Graph-theoretic design and analysis of key predistribution schemes

Michelle Kendall Keith M. Martin

May 20, 2014

Abstract

Key predistribution schemes for resource-constrained networks are methods for allocating symmetric keys to devices in such a way as to provide an efficient trade-off between key storage, connectivity and resilience. While there have been many suggested constructions for key predistribution schemes, a general understanding of the design principles on which to base such constructions is somewhat lacking. Indeed even the tools from which to develop such an understanding are currently limited, which results in many relatively ad hoc proposals in the research literature.

It has been suggested that a large edge-expansion coefficient in the key graph is desirable for efficient key predistribution schemes. However, attempts to create key predistribution schemes from known expander graph constructions have only provided an extreme in the trade-off between connectivity and resilience: namely, they provide perfect resilience at the expense of substantially lower connectivity than can be achieved with the same key storage.

Our contribution is two-fold. First, we prove that many existing key predistribution schemes produce key graphs with good expansion. This provides further support and justification for their use, and confirms the validity of expansion as a sound design principle. Second, we propose the use of incidence graphs and concurrence graphs as tools to represent, design and analyse key predistribution schemes. We show that these tools can lead to helpful insights and new constructions.

1 Introduction

Key predistribution schemes (KPSs) are methods for allocating symmetric cryptographic keys to devices in a network. Where resources are constrained, schemes are typically designed to provide a trade-off between three conflicting parameters. These are: the *key storage* requirement for each device, which should be minimised; the *connectivity*, or sharing of keys between devices, which should be maximised; and the *resilience*, a measure of how often keys are re-used throughout the network, which should be minimised. These parameters are widely used to analyse and compare KPSs.

A KPS can be represented by a key graph, where vertices correspond to devices and edges represent the sharing of keys. In addition to the three parameters introduced above, it has been recognised that the topology of the key graph is also an important design consideration. In particular, it is desirable for the key graph to have good expansion [38], which in essence means that all subsets of vertices are well-connected to the rest of the graph. The papers [17, 49] use expander graph constructions to produce KPSs. The resulting schemes achieve an extreme in the trade-off between connectivity and resilience: they provide perfect resilience at the expense of lower connectivity than is typically provided by a KPS with the same key storage. Briefly, the constructions in [17, 49] take an expander graph construction, remove any self-loops and multi-edges, and then assign a unique key to each edge. Thus, each key is used to secure exactly one link, and so perfect resilience is achieved. However, this also means that if a node stores k keys then it only has degree k in the key graph. Other KPS constructions tend to re-use keys, such that a node which stores k keys will often have degree significantly greater than k, and hence the network has higher connectivity.

In [38] it was shown that many existing KPSs with less-than-perfect resilience produce highly connected key graphs with good expansion, thereby providing a more commonly desirable trade-off between connectivity and resilience than is afforded by the strong condition of perfect resilience. We develop this analysis, proving that the key graphs of many existing classes of KPSs have good expansion, which provides further support and justification for their use. In particular, we prove lower bounds for the expansion coefficient of KPSs constructed from μ-common intersection designs and strongly regular graphs.

We also propose alternative graph methods for representing the key graph of a KPS. We show that the incidence graph (or Levi graph / hypergraph) illustrates all the parameters of a KPS in one figure. This allows us to analyse KPSs

in a different way which leads to new constructions. We link these findings to analogous results in statistical design theory using the concurrence graph, and discuss the use of optimality criteria from this subject in the setting of KPSs.

We begin in Section 2 by introducing the relevant background and definitions. In Section 3 we analyse the expansion of existing KPSs, and in Section 4 we present alternative representations for KPSs and new constructions. Finally, we conclude in Section 5 and present some open questions.

2 Background

2.1 Key predistribution schemes

We consider the distribution of cryptographic keys to networks of small, resource-constrained devices, or 'nodes'. Typically, such nodes are scattered over a large area to perform basic tasks such as monitoring or data gathering. The limited storage and computational capacities of the nodes are an important consideration in the design of a scheme, in particular meaning that any security must be provided by symmetric key cryptography, which requires less memory and is less computationally expensive than public key cryptography. For the purposes of this paper, this simply means that if two nodes are to communicate securely, they will need to store the same cryptographic key, and we examine methods to achieve this.

A *key predistribution scheme* (KPS) is a method for allocating symmetric keys to the nodes of a network before they are deployed into their chosen environment. A major drawback of KPSs is that once the keys have been predistributed, subsequent key management operations are challenging to conduct [6]. See [16, 20, 47, 52] for surveys and details of the many different approaches to creating efficient schemes. Here, it suffices to say that in order to make best use of the nodes' limited resources, it is usually desirable to minimise the *key storage* requirement whilst maximising the *connectivity* and *resilience* of a network:

Key storage the number of keys which each device is required to store. This will usually be constant and denoted by k.

Connectivity the probability that a randomly-selected pair of nodes are 'connected', denoted Pr_1. This typically means sharing a key, although in some schemes a pair of nodes are required to share more than one key before they are allowed to communicate [18, 37, 39].

Resilience the probability that the set of keys used for communication between a pair of connected, uncompromised nodes is known to an adversary which has compromised s nodes, denoted fail_s for $1 \leq s \leq v-2$, where v is the number of nodes. This is a method for assessing the re-use of keys throughout a network. If each pair of nodes is connected by a unique key then $\mathsf{fail}_s = 0$ for all s, and we say the network has *perfect resilience*.

We now present a brief introduction to the relevant combinatorics and graph theory. The following definitions are widely accepted throughout the literature and were compiled with reference to [7, 10, 14, 21, 35, 51], to which we refer the interested reader for further details and examples.

2.2 Designs

The *power set* of a set $\mathscr{X}$ is the set of all subsets of $\mathscr{X}$, and is denoted $\mathscr{P}(\mathscr{X})$. A *set system (on $\mathscr{X}$)* is a pair $(\mathscr{X},\mathscr{B})$ where $\mathscr{X}$ is a set and $\mathscr{B} \subseteq \mathscr{P}(\mathscr{X})$. A *combinatorial design* (or, when the context is clear, a *design*) is a general term used to describe a set system with some specified conditions such as regularity, uniformity or set intersection, as we shall now explain.

In the context of combinatorial designs, the elements of the set $\mathscr{X}$ are called *points* and the elements of $\mathscr{B}$ are called *blocks*. The *degree* of a point $x \in \mathscr{X}$ is the number of blocks containing x. We say that $(\mathscr{X},\mathscr{B})$ is a *regular* design of degree r if every point has degree r. The *rank* is defined to be the size of the largest block. If all blocks have the same size, k, then the design is said to be *uniform* of rank k and is often called a *block design*. Note that we have defined the blocks as sets, so that each point occurs at most once in each block; we consider only *binary* designs in this paper.

We usually add a prefix to the word 'design' to specify the properties of the set system in question, for example, we define a $t-(v,k,\lambda)$ *design* to be a pair $(\mathscr{X},\mathscr{B})$ where $|\mathscr{X}| = v$, uniform of rank k, and every set of t points is contained in exactly λ blocks.

Combinatorial designs were first proposed for use in KPSs in [15]. A KPS can be constructed from a design by associating a key with each point, and a node with each block. That is, node N_j is given the set of keys $\{K_i : i \in B_j\}$, where B_j is a block in $\mathscr{B}$. For examples, see [47]. In Section 3 we will introduce some particular classes of designs which have been used to construct KPSs.

2.3 Graphs

A *graph* $G = (V,E)$ is a set of *vertices* $V = \{x_1, \ldots, x_v\}$ and a set of *edges* $E \subseteq V \times V$. We use the notation $(x_i, x_j) \in E$ to express that there is an edge between the vertices x_i and x_j, and we say that the edge (x_i, x_j) is *incident* to its endpoints x_i and x_j. Wherever an edge (x_i, x_j) exists, x_i and x_j are said to be *adjacent*.

Unless otherwise stated, graphs considered in this paper will be *simple graphs*, that is, they are *unweighted*, *undirected* and do not contain *self-loops* or *multiple edges*. These terms respectively mean that we do not assign different weights to vertices or edges, edges are not directed from one vertex to the other, there are no edges from a node to itself, and there is at most one edge between any two vertices.

Given subsets of vertices $X, Y \subset V$, the set of edges which connect X and Y is denoted

$$E(X,Y) = \{(x,y) : x \in X, y \in Y \text{ and } (x,y) \in E\} \ .$$

The *complement* $\overline{X}$ of X is the vertices which are not in X, that is, $\overline{X} = V \setminus X$.

An ordered set of consecutive edges $\{(x_{i1}, x_{i2}), (x_{i2}, x_{i3}), \ldots, (x_{i(p-1)}, x_{ip})\}$ in which all the vertices $x_{i1}, x_{i2}, \ldots, x_{ip}$ are distinct is called a *path* of length $p-1$. We say that a graph is *connected* if there is a path between every pair of vertices, and *complete* if there is an edge between every pair of vertices.

The *diameter* of a graph is the maximum 'distance' between pairs of vertices. That is, let $D(x_i, x_j)$ be the length of the shortest path between vertices x_i and x_j. Then the diameter of the graph is given by $\max_{x_i, x_j \in V} D(x_i, x_j)$. Finally, the *degree* $d(x_i)$ of a vertex x_i is the number of edges incident to that vertex. If all nodes have the same degree d, the graph is said to be *d-regular*.

It is common to represent KPSs using simple graphs. We draw a graph of a network by representing the nodes as vertices and the 'connections' as edges. That is, we associate each node N_i with a vertex x_i. From now on, we will refer to the vertex set using the notation $V = \{N_1, N_2, \ldots, N_v\}$.

To be precise in our analysis, we distinguish between the two possible types of 'connection' and consider the separate constituent graphs of a network: the *communication graph* $G_1 = (V, E_1)$ where $(N_i, N_j) \in E_1$ if nodes N_i and N_j are within communication range, and the *key graph* $G_2 = (V, E_2)$ where $(N_i, N_j) \in E_2$ if N_i and N_j share at least q common keys. We say that two nodes N_i and N_j can *communicate securely* if $(N_i, N_j) \in E_1 \cap E_2$, that is if they are adjacent in the *intersection* graph $G_1 \cap G_2 = (V, E_1 \cap E_2)$. If nodes are to be scattered such that there will be no control over the communication graph, the only way of affecting the intersection graph is through careful design of the key graph [38]. It is therefore the key graph which we design and analyse in this paper.

Finally, we define *hypergraphs* which can be considered as a generalisation of graphs, and which we introduce to the design and analysis of KPSs in Section 4.

Definition 1. A *hypergraph* $H = (V,E)$ is a set of vertices $V = \{N_1, \ldots, N_v\}$ and a set of *hyperedges* E. A hyperedge is a subset of V of cardinality ≥ 2, written as $(N_{i1}, \ldots, N_{ir}) \in E$, where $r \geq 2$. If every edge contains r vertices, we say that the hypergraph is r-uniform. Thus, a simple graph can be thought of as a 2-uniform hypergraph.

2.4 Expansion

For a thorough survey of expander graphs and their applications, see [25, 36, 38]. Here we introduce only the aspects of expander graphs which are relevant to our study, and in particular we restrict our attention to edge expansion and finite graphs.

Definition 2. A finite graph $G = (V,E)$ is an *ε-edge expander graph*, where the expansion coefficient ε is defined by

$$\varepsilon = \min_{S \subset V : |S| \leq \frac{v}{2}} \left(\frac{|E(S, \overline{S})|}{|S|} \right) \ .$$

There is a related definition of an ε-vertex expander graph, which is defined in a similar way using the vertex boundary of S. We have chosen to restrict our study here to edge expansion, as it is perhaps more intuitive for analysing key

sharing and is the measure used in the related literature on KPSs. Since we will be consistently referring to edge expansion properties, we will omit the word 'edge' when the context is clear, for ease of notation. Another name for the edge-expansion coefficient is the *isoperimetric number*, which is closely related to the *algebraic connectivity*, and in a weighted graph where every vertex has the same weight, ε is equivalent to the *Cheeger constant*; see [23] for further details.

Although the expansion coefficient ε is defined for any graph, the phrase 'expander graph' is used informally to refer to graphs with good expansion, that is, graphs with a large value of ε. A large value of ε is desirable for many network applications, which can be seen by the following observations:

- If $\varepsilon = 0$ then there exists a subset of vertices $S \subset V$ such that $E(S,\overline{S}) = \emptyset$. This implies that the graph is not connected. A graph is connected if and only if $\varepsilon > 0$.
- If ε is small, particularly if $\varepsilon < 1$, then there are sets of vertices which are connected to the rest of the graph by relatively few edges. In a network, this can lead to vulnerabilities such as: communication bottlenecks; uneven burdens on nodes, creating uneven battery drainage; a risk of being disconnected more easily by an adversary; longer average path lengths / diameter between unconnected nodes.
- If ε is larger, then there is no 'easy' way to disconnect large sets of nodes and there is a more even spread of communication burdens, battery usage and data flow. A graph with large ε will also have low diameter, logarithmic in the size of the network [36] and contain multiple short, disjoint paths between nodes [40], the many benefits of which were discussed in [38], where it is shown that $\varepsilon \leq \min_{x \in V} d(x)$.

Thus, key graphs with good expansion are particularly desirable for resource-constrained networks, and the papers by Çamtepe et al. [17] and Shafiei et al. [49] propose KPSs based on expander graph constructions. Before we analyse the expansion of KPSs in Section 3, we now explain how eigenvalues can be used to analyse the expansion of a graph or design. We present two related methods: finding the eigenvalues of the adjacency matrix, which provides a bound on the expansion using spectral gap, and finding the eigenvalues of the Laplacian, which provide information on related measures known in statistical design theory as A-, D- and E-optimality.

2.4.1 Spectral gap

For a d-regular graph G, we can define the *spectral expansion* of G using linear algebra. The following definitions are compiled with reference to [2, 23, 25, 36].

Definition 3. The *adjacency matrix* of a graph $G = (V,E)$ is defined to be a $|V| \times |V|$ matrix A, where each entry a_{ij} is the number of edges incident to both vertex i and vertex j.

It is easy to see that the adjacency matrix of a simple graph is a symmetric $0-1$ matrix, with zeroes on the main diagonal. For a d-regular graph, the sum of the entries in each row or column is d. Since A is a real symmetric $v \times v$ matrix, it has $|V| = v$ real eigenvalues, $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_v$, where all $\lambda_i \in [-d,d]$. In fact, it can be shown that $\lambda_1 = d$, corresponding to eigenvector u_1 whose entries are all $\frac{1}{v}$. We also note that $\lambda_v = -d$ if and only if G is bipartite (two-colourable). We can now define the spectral gap as follows.

Definition 4. Let $\widetilde{\lambda}$ be the largest eigenvalue in absolute value with $|\widetilde{\lambda}| \neq d$. Then the *spectral gap* of a d-regular graph G is defined to be $\lambda_1 - \widetilde{\lambda} = d - \widetilde{\lambda}$.

The various definitions of expansion are closely related. We have that

$$\frac{d-\widetilde{\lambda}}{2} \leq \varepsilon \leq \sqrt{2d(d-\widetilde{\lambda})}\ , \tag{1}$$

where ε is the edge expansion coefficient [1, 8, 19, 28, 29]. Thus, the spectral gap may be used as a measure of the *spectral expansion*: the larger the spectral gap, the better the expansion of G. Finally, all large d-regular graphs satisfy $\widetilde{\lambda} \geq 2\sqrt{d-1} - o(1)$ [46]. A graph is said to be Ramanujan if this bound is tight, that is, if $\widetilde{\lambda} \leq 2\sqrt{d-1}$, and therefore Ramanujan graphs have asymptotically smallest possible $\widetilde{\lambda}$, making them very good spectral expanders [25].

2.4.2 Laplacian eigenvalues

Definition 5. The *Laplacian matrix* of a graph $G = (V, E)$ is the $|V| \times |V|$ matrix whose (i, i) entry is the degree of vertex i, and (i, j) entry $(i \neq j)$ is the negative of the number of edges incident to both i and j. Thus it can be seen that the sum of the entries on each row is 0, and in a simple graph, the (i, j) entries are either 0 or -1.

The Laplacian of a d-regular graph (similarly, of an r-regular, k-uniform design with $d = k(r-1)$) is defined to be $L = dI - A$, where I is the identity matrix and A is the adjacency matrix (Definition 3).

The Laplacian and its eigenvalues can be used to determine various properties of a graph [4]. In particular, it is proven in [4] that the second smallest Laplacian eigenvalue of a connected graph is strictly positive. Also known as the *algebraic connectivity*, this eigenvalue corresponds to the second largest eigenvalue of the adjacency matrix of a regular graph or design, and is therefore the eigenvalue of interest in the calculation of the spectral gap (Definition 4) for regular graphs and designs. We will return to this method of analysing the expansion or 'optimality' of a design in Section 4.3.

3 Analysing the expansion of existing KPSs

Having observed in Section 2.4 that it is desirable for the key graph of a KPS to have a large expansion coefficient, we will now analyse the expansion of various KPS constructions. We begin with those based on expander graph constructions.

3.1 KPSs made from expander graph constructions

Çamtepe et al. [17] and Shafiei et al. [49] propose KPSs based on expander graph constructions. Çamtepe et al. [17] use a construction for a Ramanujan graph, which, as we saw in Section 2.4.1, is an asymptotically optimal spectral expander graph. The construction they use is for network size $v = p + 1$ and key storage $k = q + 1$, where p and q are primes congruent to 1 mod 4 (see [36]). Shafiei et al. [49] use the zig-zag construction for an expander graph, which has the benefit of being more flexible to produce key graphs for any sizes of v and k. Both papers use the following method:

1. construct an expander graph G for the appropriate network size and degree (and, in the case of [17], remove any self-loops or multiple edges. The authors suggest that these be replaced with randomly-selected edges such that all nodes have the same degree, though they omit this step from their example);
2. assign a unique pairwise key to every edge of G;
3. preload each node with the set of keys which correspond to its set of edges.

The key graph then has good expansion. However, we claim that it is possible to achieve higher expansion in a KPS for the same network size and key storage, as we will demonstrate in the sections that follow.

3.2 Expansion of random KPSs

Random KPSs have been proposed in [32, 18], where keys are allocated to each node at random from a given key pool. The connectivity and resilience depend upon the size of the key pool and the key storage per node, and, in the case of the q-composite scheme in [18], an intersection threshold $q > 1$ such that nodes are only connected if they share at least q keys.

Connected Erdös-Rényi random graphs $G(v, p)$ [31] are good expanders with high probability [36]. As noted in [6, 27, 53], the random KPS of Eschenauer and Gligor [32] produces a key graph which is more highly connected than the Erdös-Rényi graph $G(v, \mathsf{Pr}_1)$. Therefore random KPSs produce key graphs with good expansion with high probability.

3.3 Expansion of combinatorial designs

Many deterministic KPS constructions are based on combinatorial designs; see [47] for a unifying survey. We will now analyse a subset of these constructions and prove lower bounds on the expansion of their corresponding key graphs. Many of the designs proposed for use in KPSs have the property of being *configurations*.

Definition 6. (from [44, Definition 1.2]) A design $(\mathscr{X},\mathscr{B})$ with $|\mathscr{X}| = n$, $|\mathscr{B}| = v$ is called a (n,v,r,k)-*configuration* if it is regular of degree r, uniform of rank k and any two points occur in at most one block.

More information on configurations can be found in [24, 44]. We note here that the key graph of a configuration is regular:

Lemma 1. *(from [44, Lemma 1.1]) The key graph of a (n,v,r,k)-configuration is regular of degree $k(r-1)$. This is the maximum possible degree of a graph of a (n,v,r,k)-design.*

We restrict our study of the expansion of designs to configurations, where the regularity of the key graph will simplify some of the analysis. First we provide an estimate of the expansion of a regular, uniform configuration.

Lemma 2. *For an (n,v,r,k)-configuration, the edge-expansion coefficient can be estimated by*

$$\varepsilon \approx \frac{k(r-1)}{2} \quad .$$

Proof. In the key graph of an (n,v,r,k)-configuration, the degree of a node is $k(r-1)$ (Lemma 1). Thus, for a set of nodes $S \subset V$,

$$E(S,\overline{S}) = |S|k(r-1) - 2E(S,S),$$

where $E(S,S)$ counts the number of edges whose endpoints are both in S. The factor of two is needed because the '$|S|k(r-1)$' term has counted each of these edges twice.

We now need to find an expression for $E(S,S)$. Let N_a and N_b be nodes in S, and let $\mathscr{K}_a$ and $\mathscr{K}_b$ be their respective key sets, each of size k. For simplicity of notation, label the keys in $\mathscr{K}_a$ as $K_1,\ldots,K_k$ and define random variables X_i for $1 \le i \le k$ so that

$$X_i = \begin{cases} 1 & \text{if key } K_i \in \mathscr{K}_b \\ 0 & \text{otherwise} \end{cases} \quad .$$

Then $X := \sum_{i=1}^{k} X_i$ counts the number of edges between N_a and N_b.

The expected value of X_i, which we will write as $Ex[X_i]$ to avoid confusion with the edge notation, is given by

$$Ex[X_i] = \frac{r-1}{v-1} \quad ,$$

since there are $r-1$ other nodes which know key K_i, out of a total of $v-1$ other nodes in the network. By linearity of expectation,

$$Ex[X] = kEx[X_i] = \frac{k(r-1)}{v-1} \quad .$$

Thus the expected number of edges amongst $|S|$ nodes is $\binom{|S|}{2}\frac{k(r-1)}{v-1}$, hence

$$\begin{aligned} Ex[E(S,\overline{S})] &= |S|k(r-1) - 2\binom{|S|}{2}\frac{k(r-1)}{v-1} \\ &= |S|k(r-1)\left[1 - \frac{|S|-1}{v-1}\right] \quad . \end{aligned}$$

Finally, since $\varepsilon = \min\limits_{1\le|S|\le\frac{v}{2}} \dfrac{|E(S,\overline{S})|}{|S|}$, this gives

$$\begin{aligned} \varepsilon &\approx \min_{1\le|S|\le\frac{v}{2}} \left\{ k(r-1)\left[1 - \frac{|S|-1}{v-1}\right] \right\} \\ &= k(r-1)\left[1 - \frac{\lfloor\frac{v}{2}\rfloor - 1}{v-1}\right] \\ &\approx \frac{k(r-1)}{2} \end{aligned}$$

for large v. □

Lemma 2 tells us that the *expected value* of the expansion coefficient of a configuration-based KPS is good, since in particular $\frac{k(r-1)}{2} > 1$ for practical values of k and r. However, Lemma 2 does not guarantee good expansion: consider for example the design $(\mathscr{X},\mathscr{B})$ where $\mathscr{X} = \{1,2,3,4,5,6\}$, $\mathscr{B} = \{\{1,2\},\{2,3\},\{1,3\},\{4,5\},\{5,6\},\{4,6\}\}$ which is a $(6,6,2,2)$-configuration but is disconnected and has expansion $\varepsilon = 0$.

To successfully construct a KPS from a configuration, we clearly require the graph of the configuration to be connected. In addition to being k-uniform and r-regular, configurations which are proposed as constructions for KPSs generally have further properties which guarantee connectedness. We demonstrate examples of these properties in Sections 3.3.1 and 3.3.2, where we give a brief overview of two classes of configurations which have been proposed as constructions for KPSs, namely μ-common intersection designs and strongly regular graphs, and we prove lower bounds for their expansion parameters.

3.3.1 μ-common intersection designs

A class of configurations called μ-common intersection designs were defined by Lee and Stinson in [43, 44].

Definition 7. Let $(\mathscr{X},\mathscr{B})$ be a (n,v,r,k)-configuration. We say that $(\mathscr{X},\mathscr{B})$ is a *μ-common intersection design* if for blocks B_i and B_j,

$$B_i \cap B_j \neq \emptyset \quad \Longrightarrow \quad |\left\{B_k \in \mathscr{B} : B_i \cap B_k \neq \emptyset \text{ and } B_j \cap B_k \neq \emptyset\right\}| \geq \mu.$$

In terms of the key graph of a KPS, this means that if nodes N_i and N_j corresponding to blocks B_i and B_j do not share any keys and so are not adjacent, then they have at least μ common neighbours, i.e. μ nodes with which they both share a key.

In [44], the motivation given for using μ-common intersection designs for KPSs is that if two nodes N_i and N_j wish to communicate but do not share a common key, then they can communicate via 'two hops' if they share at least one common neighbour. It is widely recognised that common neighbours can be beneficial for updating, 'reinforcing' and establishing new keys, and routing messages between nodes which do not share a common key [18, 26, 30]. We now show that having diameter 2 provides a good lower bound on the expansion of a graph, providing further support for the use of KPSs with 'two hop paths'.

Lemma 3. *Let $G = (V,E)$ be a graph with diameter 2. Then for any subset S, where $\emptyset \neq S \subset V$,*

$$|E(S,\overline{S})| \geq \min\{|S||\overline{S}|\}.$$

Proof. Without loss of generality, suppose that $|S| \leq |\overline{S}|$. Now, suppose for a contradiction that $|E(S,\overline{S})| < \min\{|S||\overline{S}|\}$, that is, suppose $|E(S,\overline{S})| < |S|$. Then there exists a node $N_i \in S$ which is not adjacent to any node in $\overline{S}$. Denote the set of nodes adjacent to N_i by V_{N_i}. We have that $V_{N_i} \subseteq S$, and so $E(V_{N_i},\overline{S}) \subseteq E(S,\overline{S})$. Thus $|E(V_{N_i},\overline{S})| \leq |E(S,\overline{S})| < |S| \leq |\overline{S}|$, and so there exists a node $N_j \in \overline{S}$ which is not adjacent to any node in V_{N_i}. This contradicts the property that the graph has diameter 2, since N_i and N_j do not have a common neighbour. □

Corollary 1. *The graph $G = (V,E)$ of a μ-common intersection design is an ε-expander graph, where $\varepsilon \geq 1$.*

Proof. Since the graph of a μ-common intersection design has diameter 2, this is a simple consequence of Lemma 3. By definition, the expansion coefficient is given by

$$\begin{aligned} \varepsilon &= \min_{S \subset V : |S| \leq \frac{v}{2}} \left\{ \frac{|E(S,\overline{S})|}{|S|} \right\} \\ &\geq \min_{S \subset V : |S| \leq \frac{v}{2}} \left\{ \frac{|S|}{|S|} \right\} \\ &\geq 1. \end{aligned}$$

□

Therefore we have shown that μ-common intersection designs are a natural choice for KPSs, not only because of the 'two-hop paths' property mentioned in [44], but also because they have expansion of at least $\varepsilon = 1$.

3.3.2 Strongly regular graphs

Strongly regular graphs may be regarded as a special type of μ-common intersection design, and are defined as follows.

Definition 8. (from [13]) A $(v,k(r-1),\lambda,\mu)$-*strongly regular graph* is a graph on v vertices which is regular of degree $k(r-1)$ and has the following properties:

- any two adjacent vertices have exactly λ common neighbours
- any two nonadjacent vertices have exactly μ common neighbours.

Equivalently, the design $(\mathscr{X},\mathscr{B})$ is a *strongly regular graph* if it is regular of degree r, uniform of rank k, and for blocks B_i and B_j,

$$B_i \cap B_j \neq \emptyset \quad \Longrightarrow \quad |\left\{B_k \in \mathscr{B} : B_i \cap B_k \neq \emptyset \text{ and } B_j \cap B_k \neq \emptyset\right\}| = \lambda$$

and

$$B_i \cap B_j = \emptyset \quad \Longrightarrow \quad |\left\{B_k \in \mathscr{B} : B_i \cap B_k \neq \emptyset \text{ and } B_j \cap B_k \neq \emptyset\right\}| = \mu \ .$$

Strongly regular graphs have been shown to exist for various combinations of the parameters v, k and r in [42]. Constructions are given in [13, 50, 44] and we refer the reader to [12] for a discussion on constructing random strongly regular graphs.

In addition to Lemma 3 we have another lower bound for strongly regular graphs:

Lemma 4. *For a connected* $(v,k(r-1),\lambda,\mu)$ *strongly regular graph,*

$$\varepsilon \geq \frac{k(r-1)}{2} - \frac{\lambda-\mu+\sqrt{(\lambda-\mu)^2+4(k(r-1)-\mu)}}{4} \ .$$

Proof. In [11] it is shown that the non-trivial eigenvalues of a strongly regular graph are the solutions of the equation

$$x^2-(\lambda-\mu)x+(k(r-1)-\mu)=0 \ .$$

Thus the larger root is given by

$$\widetilde{\lambda} = \frac{\lambda-\mu+\sqrt{(\lambda-\mu)^2+4(k(r-1)-\mu)}}{2}$$

and, using Equation (1), we have our result. □

In Section 4.2.2 we will see an example of a strongly regular graph as a KPS, and we will use Lemma 4 to show that its expansion is in fact significantly greater than 1.

4 Studying, constructing and analysing KPSs using alternative graph methods

In this section we discuss alternative graphical representations of the key graph of a KPS. We argue that these representations provide important benefits, namely:

1. the incidence graph enables us to recover all the details of a KPS;
2. the incidence graph suggests new approaches for constructing KPSs;
3. the concurrence graph and incidence graph together enable us to perform further analysis using statistical design theory.

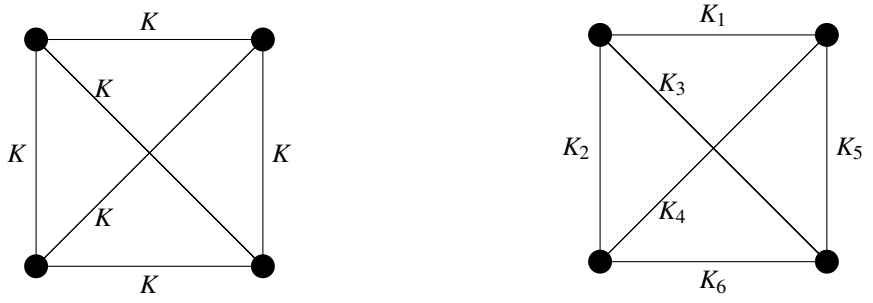
We consider these benefits in Sections 4.1, 4.2 and 4.3 respectively.

4.1 Representation of a KPS using the incidence graph

It is common in the literature to represent the key graph of a KPS in the way described in Section 2.3, where vertices correspond to nodes and edges correspond to shared keys. From now on we will call this the *block intersection graph* of a KPS, corresponding to the common terminology in combinatorial design theory. As far as we are aware, no other graphical representions have been used for KPSs. We suggest that representing a KPS using an incidence graph can have advantages over the block intersection graph representation, namely by demonstrating the key storage and resilience of the KPS in addition to the connectivity, as we now explain with reference to two motivating examples.

4.1.1 Example 1: trivial KPSs

Consider Figure 1, which shows (with labels) the block intersection graphs of two trivial KPSs on four nodes. Figure 1(a) represents a KPS where every node stores a single key, K, and Figure 1(b) represents a KPS where each pair of nodes is assigned a unique key. Notice that both graphs are complete, despite the difference in their resilience. Without the edge labels (which would be infeasible to draw on a large graph) the graphs would be isomorphic, and thus the important metrics of key storage and resilience are not be represented.



(a) Each node is assigned a single key, K

(b) Every pair of nodes is assigned a unique key

Figure 1: Trivial KPSs represented by block intersection graphs

(a) Each node is assigned a single key, K

(b) Every pair of nodes is assigned a unique key

Figure 2: Incidence graph representations of trivial key predistribution schemes

In Figure 2 we represent the same two KPSs in a different way. Nodes are again represented as black vertices, but we now introduce white vertices which correspond to keys. There is an edge between a key vertex K_i and node vertex v_j if and only if the key K_i is known to node v_j. This representation can be thought of in numerous ways, including the incidence or Levi graph of a design [45], often referred to as a bipartite graph on the node and key vertices, and/or a hypergraph, as we will explain in more detail in Section 4.2.

We observe that the incidence graph representation in Figure 2 demonstrates the key storage k and the number of nodes which store each key (or 'key repetition') r: we see from the degrees of the node vertices that $k = 1$ in Figure 2(a) and $k = 3$ in Figure 2(b), and from the degrees of the key vertices that $r = 4$ in Figure 2(a) and $r = 2$ in Figure 2(b)), which immediately tells us that the resilience is worst possible in Figure 2(a) and best possible in Figure 2(b).

4.1.2 Example 2: a KPS from a configuration

We now consider another, less extreme example. Consider the $2-(9,3,1)$ configuration: $\mathscr{X} = \{1,2,3,4,5,6,7,8,9\}$ and $\mathscr{B} = \{\{123\}, \{456\}, \{789\}, \{147\}, \{258\}, \{369\}, \{159\}, \{267\}, \{348\}, \{168\}, \{249\}, \{357\}\}$, which can be considered as a KPS where points correspond to keys and each block corresponds to a node's key set. The block intersection graph of this KPS (with key labels) is given in Figure 3. For comparison, Figures 4(a) and 4(b) show two isomorphic representations of the incidence graph. These representations respectively appeal to the hypergraph and bipartite ways of thinking about the incidence graph.

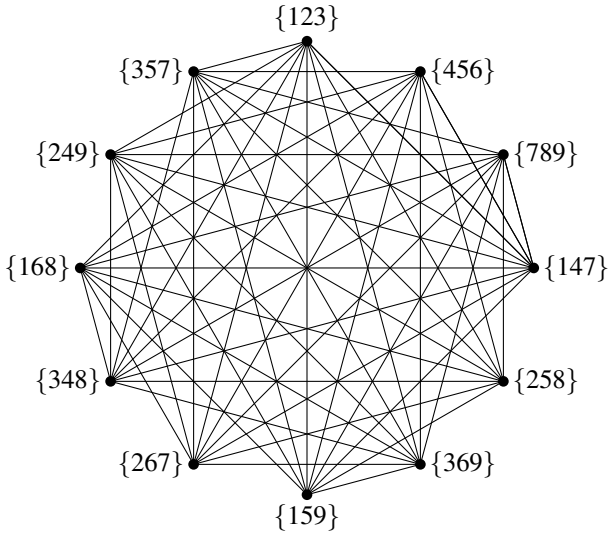


Figure 3: Block intersection graph of a $2-(9,3,1)$ configuration

Figure 3 clearly demonstrates the connectivity: each node is connected to nine other nodes, hence $\mathsf{Pr}_1 = \frac{9}{11} \approx 0.818$. However, without the key labels it would not be possible to determine k and r with certainty. By contrast, Figures 4(a) and 4(b) demonstrate clearly that $k = 3$ (node vertex degree) and $r = 4$ (key vertex degree / hyperedges uniformly incident to four vertices). A simple calculation of $k(r-1) = 3 \times 3 = 9$ tells us that each node shares a key with nine other nodes, and hence the connectivity of the KPS is $\mathsf{Pr}_1 = \frac{9}{11}$. Key labels are included in Figure 4 for clarity, but notice that they are not needed for these calculations.

Finally, we note that it is straightforward to produce the incidence graph representation of a combinatorial design via its *incidence matrix*, as each row corresponds to a key (hyperedge):

Definition 9. The *incidence matrix M* of a design is a matrix where the columns represent blocks (nodes), rows represent points (keys), and whose entries are given by $m_{ij} = \begin{cases} 1 & \text{if point } i \text{ is in block } j \\ 0 & \text{otherwise} \end{cases}$.

In summary, without key labels, a block intersection graph only provides an exact representation of the connectivity of a KPS, whereas the incidence graph, which is easy to construct, unambiguously represents the connectivity, key storage and resilience:

1. the degree of a node vertex is equal to the number of keys stored by that node, k;
2. the degree of a key vertex, r, demonstrates the number of nodes which store that key, allowing us to calculate the resilience;
3. the connectivity is given by $\frac{k(r-1)}{v-1}$.

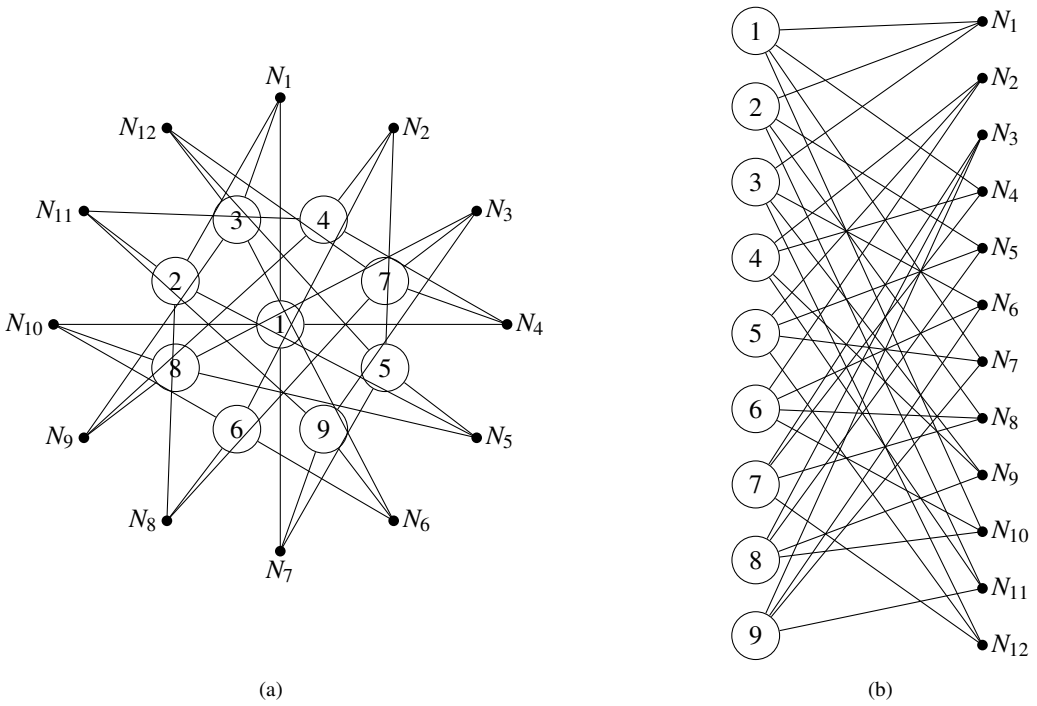


Figure 4: The incidence graph of a $2-(9,3,1)$ design; two isomorphic representations with different arrangements of the vertices

4.2 Incidence graph constructions for KPSs

In addition to providing a helpful visual representation of a key graph, the incidence graph can help us to consider further constructions for KPSs. That is, it helps to illustrate that one can consider a KPS both as an assignment of keys to nodes, and as an assignment of nodes to keys. We now consider the incidence graph as a hypergraph and show how this perspective leads to further KPS constructions.

4.2.1 Hypergraphs

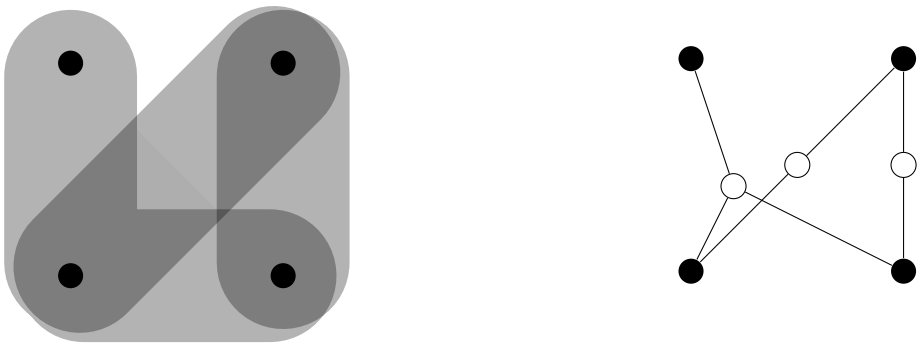
Hypergraphs are a generalisation of graphs, where each edge may be incident to more than two vertices. There are various equivalent ways to draw hyperedges: Figure 5(a) demonstrates perhaps the more intuitive method, where a hyperedge 'contains' its vertices; Figure 5(b) uses the convention of drawing each hyperedge as a new (white) vertex, adjacent to its set of incident (black) vertices. The latter method is equivalent to the incidence graph method given in Section 4.1, where hyperedges correspond to keys and black vertices correspond to nodes.

Notice that, since a (hyper)graph $G=(V,E)$ is defined by its vertex set V and (hyper)edge set E, a (hyper)graph is a set system. An r-uniform hypergraph is a set system which is uniform of rank r. Further, as we now demonstrate, the hypergraph representation of a combinatorial design can be a regular, uniform design itself. Indeed, the hypergraph $H=(V,E)$ from Figure 4(a) is given by $V=\{1,2,\ldots,12\}$, and

$$\begin{aligned} E &= \{\{1,4,7,10\},\{1,5,8,11\},\{1,6,9,12\},\{2,4,9,11\},\{2,5,7,12\},\\ &\quad \{2,6,8,10\},\{3,4,8,12\},\{3,5,9,10\},\{3,6,7,11\}\}. \end{aligned}$$

Regarding $H=(V,E)$ as a combinatorial design, that is, regarding the hyperedges as blocks, we see that this design is regular of degree 3, uniform of rank 4, and has incidence matrix M^t, the transpose of the incidence matrix M of the original $2-(9,3,1)$ configuration.

We propose the use of hypergraphs for constructing KPSs by making the following observation. The KPSs given in [17, 49] have perfect resilience and good expansion but rather low connectivity for the key storage. If their methods



(a) (b)

Figure 5: Graph and hypergraph representations of a simple KPS

could be adapted so that they did not produce KPSs with perfect resilience, they might provide further good constructions for KPSs with trade-offs between all three parameters. A naïve approach to such an adaptation would be to take an expander graph construction for a KPS and simply assign the same key to multiple edges. However, it is not entirely clear how best to do this. One way would be to create the expander graph as in [17] or [49], pick a node at random, and assign a key K_1 to two of its edges, repeat for a key K_2, etc. However, such an approach could lead to unnecessarily high key storage for some nodes, whilst others receive comparatively few keys. In addition, an expander graph with higher connectivity than in [17, 49] should be picked in the first place, as otherwise we would be reducing the resilience whilst maintaining the same connectivity. It is not immediately obvious how many edges the initial expander construction should have in order to maintain a similar magnitude of key storage and maintain a desirable trade-off between connectivity and resilience. Therefore, whilst this method may have merit, we demonstrate a 'tidier' solution: we construct KPSs from hypergraphs with good expansion, thereby creating KPSs with good expansion where each key is known to $r > 2$ nodes.

4.2.2 Hypergraph KPS constructions

The literature on hypergraphs with good expansion (sometimes called 'expanding hypergraphs') is limited. In [34], the notion of eigenvalues of a graph is extended to hypergraphs, and thus Friedman et al. are able to analyse the second eigenvalue of a hypergraph. This allows them to carry over the concept from graph theory of a spectral expander graph being one with large spectral gap (Section 2.4.1). They prove many theorems about the properties of such expanding hypergraphs, and provide a construction, namely Cayley hypergraphs. Whilst they demonstrate that Cayley hypergraphs have good expansion, they also note that the expansion of a general Cayley hypergraph is far from the optimal bounds which are proven in the paper.

There is currently no known method for generating uniform hypergraphs at random [36]. If there were, they would likely provide a very promising way of constructing KPSs, since it is proven in [34] that a random uniform hypergraph has good expansion with high probability.

In [48] two further approaches to constructions of hypergraphs with good expansion are given: explicit constructions from Gower's Theorem, and semi-explicit constructions from Fourier Analysis. However, in order to provide a simple demonstration of the method of constructing a KPS from a hypergraph, we will use the Cayley hypergraph construction from [33] which has good (though far from optimal) expansion, and we will compare its effectiveness as a construction for a KPS to the construction from [17] based on a Ramanujan expander graph.

Definition 10. Let V be a group, and $W \subset V$. The 3-*uniform Cayley hypergraph on V and W* is the hypergraph whose vertices are the elements of V and whose hyperedge set is given by

$$E = \{(x,y,z) : x,y,z \text{ distinct}, xyz \in W\}.$$

Remark 1. We note that, to the best of our knowledge, the earliest definition of a Cayley hypergraph is given in [33], as an extension of the widely-known definition of a Cayley graph (or 2-hypergraph). It is stated as the definition of a '3-regular' Cayley hypergraph on V and W. However, the word 'regular' is more commonly used to mean that the

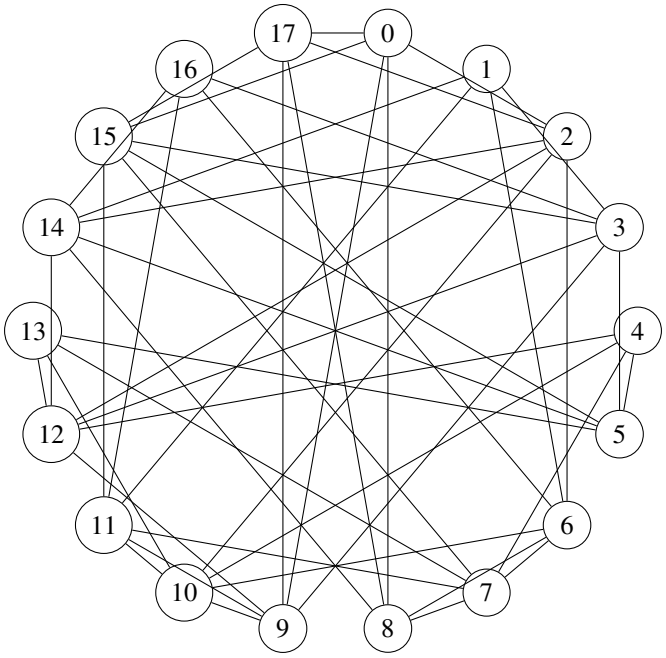


Figure 6: KPS from Ramanujan expander graph construction, from [17]

number of (hyper)edges incident to each node is constant. This is not necessarily the case in Cayley hypergraphs (as we will demonstrate below) and indeed there will generally be more than three hyperedges incident to each node. We therefore use the word 'uniform' for consistency with the recent literature.

In addition, for the hypergraph to be 3-uniform we have required that $x \neq y \neq z \neq x$. Otherwise, hyperedges of the form (x,x,z) and (x,x,x) would be included, which are not usually considered to be 3-edges. This is not stated in the definition in [33].

In the example given in [17], a KPS is created for 18 nodes from a Ramanujan expander graph. We reproduce their key graph in Figure 6. Each node stores 4, 5 or 6 keys, and there are 46 keys used in total, each securing exactly one link. Thus we have $\mathsf{fail}_s = 0$ for all $s \leq 16$ and $\mathsf{Pr}_1 = \frac{46}{\binom{18}{2}} = \frac{46}{153} \approx 0.3$.

For a direct comparison, we construct a KPS for 18 nodes using a 3-uniform Cayley hypergraph. The Cayley hypergraph $H = (V,E)$ where $V = \{0,1,\ldots,17\}$ and

$$E = \{(x,y,z) : x,y,z \text{ distinct}, x+y+z \equiv 0 \mod 18\}$$

also has 46 hyperedges (keys), but each hyperedge connects three nodes. Thus there are $46 \times 3 = 138$ 'pairs' of links, and so

$$\mathsf{Pr}_1 = \frac{138}{\binom{18}{2}} = \approx 0.902 \ ; \tag{2}$$

we have tripled the connectivity. This has not been at the cost of a large reduction in resilience:

$$\mathsf{fail}_1 = \frac{1}{3}\frac{7}{124} + \frac{2}{3}\frac{8}{122} \approx 0.0625 \tag{3}$$

because:

- One third of the nodes store 7 keys. On compromising one of these nodes, the adversary learns 7 keys. There are then $138 - (7 \times 2)$ uncompromised links remaining, of which 7 use a key known to the adversary.
- Similarly, two thirds of the nodes store 8 keys, and compromising one of these leaves $138 - (8 \times 2)$ uncompromised links, of which 8 are vulnerable.

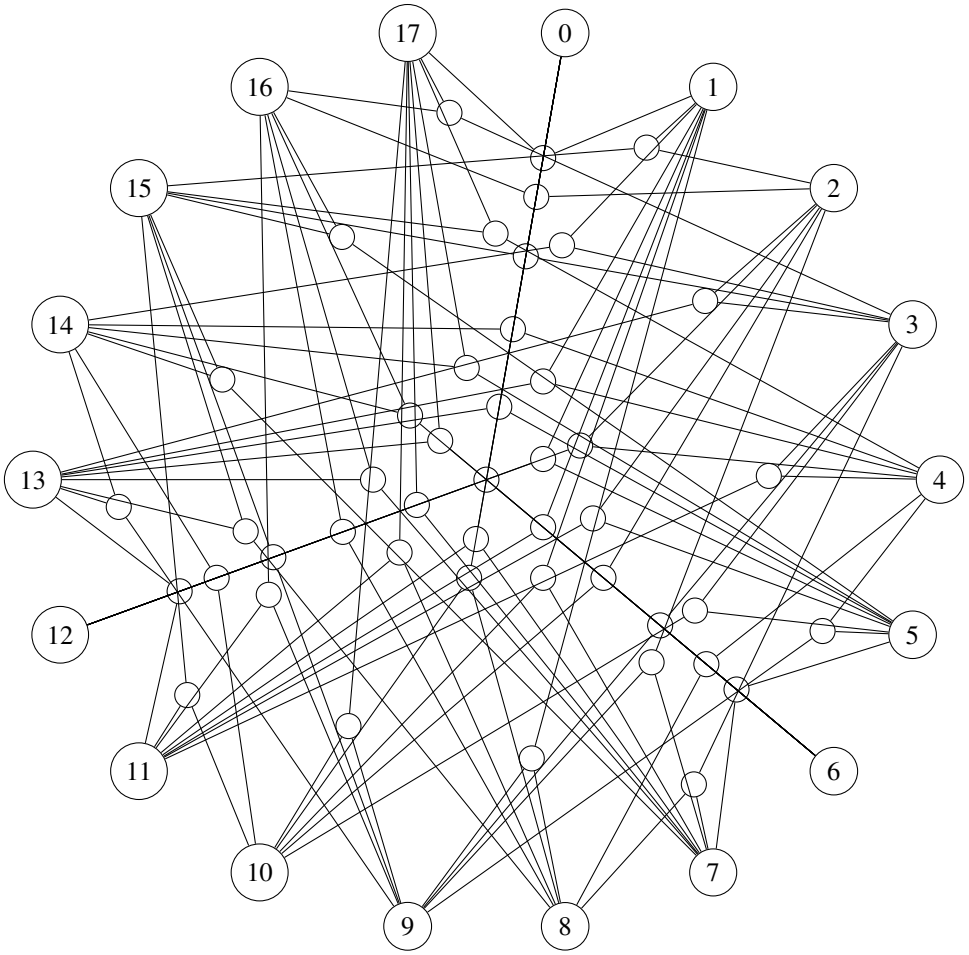


Figure 7: Cayley hypergraph

We will provide a general formula for fail_s and prove that it has expansion $\varepsilon = 8$ at the end of this section, after making a simplifying assumption.

We also note that there is a more equal spread of key storage per node in this construction than in the Ramanujan construction from [17], that is, the key storage for node N_i is $k_{N_i} \in \{7,8\}$, whereas in the Ramanujan construction $k_{N_i} \in \{4,5,6\}$. For large networks the Cayley construction does not scale well: the key storage is $\approx \frac{v-1}{2}$. Nevertheless, it serves as a simple example of the use of a hypergraph to construct a KPS with practical trade-offs between connectivity and resilience, as well as the benefits of good expansion.

But of course this is not entirely surprising. A hypergraph is a set system, and an r-uniform, k-regular hypergraph $H = (V,E)$ can be regarded as the representation of a k-uniform, r-regular design, whose incidence matrix is the transpose of the incidence matrix of H. Thus, our construction bears a strong resemblance to a combinatorial design, and the effectiveness of designs as constructions for KPSs is well established.

To recover the set system to which our Cayley hypergraph on 18 nodes corresponds, we generate the 46×18 incidence matrix M, where each row corresponds to a hyperedge and each column corresponds to a key. Thus, reading down each column gives the key set of each node, and we recover the set system $(\mathscr{X}, \mathscr{B})$ where $\mathscr{X} = \{01, 02, \ldots, 46\}$

and the blocks of $\mathscr{B}$ are:

$$\begin{array}{lll}
\{01,02,03,04,05,06,07,08\}, & \{01,09,10,11,12,13,14,15\}, & \{02,09,16,17,18,19,20\},\\
\{03,10,16,21,22,23,24,25\}, & \{04,11,17,21,26,27,28\}, & \{05,12,18,22,26,29,30,31\},\\
\{06,13,19,23,27,29,32,33\}, & \{07,14,20,24,29,34,35,36\}, & \{08,15,24,27,37,38,39\},\\
\{15,20,23,26,40,41,42,43\}, & \{08,14,19,22,40,44,45\}, & \{07,13,18,21,37,41,44,46\},\\
\{06,12,17,34,38,42,45,46\}, & \{05,11,16,32,35,39,43,46\}, & \{04,10,30,33,36,43,45\},\\
\{03,09,28,31,36,39,42,44\}, & \{02,25,31,33,35,38,41\}, & \{01,25,28,30,32,34,37,40\}.
\end{array}$$

Then $(\mathscr{X},\mathscr{B})$ is a design, regular of degree three, with rank eight. Notice that it is not uniform, as some blocks have seven points. However, if we simply add the hyperedges $\{2,8,14\}$ and $\{4,10,16\}$ to our Cayley hypergraph, we have a 3-uniform, 8-regular hypergraph, corresponding to an 8-uniform, 3-regular design. (These additional hyperedges are picked using the six nodes which have degree seven, and assigning hyperedges so that no pair of nodes is incident to more than one hyperedge.) Explicitly, we would have the design $(\mathscr{X},\mathscr{B})$ where $\mathscr{X} = \{01,02,\ldots,48\}$ and the blocks of $\mathscr{B}$ are:

$$\begin{array}{lll}
\{01,02,03,04,05,06,07,08\}, & \{01,09,10,11,12,13,14,15\}, & \{02,09,16,17,18,19,20,47\},\\
\{03,10,16,21,22,23,24,25\}, & \{04,11,17,21,26,27,28,48\}, & \{05,12,18,22,26,29,30,31\},\\
\{06,13,19,23,27,29,32,33\}, & \{07,14,20,24,29,34,35,36\}, & \{08,15,24,27,37,38,39,47\},\\
\{15,20,23,26,40,41,42,43\}, & \{08,14,19,22,40,44,45,48\}, & \{07,13,18,21,37,41,44,46\},\\
\{06,12,17,34,38,42,45,46\}, & \{05,11,16,32,35,39,43,46\}, & \{04,10,30,33,36,43,45,47\},\\
\{03,09,28,31,36,39,42,44\}, & \{02,25,31,33,35,38,41,48\}, & \{01,25,28,30,32,34,37,40\}.
\end{array}$$

Notice that this is not a $2-(48,8,1)$ design, since, for example, it can easily be checked that points 1 and 16 do not appear in the same block. However, it is a configuration (Definition 6), and for each block B_i there is exactly one block B_j with which it does not share any keys, i.e. $B_i \cap B_j = \emptyset$. Thus, B_i and B_j have 16 common neighbours, and so this is a 16-common intersection design. Further, for every B_i and B_j with $B_i \cap B_j \neq \emptyset$, there are $\lambda = 14$ common neighbours, and thus this is a $(18,16,14,16)$ strongly regular graph. We have already established that strongly regular graphs make good constructions for KPSs in Section 3.3.2, and in particular Lemma 4 gives

$$\varepsilon \geq 8 - \frac{(-2)+\sqrt{(-2)^2+4\times 0}}{4} = 8 \ .$$

By inspection we can find a subset of vertices S with $|S| = 9$ and $|E(S,\overline{S})| = 72$, giving $\varepsilon \leq 8$, hence $\varepsilon = 8$ which shows that our proposed KPS construction does have good expansion.

Indeed, it is shown in [9] that if there exists a $(v,k(r-1),\lambda,\mu)$-strongly regular graph for a given set of parameters v, k, r, λ, μ, and if

$$v \leq 2k(r-1) - \lambda$$

and

$$\lambda \geq \frac{1}{2}\left(\lambda - \mu + \sqrt{(\lambda-\mu)^2 + 4(k(r-1)-\mu)}\right),$$

then that strongly regular graph maximizes the algebraic connectivity, hence ε, amongst all $k(r-1)$-regular graphs. In our example we have $v = 18 = 2\times 8\times 2 - 14$ and $\lambda = 14 > 0$, and so we can conclude that our example has optimal expansion amongst all 16-regular graphs.

Finally, we find the connectivity and resilience of this strongly regular graph KPS. By observation,

$$\mathsf{Pr}_1 = \frac{144}{\binom{18}{2}} \approx 0.941, \tag{4}$$

and

$$\mathsf{fail}_1 = \frac{8}{128} = \frac{1}{16} = 0.0625. \tag{5}$$

Thus, by adding the extra two edges, the connectivity has increased from approximately 0.902 to 0.941 (Equations (2) and Equation (4)), and the resilience has fractionally improved: the difference between Equation (3) and Equation (5) is ≈ 0.000033.

Calculating fail_2 using conditional probabilities, we find that

$$\mathsf{fail}_2 = \frac{1}{17}\left(\frac{8}{128}+\frac{8}{112}\right)+\frac{16}{17}\left(\frac{8}{128}+\frac{7}{113}\right) = 0.125005 \tag{6}$$

because: suppose the adversary has compromised one node, which without loss of generality we will call N_1, and so the adversary has learned key set $\mathscr{K}_{N_1}$. Now there is one node N_i out of the remaining 17 nodes for which $\mathscr{K}_{N_1} \bigcap \mathscr{K}_{N_i} = \emptyset$, that is, one node which would reveal 8 further keys to the adversary. Compromising any of the other 16 nodes reveals exactly 7 new keys, as $|\mathscr{K}_{N_1} \bigcap \mathscr{K}_{N_j}| = 1$ for all $2 \leq j \leq 18$, $j \neq i$.

In [43] the following formula is given as an estimate for the resilience of a KPS composed of l copies of a strongly regular graph:

$$\mathsf{fail}_s = 1 - \frac{\binom{v-l-2}{s}}{\binom{v-2}{s}} \quad . \tag{7}$$

Table 1 gives the exact values of Equation (7) for our extended Cayley hypergraph ($l = 1$) for $s = 1, \ldots, 15$, and we see that it agrees exactly with Equation (5) and approximately with Equation (6).

s	1	2	3	4	5	6	7	8
fail_s	0.0625	0.125	0.1875	0.25	0.3125	0.375	0.4375	0.5

s	9	10	11	12	13	14	15
fail_s	0.5625	0.625	0.6875	0.75	0.8125	0.875	0.9375

Table 1: Resilience of the extended Cayley hypergraph

Table 1 shows that fail_s increases steadily with s. In particular, $\mathsf{fail}_s = 0.5$ when $s = \frac{v}{2} - 1$. As a point of comparison, the resilience of an Eschenauer Gligor scheme [18, 37] is given by

$$\mathsf{fail}_s = 1 - \left(1 - \frac{k}{n}\right)^s \quad .$$

Thus an Eschenauer Gligor scheme which also has $\mathsf{fail}_1 = 0.0625$ has parameters $\frac{k}{n} = \frac{1}{16}$, and this proportion gives $\mathsf{fail}_s > 0.5$ for $s \geq 11$. Indeed, for each s, the Eschenauer Gligor value of fail_s with $\frac{k}{n} = \frac{1}{16}$ is smaller than the approximate value of fail_s for the extended Cayley hypergraph KPS given by Equation (7).

However, if we make a comparison based on key storage and size of key pool, that is, fixing $k = 8$ and $n = 48$, we find that the KPS based on an extended Cayley hypergraph performs better, since the Eschenauer Gligor KPS has lower connectivity: $\mathsf{Pr}_1 \approx 0.796$, and poorer resilience: $\mathsf{fail}_1 \approx 0.167$ and $\mathsf{fail}_s > 0.5$ for $s \geq 4$.

In summary, although the extended Cayley hypergraph / strongly regular graph construction for a KPS does not scale well, it does provide good expansion, a lower bound for which can be easily found, and there are combinations of parameters k and n for which it outperforms the Eschenauer Gligor KPS.

4.3 Analysis using the incidence and concurrence graphs

Finally, we demonstrate how alternative graphical representations of a KPS allow us to perform further analysis. We introduce one more graph which can be used to represent a design, called the *concurrence graph*.

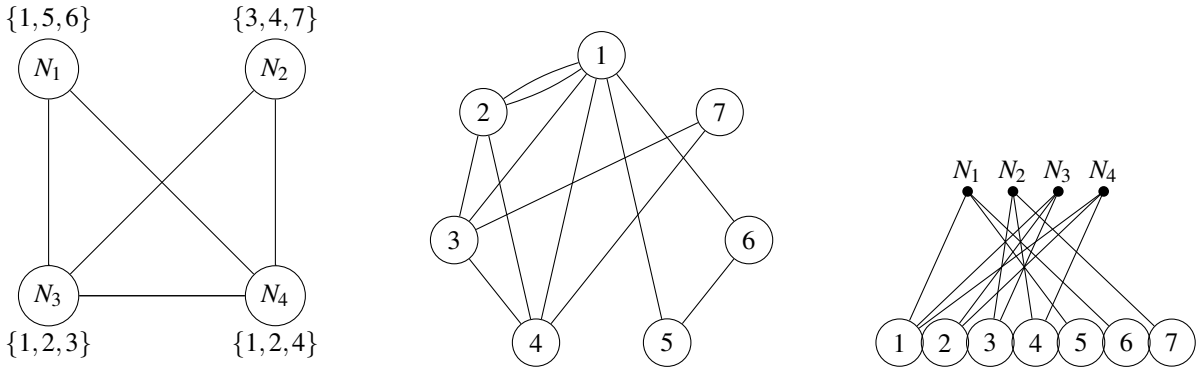
Definition 11 (from [3])**.** The *concurrence graph* of a design is defined in the following way. Each vertex corresponds to a point (or key), and distinct vertices are joined by $\gamma_{i,j}$ edges, where $\gamma_{i,j}$ is the concurrence of K_i and K_j, that is,

$$\gamma_{i,j} := |\{B_a \in \mathscr{B} : \{K_i, K_j\} \subseteq B_a\}| \quad .$$

In other words, $\gamma_{i,j}$ is the number of blocks containing keys K_i and K_j.

Notice that the concurrence graph is not necessarily a simple graph: there are no self-loops, but multiple edges may occur between a pair of vertices. However, the concurrence graph of a configuration is simple: recall that any two points occur in at most one block in a configuration, and that this maximises the degree of the design (Lemma 1). The concurrence graph of the $2-(9,3,1)$ configuration from Section 4.1.2 is therefore the complete graph on nine vertices.

In the context of KPSs, the concurrence graph helps us to understand the re-use of keys throughout the network. In particular, a concurrence of $\gamma_{i,j} > 1$ indicates that keys K_i and K_j are both known to more than one node. Let $\mathscr{B}_{i,j}$ be the set of $\gamma_{i,j}$ nodes storing keys K_i and K_j. Generally speaking, with limited key storage and connectivity, $\gamma_{i,j} > 1$ is undesirable for resilience: we would rather that a common neighbour of the set $\mathscr{B}_{i,j}$ is connected to each node of $\mathscr{B}_{i,j}$ using a different key, as illustrated by the simple example in Figure 8(a), where we consider the keys K_1 and K_2, known to nodes N_3 and N_4. Node N_1 knows key K_1, and is therefore a common neighbour of N_3 and N_4. Compromise of N_1 means that the adversary will learn keys K_1, K_5 and K_6, and thus know the key securing $\frac{1}{3}$ of the remaining connections. In contrast, node N_2 is connected to N_3 and N_4 using keys K_3 and K_4 respectively, so compromise of node N_2 leaves all remaining connections secure against an adversary.



(a) Block intersection graph (b) Concurrence graph (c) Incidence graph

Figure 8: Corresponding block intersection, incidence and concurrence graphs

The corresponding concurrence graph is given in Figure 8(b), and the incidence graph in Figure 8(c) for reference. Roughly speaking, it seems as if a more even spread of edges in the concurrence graph might correspond to a more optimal KPS. These ideas have been formalised and studied in the theory of optimal designs for statistical experiments and networks using the three measures of A-, D- and E-optimality. We refer the reader to [3, 4, 5, 22] for more details. Currently, little is known about these optimality criteria for designs with block size $k > 2$ or $n > 20$, so the results may be of limited use to KPS applications. We therefore present only a brief summary of these measures, seeking to highlight that future research in this area could have great impact on KPSs.

Definition 12. Let $\theta_1 \leq \theta_2 \leq \cdots \leq \theta_{v-1}$ be the non-trivial eigenvalues of the Laplacian of a design. For chosen values of v, b and k, a design is said to be

- *A-optimal* if it maximises the harmonic mean of $\theta_1, \theta_2, \ldots, \theta_{v-1}$;
- *D-optimal* if it maximises the geometric mean of $\theta_1, \theta_2, \ldots, \theta_{v-1}$;
- *E-optimal* if it maximises the value of θ_1 .

The following correspondences are collated from [3, 4]:

- A-optimality corresponds to minimising the pairwise effective resistances between all pairs of vertices, when the concurrence graph (similarly, the incidence graph) is viewed as an electrical network with one ohm resistance on each edge
- D-optimality corresponds to maximising the number of spanning trees in the concurrence graph and hence the incidence graph
- E-optimality corresponds to good expansion in the concurrence and incidence graphs

Thus, satisfying any of these optimality criteria is a good indicator of a strong KPS, with short average paths lengths, many paths between pairs of vertices, limited re-use of keys (lack of multi-edges in the concurrence graph) and a lack of bottlenecks. It seems very likely that progress in the area of classifying optimal designs will be useful for KPS research, either providing further support for the choice of existing combinatorial design-based KPSs, or suggesting new classes of designs for KPSs.

5 Conclusion and directions for further research

We have shown some of the many ways in which graph theory can be used to support the design and analysis of KPSs. Developing the ideas from [38], we have argued that a large expansion coefficient is desirable for the block intersection graph of a KPS, and that many existing schemes, including random KPSs and those based on μ-common intersection designs and strongly regular graphs, do provide good expansion. In Section 4 we suggested alternative graphical representations of KPSs, namely the incidence graph and the concurrence graph, and demonstrated how they can help us to analyse and design KPSs in new ways.

There are currently few constructions for hypergraphs which approach the optimal proven bounds for expansion. We have reasoned that any future constructions proposed for hypergraphs with good expansion are likely to correspond to good constructions for KPSs. However, we have noted that hypergraphs are set systems, and it is possible that any such construction may already be established in the literature as a combinatorial design. Indeed, since combinatorial designs are hypergraphs, our results in Section 3 about the expansion of configurations, μ-common intersection designs and strongly regular graphs support the claims of Section 4.

It seems possible that the problem of finding a construction for random uniform hypergraphs with good expansion may be related to the problem of finding random strongly regular graphs, as discussed in [33] and [12] respectively, and that solving either problem may lead to a good construction for a KPS. We note that in [41], Lanphier et al. consider expansion in the Levi (incidence) graph of symmetric block designs. They use vertex expansion, but it seems likely that a similar method could find bounds for the edge expansion, and/or the research on expanding hypergraphs could be adapted to consider vertex expansion, so that comparisons might be made.

Finally, we briefly introduced optimality criteria in Section 4.3. We argued that further research on A-, D- and E-optimal designs is likely to lead to further insights into KPSs, and suggest the following open questions for consideration:

- The work on expansion in KPSs is most closely linked with E-optimality. Are A-, D- and E-optimality equally important for KPSs, or is there one which should be prioritised?
- Which are the A-, D- and E-optimal designs for practical values of k, v and n for KPSs?
- Could another optimality criterion be defined which precisely captures the important features needed for a KPS?

Acknowledgements

The authors are extremely grateful to Rosemary Bailey, Aylin Cakiroglu and Leonard Soicher for the many helpful conversations on designs and optimality.

References

[1] Noga Alon and Vitali D. Milman. λ_i, Isoperimetric inequalities for graphs, and superconcentrators. *Journal of Combinatorial Theory, Series B*, 38(1):73–88, 1985.

[2] Noga Alon and Joel H. Spencer. *The Probabilistic Method*. John Wiley & Sons, Inc., Hoboken, NJ, USA, 2000.

[3] Rosemary A. Bailey and Peter J. Cameron. Combinatorics of optimal designs. *Surveys in Combinatorics 2009*, 365:19–73, 2009.

[4] Rosemary A. Bailey and Peter J. Cameron. Using graphs to find the best block designs. Preprint, 2011. `http://uk.arxiv.org/abs/1111.3768`.

[5] R. B. Bapat and A. Dey. Optimal block designs with minimal number of observations. *Statistics & Probability Letters*, 11(5):399–402, 1991.

[6] Simon R. Blackburn and Stefanie Gerke. Connectivity of the uniform random intersection graph. *Discrete Mathematics*, 309(16):5130–5140, 2009.

[7] Béla Bollobás. *Combinatorics: Set Systems, Hypergraphs, Families of Vectors, and Combinatorial Probability*. Cambridge University Press, Cambridge, 1986.

[8] Peter Buser. A note on the isoperimetric constant. *Annales scientifiques de l'École Normale Supérieure*, 15(2):213–230, 1982.

[9] Sera A. Cakiroglu. An upper bound on the algebraic connectivity of regular graphs. Preprint, 2014.

[10] Peter J. Cameron. *Combinatorics: Topics, Techniques, Algorithms*. Cambridge University Press, Cambridge, 1994.

[11] Peter J. Cameron. *Permutation Groups*. Cambridge University Press, Cambridge, 1999.

[12] Peter J. Cameron. Random strongly regular graphs? Preprint, 2001. `http://maths.qmul.ac.uk/pjc/preprints/randsrg.pdf`.

[13] Peter J. Cameron. Strongly regular graphs. Preprint, 2001. `http://designtheory.org/library/preprints/srg.pdf`.

[14] Peter J. Cameron and Jacobus H. van Lint. *Graph Theory, Coding Theory and Block Designs (London Mathematical Society Lecture Note Series)*. Cambridge University Press, Cambridge, 1975.

[15] Seyit A. Çamtepe and Bülent Yener. Combinatorial design of key distribution mechanisms for wireless sensor networks. In *Computer Security–ESORICS 2004*, volume 3193 of *Lecture Notes in Computer Science*, pages 293–308. Springer, 2004.

[16] Seyit A. Çamtepe and Bülent Yener. Key distribution mechanisms for wireless sensor networks: a survey. *Rensselaer Polytechnic Institute, Computer Science Department, Technical Report TR-05-07*, 2005.

[17] Seyit A. Çamtepe, Bülent Yener, and Moti Yung. Expander graph based key distribution mechanisms in wireless sensor networks. In *ICC 06, IEEE International Conference on Communications*, pages 2262–2267, 2006.

[18] Haowen Chan, Adrian Perrig, and Dawn Song. Random key predistribution schemes for sensor networks. In *SP '03: Proceedings of the 2003 IEEE Symposium on Security and Privacy*, pages 197–213. IEEE Computer Society, 2003.

[19] Jeff Cheeger. A lower bound for the smallest eigenvalue of the Laplacian. *Problems in Analysis*, 625:195–199, 1970.

[20] Chi-Yuan Chen and Han-Chieh Chao. A survey of key distribution in wireless sensor networks. *Security and Communication Networks*, July 2011.

[21] Wai-Kai Chen. *Applied Graph Theory: Graphs and Electrical Networks*. North-Holland Publishing Company, Amsterdam, 1976.

[22] C.-S. Cheng and Rosemary A. Bailey. Optimality of some two-associate-class partially balanced incomplete-block designs. *The Annals of Statistics*, 19(3):1667–1671, 1991.

[23] Fan R. K. Chung. *Spectral graph theory (CBMS Conference on Recent Advances in Spectral Graph Theory)*. American Mathematical Society, 1997.

[24] Charles J. Colbourn and Jeffrey H. Dinitz. *Handbook of Combinatorial Designs*. Chapman & Hall / CRC, 2010.

[25] Giuliana Davidoff, Peter Sarnak, and Alain Valette. *Elementary Number Theory, Group Theory and Ramanujan Graphs*, volume 55 of *London Mathematical Society Student Texts*. Cambridge University Press, 2003.

[26] Yvo Desmedt, Niels Duif, Henk van Tilborg, and Huaxiong Wang. Bounds and constructions for key distribution schemes. *Advances in Mathematics of Communications*, 3(3):273–293, 2009.

[27] Roberto Di Pietro, Luigi V. Mancini, Alessandro Mei, Alessandro Panconesi, and Jaikumar Radhakrishnan. Redoubtable Sensor Networks. *ACM Transactions on Information and Systems Security (TISSEC)*, 11(3):1–22, 2008.

[28] Józef Dodziuk. Difference equations, isoperimetric inequality and transience of certain random walks. *Transactions of the American Mathematical Society*, 284(2):787–794, August 1984.

[29] Józef Dodziuk and Wilfrid S. Kendall. Combinatorial Laplacians and isoperimetric inequality. *From local times to global geometry, control and physics*, 150:68–74, 1986.

[30] Danny Dolev, Cynthia Dwork, Orli Waarts, and Moti Yung. Perfectly secure message transmission. *Journal of the ACM*, 40(1):17–47, 1993.

[31] Paul Erdös and Alfréd Rényi. On the evolution of random graphs. In *Publication of the Mathematical Institute of the Hungarian Academy of Sciences*, pages 17–61, 1960.

[32] Laurent Eschenauer and Virgil D. Gligor. A key-management scheme for distributed sensor networks. In *Proceedings of the 9th ACM Conference on Computer and Communications Security - CCS '02*, pages 41–47. ACM, 2002.

[33] Joel Friedman. Some graphs with small second eigenvalue. *Combinatorica*, 15(1):31–42, 1995.

[34] Joel Friedman and Avi Wigderson. On the second eigenvalue of hypergraphs. *Combinatorica*, 15(1):43–65, 1995.

[35] Frank Harary. *Graph Theory*. Addison-Wesley, Reading, MA, 1969.

[36] Shlomo Hoory, Nathan Linial, and Avi Wigderson. Expander graphs and their applications. *Bulletin - American Mathematical Society*, 43(4):439–562, 2006.

[37] Michelle Kendall, Ed Kendall, and Wilfrid S. Kendall. A generalised formula for calculating the resilience of random key predistribution schemes. Preprint, 2012. `http://eprint.iacr.org/2012/426`.

[38] Michelle Kendall and Keith M. Martin. On the role of expander graphs in key predistribution schemes for wireless sensor networks. In *Research in Cryptology*, volume 7242 of *Lecture Notes in Computer Science*, pages 62–82. Springer, 2012.

[39] Michelle Kendall, Keith M. Martin, Siaw-Lynn Ng, Maura B. Paterson, and Douglas R. Stinson. Broadcast-enhanced key predistribution schemes. *ACM Transactions on Sensor Networks*, to appear, 2015. `http://eprint.iacr.org/2012/295`.

[40] Jon Kleinberg and Ronitt Rubinfeld. Short paths in expander graphs. *IEEE Annual Symposium on Foundations of Computer Science*, 37:86–95, 1996.

[41] Dominic Lanphier, C. Miller, Jason Rosenhouse, and A. Russell. Expansion properties of Levi graphs. *Ars Combinatoria*, 80(3):1–7, 2006.

[42] Jooyoung Lee and Douglas R. Stinson. A combinatorial approach to key predistribution for distributed sensor networks. In *IEEE Wireless Communications and Networking Conference*, pages 1200–1205. IEEE, 2005.

[43] Jooyoung Lee and Douglas R. Stinson. Deterministic key predistribution schemes for distributed sensor networks. In *Selected Areas in Cryptography*, volume 3357 of *Lecture Notes in Computer Science*, pages 294–307. Springer, 2005.

[44] Jooyoung Lee and Douglas R. Stinson. Common intersection designs. *Journal of Combinatorial Designs*, 14(4):251–269, 2006.

[45] Friedrich Wilhelm Levi. *Finite Geometrical Systems: six public lectures delivered in February, 1940, at the University of Calcutta*. The University of Calcutta, 1942.

[46] Alon Nilli. On the second eigenvalue of a graph. *Discrete Mathematics*, 91(2):207–210, 1991.

[47] Maura B. Paterson and Douglas R. Stinson. A unified approach to combinatorial key predistribution schemes for sensor networks. *Designs, Codes and Cryptography,* advance online publication, doi: 10.1007/s10623-012-9749-4. Springer, 2012.

[48] Eric Purdy. *Locally expanding hypergraphs and the unique games conjecture*. PhD thesis, University of Chicago, 2008.

[49] Hosein Shafiei, Arash Mehdizadeh, Ahmad Khonsari, and Mohamed Ould-Khaoua. A combinatorial approach for key-distribution in wireless sensor networks. In *IEEE Global Telecommunications Conference*, pages 1–5. IEEE, 2008.

[50] Jacobus H. van Lint and Alexander Schrijver. Construction of strongly regular graphs, two-weight codes and partial geometries by finite fields. *Combinatorica*, 1(1):63–73, 1981.

[51] Walter D. Wallis. *Combinatorial Designs (Pure and Applied Mathematics)*. Marcel Dekker Inc, 1988.

[52] Yang Xiao, Venkata K. Rayi, Bo Sun, Xiaojiang Du, Fei Hu, and Michael Galloway. A survey of key management schemes in wireless sensor networks. *Computer Communications*, 30(11-12):2314–2341, 2007.

[53] Osman Yağan and Armand M. Makowski. On the existence of triangles in random key graphs with a note on their small-world property. Preprint, 2013. `http://andrew.cmu.edu/user/oyagan/Journals/IT_Triangle.pdf`.