of Symmetric and Probabilistic Designs to cope with scalability problems. Basically, we use $n^2 + n + 1$ blocks of the Symmetric Design and select uniformly at random remaining $N - (n^2 + n + 1)$ blocks among the $(k = n + 1)$ -subsets of the Complementary Symmetric Design.

3.2 Mapping from Generalized Quadrangles to Key Distribution

In this work, we are interested in three known $GQ(s, t)$: $GQ(q, q)$, $GQ(q, q^2)$ and $GQ(q^2, q^3)$. Table-2 gives details about their parameters.

Mapping: Consider a sensor network of N nodes where each node requires a *key-chain* having K keys coming from a *key-pool* P . Assume also that, not all pairs of neighboring nodes need to share a key directly, they can communicate through a secure path on which every pair of neighboring nodes shares a key. GQ can be used to generate key-chains for such networks. Namely, points in GQ can be considered as the keys and lines as the key-chains. Mapping between GQ and Key Distribution is given in Table-3.

GQ	PG	Points	Canonical Equation for PG
$GQ(q, q)$	$PG(4, q)$	$(x_0, x_1, x_2, x_3, x_4)$	$x_0^2 + x_1x_2 + x_3x_4 = 0$
$GQ(q, q^2)$	$PG(5, q)$	$(x_0, x_1, x_2, x_3, x_4, x_5)$	$f(x_0, x_1) + x_2x_3 + x_4x_5 = 0$
$GQ(q^2, q^3)$	$PG(4, q^2)$	$(x_0, x_1, x_2, x_3, x_4)$	$x_0^{q+1} + x_1^{q+1} + x_2^{q+1} + x_3^{q+1} + x_4^{q+1} = 0$

Table 4. Projective Space Equations

In GQ, there are $t+1$ lines passing through a point, and a line has $s+1$ points. That means, a line shares a point with exactly $t(s+1)$ other lines. Moreover, if two lines, say lines A and B , do not share a point, then for each point pt_A on line A , there is a point pt_B on line B such that there exist a line C passing through both points pt_A and pt_B . That means, if two lines A and B do not share a point, there are $s+1$ distinct lines which share a point with both lines A and B . In terms of Key Distribution, meaning that, a block shares a key with $t(s+1)$ other blocks. Additionally, if two blocks do not share a key, there are $s+1$ other blocks sharing a key with both.

Construction: The three $GQ(s, t)$'s used in this work are incidence relations between points and lines in a *Projective Space* $PG(d, q)$ and $PG(d, q^2)$ with dimension d . Points of the space are vectors with $d+1$ elements of the form $(x_0, x_1, x_2, \dots, x_d)$ where $x_i < q$ for $PG(d, q)$ and $x_i < q^2$ for $PG(d, q^2)$. They hold the projective plane equations given in Table-4. We use irreducible binary quadratic $f(x_0, x_1) = dx_0^2 + x_0x_1 + x_1^2$ for $GQ(q, q^2)$ as given in Table-4. Our construction algorithm can be summarized as follows:

1. Given network size of N , find a prime power q where:
 - $b = q^3 + q^2 + q + 1 \geq N$ for $GQ(q, q)$.
 - $b = q^5 + q^3 + q^2 + 1 \geq N$ for $GQ(q, q^2)$.
 - $b = q^8 + q^5 + q^3 + 1 \geq N$ for $GQ(q^2, q^3)$.
2. Find all points in Projective Space $PG(4, q)$ for $GQ(q, q)$, $PG(5, q)$ for $GQ(q, q^2)$ and $PG(4, q^2)$ for $GQ(q^2, q^3)$. That is, find all points holding given canonical equation.
3. Construct bilinear groups of size $s+1$ from v points, that is, find $s+1$ points which are on the same line. Note that each point is incident to $t+1$ lines.

Analysis: In a $GQ(s, t)$, there are $b = (t+1)(st+1)$ lines and a line intersects with $t(s+1)$ other lines. Thus, in a design generated from a GQ, a block shares an object with $t(s+1)$ other blocks. Probability P_{GQ} that two blocks share at least one object, or equivalently, probability P_{GQ} that a pair of nodes shares at least one key is:

$$P_{GQ} = \frac{t(s+1)}{b} = \frac{t(s+1)}{(t+1)(st+1)}.$$

Table-5 lists key share probabilities for the three GQs.

GQ	Pairwise Key Sharing Probability
$GQ(q, q)$	$P_{QQ} = \frac{q^2+q}{q^3+q^2+q+1}$
$GQ(q, q^2)$	$P_{QQ^2} = \frac{q^3+q^2}{q^5+q^3+q^2+1}$
$GQ(q^2, q^3)$	$P_{Q^2Q^3} = \frac{q^5+q^3}{q^8+q^5+q^3+1}$

Table 5. Pairwise Key Sharing Probabilities

	I	II	III	IV
$GQ(q, q)$	2	3	3	2
$GQ(q, q^2)$	3	2	1	4
$GQ(q^2, q^3)$	4	1	2	3
Symmetric	1	4	4	1

Table 6. Comparison of GQ and Symmetric Designs

$GQ(s, t)$ provides better resilience than the symmetric design. An unlucky attacker may need to capture $st(t + 1)$ blocks before selecting a block which includes a *specific point*. Therefore, an unlucky attacker needs to capture $st^2 + st + 1$ nodes to recover all the keys. For example, with the key-chain size of $q + 1$ in $GQ(s, t) = GQ(q, q^2)$ design, an unlucky attacker will need to capture $q^5 + q^3 + 1$ nodes before recovering all the keys. A wise attacker who is capturing nodes having the same *specific key* will capture $t + 1$ nodes. Attacker will be able to recover $ts + s + 1$ out of $ts^2 + ts + s + 1$ keys this time. With $GQ(s, t) = GQ(q, q^2)$ design, a wise attacker only capturing $q^2 + 1$ nodes having the same specific key will be able to recover $q^3 + q + 1$ keys out of $q^4 + q^3 + q + 1$ keys.

Comparison of Symmetric and GQ Designs for Key Distribution: We are interested in four metrics when comparing the designs: (I) probability of *object share* between any pair of blocks, (II) *block size* which is the number of the objects in a block, (III) *number of blocks* generated by the design, and (IV) *complexity* of the design. *Object-pool size*, which is the number of the objects in the key-pool, might be another metric. But, *object-pool size* metric shows same characteristics as *number of blocks* metric.

Table-6 compares GQ and Symmetric designs based on these four metrics. Rows are the designs and columns are the metrics. Table entry for *row - i* and *column - j* is the rank of the design in *row - i*, based on the metric in *column - j*. Designs are ranked from best (rank 1) to worst (rank 4) in each column. Symmetric Design is the simplest design and provides highest number of object share. $GQ(q, q^2)$ provides highest number of blocks for the fixed block size. $GQ(q^2, q^3)$ provides smallest block size for the fixed number of blocks. The table can be used to decide on the proper design to generate networks with given objectives (connectivity, size, storage limits, etc.)

Design	k	r	b	v
<i>Symmetric</i>	$n + 1$	$n + 1$	$n^2 + n + 1$	$n^2 + n + 1$
<i>Complementary Symmetric</i>	n^2	n^2	$n^2 + n + 1$	$n^2 + n + 1$
<i>GQ(n, n)</i>	$n + 1$	$n + 1$	$n^3 + n^2 + n + 1$	$n^3 + n^2 + n + 1$
<i>Complementary GQ(n, n)</i>	$n^3 + n^2$	$n^3 + n^2$	$n^3 + n^2 + n + 1$	$n^3 + n^2 + n + 1$
<i>GQ(n, n²)</i>	$n + 1$	$n^2 + 1$	$n^5 + n^3 + n^2 + 1$	$n^4 + n^3 + n + 1$
<i>Complementary GQ(n, n²)</i>	$n^4 + n^3$	$n^5 + n^3$	$n^5 + n^3 + n^2 + 1$	$n^4 + n^3 + n + 1$
<i>GQ(n², n³)</i>	$n^2 + 1$	$n^3 + 1$	$n^8 + n^5 + n^3 + 1$	$n^7 + n^5 + n^2 + 1$
<i>Complementary GQ(n², n³)</i>	$n^7 + n^5$	$n^8 + n^5$	$n^8 + n^5 + n^3 + 1$	$n^7 + n^5 + n^2 + 1$

Table 7. Parameters k, r, v, b for Symmetric, GQ and their Complementary Designs

Probabilistic key distribution is the simplest and most scalable solution when compared with GQ and Symmetric Designs. Next, in Section-4, we propose Hybrid Symmetric and GQ Designs which provide solutions as scalable as probabilistic key distribution schemes, yet taking advantages of underlying GQ and Symmetric Designs.

4 Hybrid Designs for Scalable Key Distributions

The main drawback of the combinatorial approach comes from the difficulty of their construction. Given a desired number of sensor nodes or a desired number of keys in the pool, we may not be able to construct a combinatorial design for the target parameters.

In this work, we present a novel approach called *Hybrid Design* which combines deterministic core and probabilistic extensions. We will consider two Hybrid Designs: *Hybrid Symmetric Design* and *Hybrid GQ Design*. By using Symmetric or GQ Design and its complement, we preserve nice properties of combinatorial design yet take advantages of flexibility and scalability of probabilistic approaches to support any network sizes.

4.1 Mapping

Consider a sensor network where there are N nodes, therefore N key-chains are required. Due to memory limitations, key-chains can store at most K keys coming from key-pool P . We can employ Hybrid Design for the cases where there is no known combinatorial design technique to generate design with N nodes for the given key-chain size K . Basically, Hybrid Design finds largest prime power n such that $k \leq K$ and generates N blocks of size k where objects come from object set S of size $|S| = v$. The b of N blocks are generated by base Symmetric or GQ Design and $N - b$ blocks are randomly selected among k -subsets of the Complementary Design blocks. We define mappings as in Table-8.

Hybrid Symmetric Design	Key Distribution
Object Set (S)	→ Key-Pool (P)
Object Set Size ($ S = v$)	→ Key-Pool Size ($ P $)
Blocks of base design and selected (k)-subsets from Complementary Design	→ Key-Chains
# blocks from base design (b) + # selected (k)-subsets ($N - b$)	→ # Key-Chains (N)
# blocks from base design (b) + # selected (k)-subsets ($N - b$)	→ # Sensor Nodes (N)
# Objects in a Block ($k \leq K$)	→ # Keys in a Key-Chain (K)
Two Blocks share zero or more Objects	→ Two Key-Chains share (χ) Keys

Table 8. Mapping from Hybrid Design to Key Distribution

4.2 Construction

For a given key-chain size K and network size N , Hybrid Design first generates the Base Symmetric or GQ Design with largest possible prime power n where $k \leq K$. Base Symmetric or GQ Design has b blocks of size k . Table-7 lists the relations between block size k and number of blocks b for the prime power n . Next step is to generate Complementary Design where there are b blocks of size $v - k$. Table-7 lists the parameters of the Complementary Designs. Due to the fact that $v - k > k$ for Symmetric and GQ designs, Complementary Design blocks can't be used as the key-chains, but their subsets can. To scale the base design up to given network size, Hybrid Design randomly selects remaining $N - b$ blocks uniformly at random among k -subsets of the Complementary Design blocks. Selected k -subsets along with the blocks of the base design form blocks of the Hybrid Design. Algorithm can be summarized as follows:

1. Given N sensor nodes where each can store a key-chain of size K , find largest possible prime power n such that $k \leq K$ for k values given in Table-7.
2. Generate the base design (Symmetric or GQ):
 - Generate an object pool $P = \{a_1, a_2, \dots, a_v\}$ of size v ,
 - Generate the blocks $B = \{B_1, B_2, \dots, B_b\}$ where $|B_i| = k$ for $1 \leq i \leq b$ and $B_i \subset P$.
3. Generate Complementary Design from the base design:
 - Generate blocks $\overline{B} = \{\overline{B}_1, \overline{B}_2, \dots, \overline{B}_b\}$ where $\overline{B}_i = P - B_i$ and $|\overline{B}_i| = v - k$ for $1 \leq i \leq b$.
4. Generate $N - b$ hybrid blocks $H = \{H_1, H_2, \dots, H_{N-b}\}$ of size $|H_i| = k$ ($1 \leq i \leq N - b$) from the Complementary Design $\overline{B} = \{\overline{B}_1, \overline{B}_2, \dots, \overline{B}_b\}$. Use variable s_i to hold index of the block in \overline{B} from which the block H_i is obtained:
 - Consider all k -subsets of all blocks in \overline{B} ,
 - Randomly select $N - b$ distinct k -subsets to generate the set H ,
 - For each selected k -subset H_i ($1 \leq i \leq N - b$), find the block $\overline{B}_j \in \overline{B}$ ($1 \leq j \leq b$) from which block H_i is obtained. Set $s_i = j$.
5. Blocks of the Hybrid Design are $B \cup H$.

Example 5: Assume that we would like to generate key-chains for a network with $N = 10$ nodes. Assume also that nodes have very limited memories, so that they can store at most $K = 3$ keys in their key-chains. Hybrid Symmetric Design can be used to generate design for this network. Symmetric Design $(v, k, \lambda) = (7, 3, 1)$ of *Example-1* can be used as the base design to generate $b = 7$ blocks out of $v = 7$ objects where block size is $k = 3$. Blocks of Symmetric Design form the set $B = \{\{1,2,3\}, \{1,4,5\}, \{1,6,7\}, \{2,4,6\}, \{2,5,7\}, \{3,4,7\}, \{3,5,6\}\}$. Remaining $N - b = 3$ blocks are selected uniformly at random among the 3-subsets of the Complementary Symmetric Design $\overline{B} = \{\{4,5,6,7\}, \{2,3,6,7\}, \{2,3,4,5\}, \{1,3,5,7\}, \{1,3,4,6\}, \{1,2,5,6\}, \{1,2,4,7\}\}$. Assume that selected blocks are $\{4,5,6\}$, $\{2,3,6\}$ and $\{1,5,7\}$ which are the 3-subsets of the sets $\{4,5,6,7\}$, $\{2,3,6,7\}$ and $\{1,3,5,7\}$ respectively. These blocks (3-subsets) form the set $H = \{\{4,5,6\}, \{2,3,6\}, \{1,5,7\}\}$. The blocks of the Hybrid Symmetric Design are then $B \cup H = \{\{1,2,3\}, \{1,4,5\}, \{1,6,7\}, \{2,4,6\}, \{2,5,7\}, \{3,4,7\}, \{3,5,6\}, \{4,5,6\}, \{2,3,6\}, \{1,5,7\}\}$.

4.3 Analysis

In this section, we analyze some important properties of Hybrid Symmetric and Hybrid GQ Designs. We will look for some useful properties coming from underlying combinatorial design. Based on these properties, we will analyze object share probabilities between any pair of blocks in Hybrid Design $B \cup H$, where B is the set of blocks from the base (Symmetric or GQ) design and H is the set of blocks which are uniformly at random selected among k -subsets of the Complementary Design blocks \overline{B} (variable s_i holds index of the block in \overline{B} from which block $H_i \in H$ is obtained). Proofs for the properties and theorems are given in the appendix.

Hybrid Symmetric Design:

Property 1. Given Hybrid Design $B \cup H$, $\forall \beta \in B$ and $\theta \in H$, $\exists b \in \beta | b \notin \theta$. \square

Property-1 doesn't hold among the blocks in H . To see that, consider two such distinct blocks $H_i \in H$ and $H_j \in H$ where $s_i \neq s_j$. Complementary Design of a Symmetric Design has the property that any pair of blocks has $n^2 - n$ objects in common. For $n > 2$, when $(n^2 - n) > (n + 1)$, it can be the case that randomly selected blocks (k -subsets) H_i and H_j of size $k = n + 1$ are equivalent.

Property 2. Given the key chain size $k = n + 1$, Hybrid Symmetric Design can support network sizes up to:

$$\binom{v}{k} = \binom{n^2 + n + 1}{n + 1}. \quad \square$$

This is the maximum network size that the simple probabilistic key pre-distribution scheme can support for the key-chain size $k = n + 1$ and the key-pool size $v = n^2 + n + 1$. Probabilistic scheme can go beyond this limit by simply increasing the key-pool size v for a fixed key-chain size k . To provide the

same scalability, we employ Hybrid GQ Designs which are analyzed in the next section. For fixed key chain size $k = n + 1$, $GQ(n, n^2)$ will be able to generate designs for networks up to:

$$\binom{v}{k} = \binom{n^4+n^3+n+1}{n+1} .$$

This is the upper limit of our deterministic algorithms. Numerically, for key chain size of 4, our Hybrid $GQ(n, n^2)$ Design supports network sizes up to 6, 210, 820. It supports (2.54×10^{14}) nodes for $k = 6$, (8.08×10^{22}) nodes for $k = 8$, (1.18×10^{32}) nodes for $k = 10$, (5.78×10^{41}) nodes for $k = 12$ and so on.

Consider blocks $B \cup H$ of the Hybrid Symmetric Design. Any pair of blocks (β, θ) selected from the set $B \cup H$ can be either one of the four types:

1. *Type-BB*: $\beta \in B$ and $\theta \in B$.
2. *Type-HH*: $\beta = H_i \in H$ and $\theta = H_j \in H$ and $s_i \neq s_j$.
3. *Type-H*: $\beta = H_i \in H$ and $\theta = H_j \in H$ and $s_i = s_j$.
4. *Type-HB*: $(\beta \in B$ and $\theta \in H)$ or $(\beta \in H$ and $\theta \in B)$.

Properties 3 to 6 below give probability of sharing at least one object for pairs of the types BB , HH , H and HB respectively. Property-7 presents probabilities that these types happen. Finally, Theorem-1 presents probability of object share in Hybrid Symmetric Design.

Property 3. The probability P_{BB} that any pair of blocks (β, θ) (where $\beta \in B$ and $\theta \in B$) of the base Symmetric Design has exactly one object in common is $P_{BB} = 1$. \square

Property 4. The probability P_{HH} that any pair of blocks (β, θ) where $\beta = H_i \in H$, $\theta = H_j \in H$ and $s_i \neq s_j$ has at least one object in common approaches to 1 as n (block size) increases. \square

Property 5. The probability P_H that any pair of blocks (β, θ) where $\beta = H_i \in H$, $\theta = H_j \in H$ and $s_i = s_j$ has at least one object in common is :

$$P_H = 1 - \frac{\binom{n^2-n-1}{n+1}}{\binom{n^2}{n+1}} . \quad \square$$

Property 6. The probability P_{HB} that any pair of blocks (β, θ) where $(\beta \in B$ and $\theta \in H)$ or $(\beta \in H$ and $\theta \in B)$ has at least one object in common is:

$$\frac{\frac{1}{2}n^2 + \frac{3}{2}n + 1}{n^2 + n + 1} \leq P_{HB} \leq \frac{n^2 + 2}{n^2 + n + 1} . \quad \square$$

Property 7. Consider any pairing between N blocks of $B \cup H$ of Hybrid Symmetric (or GQ) Design. Any pair of blocks selected from this set can be either one of the four types: BB , HB , H or HB . Probability Q_α that a pair of blocks is Type- α for $\alpha \in \{BB, HB, H, HB\}$ is:

$$Q_{BB} = \frac{b(b-1)}{N(N-1)} , Q_{HB} = \frac{2b(N-b)}{N(N-1)} , Q_H = \frac{(N-b)(N-2b)}{bN(N-1)} , Q_{HH} = \frac{(b-1)(N-b)^2}{bN(N-1)} .$$

where $Q_{BB} + Q_{HB} + Q_H + Q_{HH} = 1$. \square

Thus we have the following theorem:

Theorem 1. *Probability P_{HSYM} that any pair of blocks shares a key in Hybrid Symmetric Design is:*

$$P_{HSYM} = P_{BB}Q_{BB} + P_{HB}Q_{HB} + P_HQ_H + P_{HH}Q_{HH} . \quad \square$$

Hybrid GQ Designs:

Property 8. Given key chain size $k = n + 1$, Hybrid GQ Design can support network sizes up to:

$$\binom{v}{s+1} = \binom{(s+1)(st+1)}{s+1} . \quad \square$$

Consider blocks $B \cup H$ of the Hybrid GQ Design. Any pair of blocks (β, θ) selected from the set $B \cup H$ can be either one of the four types: BB , HH , H and HB . Properties 9 to 12 below give probability of sharing at least one object between pairs of the types BB , HH , H and HB respectively. Finally, Theorem-2 presents probability of object share in Hybrid GQ Design.

Property 9. The probability P'_{BB} that any pair of blocks (β, θ) (where $\beta \in B$ and $\theta \in B$) of the base GQ Design has exactly one object in common is given in Table-5 for $GQ(q, q)$, $GQ(q, q^2)$ and $GQ(q^2, q^3)$. \square

Property 10. The probability P'_{HH} that any pair of blocks (β, θ) where $\beta = H_i \in H$, $\theta = H_j \in H$ and $s_i \neq s_j$ has at least one object in common approaches to 1 as n (block size) increases. \square

Property 11. The probability P'_H that any pair of blocks (β, θ) where $\beta = H_i \in H$, $\theta = H_j \in H$ and $s_i = s_j$ has at least one object in common is :

$$P'_H = 1 - \frac{\binom{(s+1)(st-1)}{s+1}}{\binom{st(s+1)}{s+1}} . \quad \square$$

Property 12. The probability P'_{HB} that any pair of blocks (β, θ) where $(\beta \in B$ and $\theta \in H)$ or $(\beta \in H$ and $\theta \in B)$ has at least one object in common is:

$$\frac{(s+1)(t-s/2+1)}{(t+1)(st+1)} \leq P'_{HB} \leq \frac{(st-s+t+2)}{(t+1)(st+1)} . \quad \square$$

Theorem 2. *Probability P_{HGQ} that any pair of blocks shares a key in Hybrid GQ Design is:*

$$P_{HGQ} = P'_{BB}Q_{BB} + P'_{HB}Q_{HB} + P'_H Q_H + P'_{HH}Q_{HH} . \quad \square$$

5 Computational Results

We have implemented *Random Key Pre-distribution Scheme* by Eschenauer *et al.* [13], *Symmetric Design*, $GQ(q, q)$, $GQ(q, q^2)$, *Hybrid Symmetric Design*, and compared them with each other. In random key pre-distribution scheme, we initially generate a large pool of P keys and their identities. For each sensor, we uniformly at random draw k keys from the key-pool P without replacement. These k keys and their identities form the *key-chain* for a sensor node.

Basically, for a network of size N , we generate N key-chains and assign them to N sensor nodes. Then, we uniformly at random distribute N nodes in to a 1×1 unit grid. Every wireless sensor has a coverage of radius r where $r = d(\ln N)/N$, all nodes within this coverage area is accepted as neighbors. Note that, parameter d can be used to play with radius r and therefore average degree of the network.

Pool Size (P)	Key Chain Size (k)	Number Sensor Nodes	Random Prob.	Symmetric Prob.	Random Avg. Key Path	Symmetric Avg. Key Path	Avg. Node Degree
100807	318	100807	0.634	1.0	—	1.0	—
10303	102	10303	0.639	1.0	1.35	1.0	56
5113	72	5113	0.642	1.0	1.35	1.0	51
2863	54	2863	0.645	1.0	1.35	1.0	47
1407	38	1407	0.651	1.0	1.34	1.0	42
553	24	553	0.663	1.0	1.33	1.0	35

Table 9. Symmetric Design vs Random Key Pre-distribution

After the deployment, two neighboring nodes compare the keys in their key-chains by using the key id's. If they have a key in common, it is used to secure the communication. If there is no key in common, they try to find a shortest possible path where each pair of nodes on the path shares a key. Length of this path is called *Key Path Length* where *Key Path Length* for two nodes directly sharing a key is 1. *Average Key Path Length* is one of the metrics that we use to compare random key pre-distribution scheme with our Deterministic and Hybrid Design schemes.

Probability p that two key-chains share at least one key is another metric we use in comparison. For random key pre-distribution scheme, for a given key-pool size P and key-chain size k , Eschenauer *et al.* in [13] approximate probability p as:

$$P_{RAND} = \left[1 - \frac{(1 - \frac{k}{P})^{2(P-k+1/2)}}{(1 - \frac{2k}{P})^{(P-2k+1/2)}} \right].$$

In Symmetric Design, $P_{SYM} = 1$ since any pair of key-chains shares exactly one key. In $GQ(s, t)$, probability of key share P_{QQ} for $GQ(q, q)$, P_{QQ^2} for $GQ(q, q^2)$ and $P_{Q^2Q^3}$ for $GQ(q^2, q^3)$ is given in Table-5.

Probability of key share P_{HSYM} is given in analysis section of the Hybrid Symmetric Design. Similarly, probability of key share P_{HGQ} for Hybrid GQ Design is given in analysis section of the Hybrid GQ Designs.

Tables 9, 10, 11, 12 and 13 summarize the computational results: (i) analytical solution for probability p that two key-chains share at least one key, and (ii) simulation results for *Average Key Path Length*.

Symmetric Design is compared with Random Key Pre-distribution scheme in Table-9. For the same network size, key-chain size and pool-size, Symmetric Design provides better probability of key share between any two key-chains. Simulation results for average key path length supports this advantage. In Random Key Pre-distribution scheme, a pair of nodes requires to go through a path of 1.35 hops on average to share a key and communicate securely. This path length is 1 for Symmetric Design.

$GQ(q, q)$ is compared with Random Key Pre-distribution scheme in Table-10. $GQ(q, q)$ decreases key-chain size, causing a small decrease in key sharing probability. Analytical solution shows that random key pre-distribution scheme

Pool Size (P)	Key Chain Size (k)	Number Sensor Nodes	Random Prob.	$GQ(q, q)$ Prob.	Random Avg. Key Path	$GQ(q, q)$ Avg. Key Path	Avg. Node Degree
7240	20	7240	0.053	0.052	2.68	2.69	205
5220	18	5220	0.060	0.058	2.89	2.88	148
2380	14	2380	0.079	0.076	3.17	3.18	88
1464	12	1464	0.094	0.090	2.73	2.71	81
400	8	400	0.150	0.140	3.61	3.49	32
156	6	156	0.212	0.192	2.82	2.53	25

Table 10. Generalized Quadrangle $GQ(q, q)$ vs Random Key Pre-distribution

Pool Size (P)	Key Chain Size (k)	Number Sensor Nodes	Random Prob.	$GQ(q, q^2)$ Prob.	Random Avg. Key Path	$GQ(q, q^2)$ Avg. Key Path	Avg. Node Degree
15984	12	162504	0.0008	0.0089	–	–	
2753	8	17200	0.003	0.022	2.96	2.96	631
756	6	3276	0.010	0.045	3.12	3.22	179
112	4	280	0.056	0.128	5.63	5.49	29
27	3	45	0.190	0.266	2.23	2.14	15

Table 11. Generalized Quadrangle $GQ(q, q^2)$ vs Random Key Pre-distribution

provides slightly better probability of key share between key-chains, but $GQ(q, q)$ is still competitive to random key pre-distribution scheme. When two key-chains do not share a key, $GQ(q, q)$ guarantees existence of third one which shares a key with both.

$GQ(q, q^2)$ is compared with Random Key Pre-distribution scheme in Table-11. Generalized Quadrangle provides better key sharing probability and performs better in simulation by producing shorter *Average Key Path Lengths*.

$GQ(q^2, q^3)$ is compared with Random Key Pre-distribution scheme in Table-12. Generalized Quadrangle provides much better key sharing probability, we expect that simulation results will produce much shorter *Average Key Path Length* compared to random key pre-distribution scheme.

Hybrid Symmetric Design is compared with Random Key Pre-distribution Scheme in Table-13. Hybrid Symmetric Design makes use of Symmetric Design, yet taking advantages of the scalability of probabilistic approach. Given target network size N and key chain size k for which there is no known design, computational results shows that Hybrid Symmetric Design shows much better performance than Probabilistic Design.

Pool Size (P)	Key Chain Size (k)	Number Nodes	Random Prob.	$GQ(q^2, q^3)$ Prob.
81276	6	393876	0.000091	0.008251
2440	4	6832	0.002340	0.039519
165	3	297	0.030098	0.134680

Table 12. Generalized Quadrangle $GQ(q^2, q^3)$ vs Random Key Pre-distribution

Pool Size (P)	Key Chain Size (k)	Number Sensor Nodes	Random Prob.	Hybrid Sym. Prob.	Random Avg. Key Path	Hybrid Sym. Avg. Key Path	Avg. Node Degree
10303	102	10500	0.632	0.99	1.36	1.01	56
5113	72	5250	0.632	0.99	1.35	1.01	51
2863	54	3000	0.628	0.98	1.35	1.03	47
1407	38	1500	0.627	0.97	1.34	1.04	42
553	24	750	0.547	0.89	1.33	1.15	37
183	14	250	0.563	0.89	1.31	1.14	29

Table 13. Hybrid Symmetric Design vs Random Key Pre-distribution

6 Conclusions

In this work we presented novel approaches to the key distribution problem in large scale sensor networks. In contrast with prior work, our approach is combinatorial based on Combinatorial Block Designs. We showed how to map from two classes of combinatorial designs to deterministic key distribution mechanisms. We remarked the scalability issues in the deterministic constructions and proposed hybrid mechanisms. Hybrid constructions combine a deterministic core design with probabilistic extensions to achieve key distributions to any network sizes.

The analysis and computational comparison to the randomized methods show that the combinatorial approach has clear advantages: (i) it increases the probability of a pair of sensor nodes to share a key, and (ii) decreases the key-path length while providing scalability with hybrid approaches.

7 Acknowledgements

We would like to thank to Malik Magdon-Ismail for his useful comments and suggestions on the proof of property-4.

References

1. I. Anderson, "Combinatorial Designs: Construction Methods," Ellis Horwood Limited, 1990.

2. R. Blom, "An optimal class of symmetric key generation systems," *EUROCRYPT* 84, 1985.
3. C. Blundo, A. De Santis, A. Herzberg, S. Kutten, U. Vaccaro, M. Yung, "Perfectly-secure key distribution for dynamic conferences," In *Advances in Cryptography - CRYPTO'92*, 1993.
4. D.W. Carman, B.J. Matt and G.H. Cirincione, "Energy-efficient and Low-latency Key Management for Sensor Networks", In *Proceedings of 23rd Army Science Conference*, 2002.
5. H. Chan, A. Perrig and D. Song, "Random Key Predistribution Schemes for Sensor Networks," In *2003 IEEE Symposium on Research in Security and Privacy*, 2003.
6. M. Chen, W. Cui, V. Wen and A. Woo, "Security and Deployment Issues in a Sensor Network," *Ninja Project, A Scalable Internet Services Architecture*, Berkeley, <http://citeseer.nj.nec.com/chen00security.html>, 2000.
7. C.J. Colbourn, J.H. Dinitz, "The CRC Handbook of Combinatorial Designs," CRC Press, 1996.
8. W. Du, J. Deng, Y. S. Han, P. Varshney, "A Pairwise Key Pre-distribution Scheme for Wireless Sensor Networks," In *Proceedings of the 10th ACM Conference on Computer and Communications Security (CCS)*, 2003.
9. W. Du, J. Deng, Y. S. Han, S. Chen, P. K. Varshney, "A Key Management Scheme for Wireless Sensor Networks Using Deployment Knowledge," *INFOCOM*, 2004.
10. J. Deng, R. Han and S. Mishra, "Enhancing Base Station Security in Wireless Sensor Networks," *Technical Report CU-CS-951-03*, Department of Computer Science, University of Colorado, 2003.
11. J. Deng, R. Han, and S. Mishra, "A Performance Evaluation of Intrusion-Tolerant Routing in Wireless Sensor Networks," *2nd International Workshop on Information Processing in Sensor Networks (IPSN '03)*, 2003.
12. P. Dembowski, "Finite Geometries," Springer Verlag, 1968.
13. L. Eschenauer, V. D. Gligor, "A key-management scheme for distributed sensor networks", *Proceedings of the 9th ACM conference on Computer and communications security*, 2002.
14. M. Hall, "Combinatorial Theory," Blaisdell Publishing Company, 1967.
15. J.W.P. Hirschfeld, "Projective Geometries Over Finite Fields," Clarendon Press Oxford, 1979.
16. D. Liu and P. Ning, "Efficient Distribution of Key Chain Commitments for Broadcast Authentication in Distributed Sensor Networks", *The 10th Annual Network and Distributed System Security Symposium*, February 2003
17. D. Liu, P. Ning, K. Sun, "Efficient self-healing group key distribution with revocation capability," *Proceedings of the 10th ACM conference on Computer and communication security*, 2003.
18. D. Liu, P. Ning, "Establishing pairwise keys in distributed sensor networks," *Proceedings of the 10th ACM conference on Computer and communication security*, 2003.
19. R. Merkle, "Secure Communication over insecure channels," *Communications of the ACM*, 1978.
20. S. E. Payne, J. A. Thas, "Finite Generalized Quadrangles," *Research Notes in Mathematics*, Pitman Advanced Publishing Program, 1984.
21. D. Pedoe, "An introduction to Projective Geometry," Oxford, 1963.
22. A. Perrig, R. Szewczyk, V. Wen, D. Culler and J. D. Tygar, "SPINS: Security Protocols for Sensor Networks," *Wireless Networks Journal (WINE)*, 2002.

23. S. Slijepcevic, M. Potkonjak, V. Tsitsis, S. Zimbeck, M. B. Srivastava, "On communication Security in Wireless Ad-Hoc Sensor Network," Eleventh IEEE International Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises (WETICE'02), 2002.
24. F. Stajano, R. Anderson, "The resurrecting duckling: security issues for ad-hoc wireless networks," AT&T software symposium, 1999.
25. D. R. Stinson, S. A. Vanstone, "A combinatorial approach to threshold schemes," Advances in Cryptology - CRYPTO '87, 1987.
26. D. R. Stinson, "A construction for authentication / secrecy codes from certain combinatorial designs," Advances in Cryptology - CRYPTO '87, 1987.
27. D. R. Stinson, "Combinatorial characterizations of authentication codes," Advances in Cryptology - CRYPTO '91, 1991.
28. Y. Song, A. Wool, B. Yener, "Combinatorial Design of Multi-ring Networks with Combined Routing and Flow Control," in Computer Networks Vol.3 No: 3, pp 247-267, 2003.
29. J. Undercoffer, S. Avancha, A. Joshi, and J. Pinkston, "Security for Sensor Networks," CADIP Research Symposium, 2002.
30. W.D. Wallis, "Combinatorial Design," Marcel Dekker Inc., 1988.
31. B. Yener, Y. Ofek, M. Yung, "Combinatorial Design of Congestion Free Networks," In IEEE/ACM Transactions on Networking, Vol. 5, No. 6, pages: 989-1000, December 1997.
32. S. Zhu, S. Xu, S. Setia, S. Jajodia, "Establishing Pairwise Keys for Secure Communication in Ad Hoc Networks: A Probabilistic Approach," 11th IEEE International Conference on Network Protocols (ICNP'03), 2003.
33. S. Zhu, S. Setia, S. Jajodia, "LEAP: efficient security mechanisms for large-scale distributed sensor networks," Proceedings of the 10th ACM conference on Computer and communication security, 2003.

Appendix

Proof. (Property-1) From the definition of Symmetric Design. Since any pair of blocks shares exactly one object, a block in B can not be the subset of the complement of another block in B .

Given Hybrid Design $B \cup H$, consider any block $H_i \in H$ where $s_i = l$. Assume that $\exists B_j \in B \mid H_i = B_j$. We know that, $\forall x, x \in H_i \mid x \in \overline{B_l}$ and $x \notin B_l$. That means, blocks B_j and B_l do not share an object, contradicting the fact that B_j and B_l are blocks of Symmetric Design. Therefore, all blocks in H are distinct from the blocks of the Symmetric Design.

Proof. (Property-2) This can be shown by using the definition of Symmetric Design. For any set of k objects from the object pool P , if it is not one of the blocks in B , there is an upper bound for the number of blocks in B that objects of this set can appear in. This upper bound is less than the total number of Symmetric Design blocks. There are blocks in B that this set does not share an object, but is a subset of their complements.

Given Hybrid Design $B \cup H$, consider any set β of k objects where $\beta \notin B$. From the definition of Symmetric Design, we know that, any pair of elements of object pool must be in exactly $\lambda = 1$ block of the Symmetric Design. Moreover, each object must be in $r = k$ blocks. Consider the case where β shares one or more objects with maximum possible number of blocks in B . This happens when β shares $k - 1$ objects with one of the blocks in B . Remaining k^{th} object of β appears in the other $r = n + 1$ blocks of B . Each of $k - 1$ objects must be pairing with k^{th} object $\lambda = 1$ time. Therefore, $k - 1$ objects will appear alone in $r - 2 = n - 1$ other blocks and β shares one or more objects with at most:

$$1 + r + (r - 1)(r - 2) = n^2 + 2 .$$

blocks in B . But, $|B| = n^2 + n + 1$ and $n^2 + n + 1 > n^2 + 2$ for $n > 1$. That means, $\exists \theta \in B$ and $\overline{\theta} \in \overline{B} \mid \theta \cap \beta = \emptyset$ and $\beta \subset \overline{\theta}$. Therefore, Hybrid Symmetric Design can generate every k -subset of the object set P .

Proof. (Property-3) This is direct result from definition of Symmetric Design.

Proof. (Property-4) Instead of finding the probability, we bound it. Given Hybrid Design $B \cup H$, consider blocks H_l and H_k in H which are subsets of $\overline{B_L} \in \overline{B}$ and $\overline{B_K} \in \overline{B}$ respectively. Consider random variables X_i and Y_i for $1 \leq i \leq n^2 - n$. X_i takes on value 1 if i^{th} common element in $\overline{B_L}$ is selected. Similarly, Y_i takes on value 1 if i^{th} common element in $\overline{B_K}$ is selected. We are interested in case where X_i and Y_i both takes on value 1, meaning that the i^{th} is selected for both blocks. We define new indicator $Z_i = X_i.Y_i$ where:

$$E[Z] = \sum_{i=1}^{n^2-n} E[Z_i] = \sum_{i=1}^{n^2-n} E[X_i]E[Y_i]$$

$$E[Z] = \left(\frac{n+1}{n^2} \right)^2 (n^2 - n) > 1 .$$

$E[Z] > 1$ using Markov bound $P(Z \geq 1) \leq E[Z]$ and after checking second moment which shows no significant variance, we conclude that $P(Z \geq 1)$ approaches to 1 as n increases.

Proof. (Property-5) Given Hybrid Design $B \cup H$, consider Complementary Design \overline{B} where blocks in H are picked uniformly at random among the k -subsets of the blocks in \overline{B} . Let $\overline{P_H}$ be the probability that two k -subsets of a block $\beta \in \overline{B}$ have no objects in common. Then, $P_H = 1 - \overline{P_H}$. Let L be the number of all k -subsets of β . Consider a k -subset $\alpha \subset \beta$, where α does not share an object with all k -subsets of the set $\beta \setminus \alpha$. Let M be the all k -subsets of the set $\beta \setminus \alpha$. The k -subset α may pair with $L - 1$ other k -subsets and does not share an object with M of them. Therefore:

$$P_H = 1 - \overline{P_H}, \quad M = \binom{n^2 - n - 1}{n+1}, \quad L = \binom{n^2}{n+1}$$

$$\overline{P_H} = \frac{M}{L-1} \approx \frac{M}{L}.$$

Proof. (Property-6) Given Hybrid Design $B \cup H$, from the definition of Symmetric Design, (i) H_i may be sharing at most two objects with some blocks in B , (ii) H_i may be sharing $k - 1$ objects with exactly one of the blocks and share one or two objects with some other blocks in B . In the first case, an object needs to be coupled with $k - 1$ objects in $k - 1$ different blocks and it needs to be alone in $r - k + 1 = 1$ block. Number of the blocks that H_i shares an object with will be:

$$\binom{n+1}{2} + n + 1 = \frac{1}{2}n^2 + \frac{3}{2}n + 1.$$

From the discussion in proof of Property-2, in the second case, H_i will be sharing one or more objects with $n^2 + 2$ blocks in B where $|B| = b = n^2 + n + 1$. Therefore P_{HB} will be:

$$\frac{\frac{1}{2}n^2 + \frac{3}{2}n + 1}{n^2 + n + 1} \leq P_{HB} \leq \frac{n^2 + 2}{n^2 + n + 1}.$$

Proof. (Property-7) There are N blocks in block set $B \cup H$ of Hybrid Symmetric (or GQ) Design. Consider a pair of block uniformly randomly selected from this set.

(1) probability Q_{BB} that both blocks come from B is:

$$Q_{BB} = \frac{\binom{b}{2}}{\binom{N}{2}} = \frac{b(b-1)}{N(N-1)}.$$

(2) probability Q_{HB} that one block comes from H ($|H| = N - b$) and other from B ($|B| = b$) is:

$$Q_{HB} = \frac{b(N-b)}{\binom{N}{2}} = \frac{2b(N-b)}{N(N-1)}.$$

Probability that both blocks are selected from H is:

$$\frac{\binom{N-b}{2}}{\binom{N}{2}} = \frac{(N-b)(N-b-1)}{N(N-1)} .$$

(3) $N-b$ blocks in H are uniformly at random selected among the k -subset of the b blocks in \overline{B} . That means, average $(N-b)/b$ blocks are selected among the k -subsets of each block in \overline{B} . Therefore, probability Q_H that both blocks are from H , and are subsets of the same block in \overline{B} is:

$$\begin{aligned} Q_H &= \frac{(N-b)(N-b-1)}{N(N-1)} \frac{b \binom{(N-b)/b}{2}}{\binom{N-b}{2}} \\ &= \frac{(N-2b)(N-b)}{bN(N-1)} . \end{aligned}$$

(4) Probability Q_{HH} that both blocks are from H and are subsets of the distinct blocks in \overline{B} is:

$$\begin{aligned} Q_{HH} &= \frac{(N-b)(N-b-1)}{N(N-1)} \left[1 - \frac{b \binom{(N-b)/b}{2}}{\binom{N-b}{2}} \right] \\ &= \frac{(b-1)(N-b)^2}{bN(N-1)} . \end{aligned}$$

Proof. (Theorem-1) Consider any pairing between N blocks of Hybrid Symmetric Design $B \cup H$. Any pair of blocks selected from this set can be either one of the four types given in Property-7 (with probabilities Q_{BB} , Q_{HB} , Q_H and Q_{HH}). Probability of sharing at least one object between the pairs of each type is given in Properties 3, 4, 5 and 6 (P_{BB} , P_{HB} , P_H and P_{HH}). Then, probability P_{HSYM} that any pair of blocks shares at least one object in Hybrid Symmetric Design is:

$$P_{HSYM} = P_{BB}Q_{BB} + P_{HB}Q_{HB} + P_H Q_H + P_{HH}Q_{HH} .$$

Proof. (Property-8) Similar to proof of Property-2. This can be shown by using definition of GQ Design. For any set of k objects from the object pool P , if it is not one of the blocks in B , there is an upper bound for the number of blocks in B that objects of this set can appear in. This upper bound is less than the total number of GQ Design blocks. There are blocks in B that this set does not share an object, but is a subset of their complements.

Given Hybrid Design $B \cup H$, consider any set β of k objects from object pool P where $\beta \notin B$. Consider the case where β shares one or more objects with maximum possible number of blocks in B . From the definition of GQ Design, this happens when the set β shares s objects with a block in B . Remaining $(s+1)^{th}$ object will appear in the other $t+1$ blocks of B . Each of s objects must

be pairing with $(s + 1)^{th}$ object once. Therefore, s object will appear alone in $t - 1$ other blocks and β shares one or more objects with at most:

$$1 + (t + 1) + s(t - 1) = st + t - s + 2 .$$

blocks in B . But, there are $(t + 1)(st + 1)$ blocks in GQ Design and $st^2 + st + t + 1 \gg st + t - s + 2$. That means, $\exists \theta \in B$ and $\bar{\theta} \in \bar{B} \mid \theta \cap \beta = \emptyset$ and $\beta \subset \bar{\theta}$. Therefore, Hybrid GQ Design can generate any k -subset of the object set P .

Proof. (Property-10) From the discussion in property-4 it follows.

Proof. (Property-11) From the discussion in property-5 it follows.

Proof. (Property-12) From the discussion in property-6 it follows.

Proof. (Theorem-2) Similar to Theorem-1. From Properties 9, 10, 11, 12 and 7 it follows.