# Code Based KPD Scheme with Full Connectivity: Deterministic Merging

Pinaki Sarkar, Aritra Dhar

# Code Based KPD Scheme With Full Connectivity: Deterministic Merging

Pinaki Sarkar[1] and Aritra Dhar[2]

[1] Department of Mathematics, Jadavpur University, Kolkata – 700032, INDIA
pinakisark@gmail.com
[2] Department of Computer Science, III Technology – Delhi, New Delhi – 110020, INDIA
aritra1204@iiitd.ac.in

**Abstract.** Key PreDistribution (KPD) is one of the standard key management techniques of distributing the symmetric cryptographic keys among the resource constrained nodes of a Wireless Sensor Network (WSN). To optimize the security and energy in a WSN, the nodes must possess common key(s) between themselves. However there exists KPDs like the Reed Solomon (RS) code based schemes, which lacks this property. The current work proposes a deterministic method of overcoming this hazard by merging exactly two nodes of the said KPD to form blocks. The resultant merged block network is fully connected and comparative results exhibit the improvement achieved over existing schemes. Further analysis reveal that this concept can yield larger networks with small key rings.

**Keywords:** Key predistribution (KPD), Reed Solomon (RS) codes, Combinatorial Designs, Deterministic Merging Blocks, Connectivity, Security

## 1 Introduction

The increasing necessity of dealing with classified information from hazardous deployment area is enhancing the popularity of Wireless sensor networks (WSN). Such networks typically consists of Key Distribution Server (KDS) or Base Station (BS), identical (low cost) ordinary sensors (or nodes) and at times some special nodes. The BS links the network to the user and sometimes these networks have more than one BS. This along with other flexibilities like lack of fixed infrastructure imply that such networks are *Ad Hoc* in nature.

Each entity constituting a WSN typically consists of a power unit, a processing unit, a storage unit and a wireless transceiver. Capacities of each such unit in any ordinary node is quite limited for any WSN while the KDS is quite powerful. Resource constrained nodes are supposed to gather specific information about the surrounding, process them and communicate to the neighboring nodes, i.e., nodes within their (small) *radius of communication*. The processed data is relayed up to the KDS which has relatively (large) *radius of communication* for further analysis before informing the user.

In spite of all the weaknesses in the basic building blocks of WSNs, these networks have several military applications like monitoring enemy movements, etc. Besides they are utilized for other scientific purposes like smoke detection, wild fire detection, seismic activity monitoring etc. In all its applications, WSNs once deployed are expected to work unattended for long duration of time while its constituent nodes deals with lot of sensitive data.

### 1.1 Related Works

Most recent applications of WSNs require secure message exchange among the nodes. One ideally likes to apply lightweight symmetric key cryptographic techniques in order to avoid heavy or costly computations within the resources constraints nodes. Such cryptographic techniques demands the communicating parties to possess the same key prior to message exchange. Standard online key exchange techniques involving public parameters or using trusted authorities are generally avoided. Instead, *Key PreDistribution (KPD)* techniques are preferred. Eschenauer and Gligor [5] suggested the pioneering idea of KPD scheme where:

- *Preloading of Keys* into the sensors prior to deployment.
- *Key establishment*: this phase consists of
  - *Shared key discovery*: establishing shared key(s) among the nodes;
  - *Path key establishment*: establishing path via other node(s) between a given pair of nodes that do not share any common key.

Random preloading of keys means that the *key rings* or *key chains* are formed randomly. In [5], *key establishment* is done using *challenge and response* technique. Schemes following similar random preloading and probabilistically establishing strategy are called *random KPD schemes*. More examples of such schemes are [4, 7]. Çamptepe and Yener [2] presents an excellent survey of such schemes.
On the other hand, there exists KPD schemes based on deterministic approach, involving *Mathematical* tools. Çamptepe and Yener [1] were first to propose a deterministic KPD scheme where keys are preloaded and established using *Combinatorial Designs*. Following their initial work, numerous deterministic KPD schemes based on combinatorial designs like [6, 8, 10] have been proposed. There exists *hybrid* KPDs like [3, 9] that use both random and deterministic techniques. There exists some interesting designs like one in [8] using Reed Solomon (RS) code which can be viewed as a combinatorial design. One may refer to [6] for discussions about various combinatorial designs necessary for this paper. For the sake of completeness, an outline on combinatorial designs is presented in Section 3. The said section establishes that the RS code based KPD [8] can be treated as a Group–Divisible Design (GDD) or Transversal Design (TD).

### 1.2 Contributions in this paper

The original scheme of [6] lacks full communication among nodes as a pair of nodes may not share a common key. This involves an intermediate node which increase the communication overhead. As a remedial strategy, Chakrabarti et al. [3] first suggested the idea of *random merging of nodes* to form blocks. Their strategy was to randomly merge '$z$' nodes of [6] to form blocks having bigger key rings. The resultant network thus possessed '$\lfloor \mathscr{N}/z \rfloor$' blocks where $\mathscr{N}$ is the number of nodes in the original KPD [6]. However full communication was still not guaranteed and many aspects of their design, like the basic concept of merging, choice of nodes while merging, the heuristic in [3, Section 4] etc.have not been explained. A similar random merging concept was proposed by Sarkar and Dhar [9] for the RS code based KPD [8]. Full connectivity is guaranteed for $z \geq 4$ (see [3, Theorem 1]) The solution to be presented here is entirely different and performs more efficiently.
Motivated by the merging concept, the present authors thought of proposing a deterministic merging technique. Here exactly two (2) nodes of the KPD [8] are

merged. Theorem 2 of Section 4 establishes that merging two nodes of the KPD [8] in a certain fashion results in full communication among the newly formed (merged) blocks. The resiliency and scalability are also comparable.

## 2 Basics of Combinatorial Design

This section briefly describes some basic notion of *combinatorial design* necessary for understanding Ruj and Roy [8] scheme.

*Group-Divisible Design (GDD) of type $g^u$, block size k:* is a triplet $(\mathscr{X}, \mathscr{H}, \mathscr{A})$:

1. $\mathscr{X}$ is a finite set with $|\mathscr{X}| = gu$.
2. $\mathscr{H}$ is a partition of $\mathscr{X}$ into $u$ parts, that is, $\mathscr{H} = \{H_1, H_2, H_3, \ldots, H_u\}$ with $\mathscr{X} = H_1 \cup H_2 \cup H_3 \cup \ldots \cup H_u$, $|\mathscr{H}_i| = g \; \forall \; 1 \leq i \leq u$ and $\mathscr{H}_i \cap \mathscr{H}_j = \phi \; \forall \; 1 \leq i \neq j \leq u$.
3. $\mathscr{A}$ is the collection of blocks of $\mathscr{X}$ having the following properties: $|H \cap A| \leq 1 \; \forall \; H \in \mathscr{H}$, $\forall \; A \in \mathscr{A}$, given any pair of varieties $x \in H_i, y \in H_j$ with $i \neq j \; \exists$ unique $A \in \mathscr{A}$ such that $x, y \in A$.

*Transversal Designs (TD(k,n)):* are special type of GDDs with $g = n, u = k$, while the parameter $k$ remains the same. These can be shown to form $(nk, n^2, n, k)-$ configuration. One is referred to [6, Section III] for definition of configuration and other related concepts.

*Common Intersection Design (CID):* Let $(\mathscr{X}, \mathscr{A})$ is a $(v, b, r, k)$–configuration. Recall from [6], $(\mathscr{X}, \mathscr{A})$ is called a $\mu-$common intersection design ($\mu$–CID) if: $|\{A_\alpha \in \mathscr{A} : A_i \cap A_\alpha \neq \phi \text{ and } A_j \cap A_\alpha \neq \phi\}| \geq \mu$ whenever $A_i \cap A_j = \phi$. While for the sake of consistency, in case $A_i \cap A_j \neq \phi, \forall \; i, j$ one defines $\mu = \infty$ .

*Maximal CID:* For any given set of parametric values of $(v, b, r, k)$, such that a configuration can be obtained with them, one would like to construct a configuration with maximum possible $\mu$. This *maximal value of* $\mu$ is denoted $\mu^*$. *Theorem 14.* of [6, Section IV] establishes $TD(k, n)$ designs are $k(k-1)^* - CID$.

## 3 KPD Using Reed Solomon (RS) codes

This section is devoted to the description of KPD scheme proposed by Ruj and Roy in [8]. The scheme uses Reed Solomon (RS) codes to predistribute and establish the communication keys among the sensor nodes. The construction of RS codes has been given in [8]. Salient features are being sketched below:

To construct $(n, q^l, d, q)$ RS code having alphabet in the finite field $\mathbb{F}_q$ ($q$: prime or prime power $> 2$), consider the following set of polynomials over $\mathbb{F}_q$:

$$\mathscr{P} = \{g(y) : g(y) \in \mathbb{F}_q[y] \; deg(g(y) \leq l-1\}.$$

Thus the number of elements in $\mathscr{P}$ denoted by $|\mathscr{P}| = q^l$. Let $\mathbb{F}_q^* = \{\alpha_1, \alpha_2, \alpha_3, \ldots, \alpha_{q-1}\}$ be the set of non-zero elements of $\mathbb{F}_q$. For each polynomial $p_m(y) \in \mathscr{P}$, Define $cp_m = (p_m(\alpha_1), p_m(\alpha_2), \ldots, p_m(\alpha_{q-1}))$ to be the $m^{\text{th}}$ codeword of length $n = q-1$. Let $C = \{cp_m : p_m(y) \in \mathscr{P}\}$ be the collection of all such code words formed out of the polynomials over $\mathbb{F}_q$. This results in a RS code. Since the number of code-words is $q^l$, the system can support up to $q^l$ nodes.

Now the polynomial $p_m$ and the corresponding codeword $cp_m$ are given to the $m^{\text{th}}$ node. For the codeword $cp_m = (a_1, a_2, \ldots, a_n)$, one assigns the keys having key-identifiers $(a_1, \alpha_1), (a_2, \alpha_2), \ldots, (a_n, \alpha_n)$ where $a_j = p_m(\alpha_j), j = 1, 2, \ldots, n$ to the

$m^{\text{th}}$ node. The node id of the $m^{\text{th}}$ node is obtained by evaluating the polynomial $p_m$ at $x = q$ and taking only the numerical value. That is the $m^{\text{th}}$ node has the node id $p_m(q)$ (without going modulo '$p$').
A WSN with 16 nodes based on RS code parameters $q = 4, n = 3$ and $l = 2$ is presented in Table 3.1. Here '2' means the polynomial '$x$' and '3' means the polynomial '$x+1$' modulo the irreducible polynomial $x^2+x+1$ over $\mathbb{F}_2[x]$ which are commonly referred to as $\bar{x}$ and $\overline{x+1}$. Thus $0,1,2,3$ forms the finite field $\mathbb{F}_4$. The nodes' polynomials $i+jy \in \mathbb{F}_4[y]$ for $0 \le i, j \le 3$ are given in $2^{\text{nd}}$ row of Table 3.1. By evaluating these polynomials at non-zero points, the keys $(p_m(b), b), 0 \ne b \in \mathbb{F}_q, 0 \le i, j \le 3$ have been derived and tabulated in the corresponding columns. Table 3.1 constructed by similar computations is being presented in a slightly different manner from Ruj and Roy [8]. This *GDD* form of presentation helps one realize the similarity of the RS code based KPD of [8] with the $TD(q-1,q)$ with parameters $q-1,q$ of [6]. Though in Theorem 6 of [6, Section III], constructions $TD(k,p), 2 \le k \le q, p$ a prime is given, it can be extended to $TD(k,q), q = p^r$. Since the construction of the KPDs $TD(k,p)$ of [6] utilized the field properties of $\mathbb{F}_p$, one can extend it to $\mathbb{F}_q = \mathbb{F}_{p^r}$. Extending the base field from $\mathbb{F}_p$ to $\mathbb{F}_q = \mathbb{F}_{p^r}$ and following similar constructions as given in [6, Section III] yields $TD(k,q), q = p^r, r \in \mathbb{N}$. Now taking $k = q-1$ results in $TD(q-1,q)$. However it is important to state that in $TD(q-1,q)$ design is different from RS code. In $TD(q-1,q)$ design, the evaluation is done for $y = 0,1,\ldots,q-2$ while in RS code based design, it is done at non–zero points, $y = 1,2,3,\ldots,q-1$.
$N_0$ to $N_{15}$ denotes the nodes with ids ranging from 0 to 15 whose polynomials are represented in the column immediately below it. Key ids contained in a node are presented in the columns below each node. $V/C$ denotes the distinct *Variety Classes* $H_1, H_2, H_3$, where $H_d = \{(i,d) : 0 \le i \le 3\}$ for $d = 1,2,3$. One notes that the scheme under consideration is a $(q-1)(q-2)-$CID as the number of keys per node $= k = q-1$ (see Section 2). Thus for nodes not sharing any key, there are enough nodes which can play the role of the intermediate node in multi-hop (2-hop) process. This encourages the search for a deterministic merging design with exactly two nodes per block yielding full communication among the blocks.

### 3.1 Weakness of the above RS code based KPD

Apart from other possible weaknesses, the RS code based KPD presented in [8] lacks full communication among nodes (by the discussions above). So multi-hop communications occur among the nodes. Here multi-hop means some third party node other than the sender and receiver decrypts and encrypts the ciphertext. Other than increasing the cost of communication, this enhances the chances of adversarial attacks on such communication. Thus the energy efficiency as well as the security of message exchange of the entire network might be grossly affected.

## 4 Remedy: Deterministic Merging of Nodes

Lack of direct communication for any arbitrarily chosen pair of nodes of the KPD [8] can be tackled by merging certain number of nodes. For this, observe that Table 3.1 indicates the network having 16 nodes can be partitioned into 4 classes each containing 4 nodes on the basis of their key sharing. These classes are separated by double column partitioning lines after each set of 4 nodes: $N_0, N_1, N_2, N_3$;

| *Nodes* | $N_0$ | $N_1$ | $N_2$ | $N_3$ | $N_4$ | $N_5$ | $N_6$ | $N_7$ | $N_8$ | $N_9$ | $N_{10}$ | $N_{11}$ | $N_{12}$ | $N_{13}$ | $N_{14}$ | $N_{15}$ |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $V/C$ | $0y+0$ | 1 | 2 | 3 | $y$ | $y+1$ | $y+2$ | $y+3$ | $2y$ | $2y+1$ | $2y+2$ | $2y+3$ | $3y$ | $3y+1$ | $3y+2$ | $3y+3$ |
| $H_1$ | (0,1) | (1,1) | (2,1) | (3,1) | (1,1) | (0,1) | (3,1) | (2,1) | (2,1) | (3,1) | (0,1) | (1,1) | (3,1) | (2,1) | (1,1) | (0,1) |
| $H_2$ | (0,2) | (1,2) | (2,2) | (3,2) | (2,2) | (3,2) | (0,2) | (1,2) | (3,2) | (2,2) | (1,2) | (0,2) | (1,2) | (0,2) | (3,2) | (2,2) |
| $H_3$ | (0,3) | (1,3) | (2,3) | (3,3) | (3,3) | (2,3) | (1,3) | (0,3) | (1,3) | (0,3) | (3,3) | (2,3) | (2,3) | (3,3) | (0,3) | (1,3) |

**Table 1.** Polynomials, Node and Key identifiers for $q^2 = 16$ nodes. Table adapted from section 3.1 of Ruj and Roy [8]. Alternative presentation: Group–Divisible Design (GDD) form.

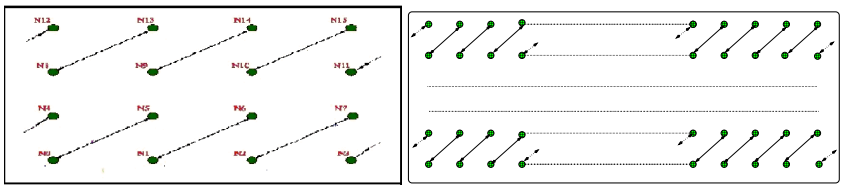
$N_4,N_5,N_6,N_7$; $N_8,N_9,N_{10},N_{11}$; and $N_{12},N_{13},N_{14},N_{15}$. Every class has the property that the coefficient of $y$ in their respective polynomials is same. Equating each other's polynomials $i+jy$ with $0 \le i \le 3$ for some fixed $j = 0,1,2$ or 3 results in no common solution $\implies$ no common key. For eg. with $j = 1$ with $i = 0,1,2,3$, the corresponding 4 polynomials: $0+y, 1+y, 2+y, 3+y$ do not have any common solution. Hence no shared keys for corresponding nodes.

The other case for any pair of nodes not sharing any common key is whenever their constant term is same since only non–zero values of $y$ are allowed. This gives rise to alternative type of partition: $N_0,N_4,N_8,N_{12}$; $N_1,N_5,N_9,N_{13}$; $N_2,N_6,N_{10},N_{14}$; and $N_3,N_7,N_{11},N_{15}$. This motivates one to visualize the key sharing of the 16 nodes, $N_0$ to $N_{15}$, like a 'square-grid' as presented in Figure 1(a). Any pair of nodes, other than the ones lying in the same row or column shares exactly 1 key as the equations: $(j-j')y = (i'-i)$ has unique solution over non–zero points of $\mathbb{F}_4$ (since $q = 4$) that is with $0 \le i \ne i', j \ne j' \le 3$. Merging of nodes in pairs for the case $q = 4$ can be now achieved as indicated by the slanted line in Figure 1(a). Basically the strategy is to merge the nodes: $N_{(i,j)}$ and $N_{(i\oplus 1, j\oplus 1)}$ where $\oplus$: addition in $\mathbb{F}_4$ (addition modulo 2), for $j = 0,2$.

A natural deterministic merging strategy of 2 nodes can now be visualized for $q = 2^r \implies \mathcal{N} = q^2 = 2^{2r}$. Figures 1(b) demonstrate the strategy. Nodes occurring at the ends of a slanted line are merged. Idea is to break up the network into pairs of rows, i.e. $\{1,2\};\{3,4\},\ldots,\{2^{r-1},2^r\}$ and apply similar process.

Before explaining the general odd case, it is useful to visualize the case when $q = 5$, i.e. a network with $q^2 = 5^2 = 25$ nodes as presented in Figure 2(a). Rest of the discussion is similar to that of the case $q = 4 = 2^2$ except for the merging of last three rows. As usual the arrows indicate the merging strategy. The strategy indicated in Figure 2(a) is same for the first and second rows while differs in the last three rows when compared to Figure 1(a). This is because a similar strategy like previous cases would imply one row is left out. Of course for $q = p = 5$ all arithmetic operations are 'modulo 5' arithmetic operations as done in $\mathbb{F}_5$.

For general odd prime power case $q = p^r : p$ is a odd prime, any pair of nodes among $\{N_{(i,j)} : 0 \le j \le q-1\}$ for every $0 \le i \le q-1$ (i.e., nodes with ids $i+jq$ and occurring in $i^{\text{th}}$ row; $q-$fixed) do not share any common key. Same is the case with the nodes $\{N_{m,j} : 0 \le m \le q-1\}$ *for every* $0 \le j \le q-1$ with ids $m+jq$ (occurring in $j^{\text{th}}$ column of Figure 2(b)). For any other pair of nodes, equating corresponding linear polynomials they find exactly one (1) common shared key between them (since $l = 2$). The general case of $q^2$ nodes ($l = 2$) can visualized as a $q \times q$ 'square-grid' as in Figure 2(b) with nodes in same row or column having no key in common while any other pair of nodes share exactly one common key. This merging strategy is indicated in Figure 2(b) by slanted arrows as before. Nodes occurring at the ends of a slanted line are merged. The idea is to look



(a) MB Design 1: Particular case $q = 4 = 2^2 \implies \mathscr{N} = 16 = 4^2$.

(b) MB Design 1: General even case for $q = 2^r, r \in \mathbb{N} \implies \mathscr{N} = 2^{2r}$ nodes

**Fig. 1.** Deterministic Merging Blocks Strategy for all even prime power cases $q = 2^r, r \in \mathbb{N}$

at two rows at a time and form blocks containing one node of each except for last 3 rows. For fixed $0 \leq i \leq q-2$, merge the nodes $N_{(i,j)}$ and $N_{(i\oplus 1, j\oplus 1)}(\oplus :$ addition modulo $q)$, for $0 \leq j \leq q-3$ (with increment of 2) $\forall\ q > 4$. The last node of every odd row is merged with the first node of the row above it. Since $q$ is odd, taking combination of two row would have left out one row, so top three row are combined separately.
Note that, in case merging of nodes is done randomly, one may end up with merged pairs like $N_{(0,0)} \cup N_{(0,1)}$ and $N_{(0,2)} \cup N_{(0,3)}$, (for $q \geq 4$) which do not share any common key, thus not be able to communicate even after merging.

### 4.1 Assured full communication: theoretical results

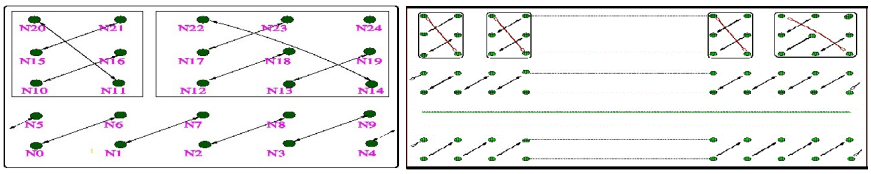
Equating the polynomials of the 4 nodes constituting any 2 merged blocks yields:

**Theorem 1.** *The proposed Deterministic Merging Block Strategy where 2 nodes of the RS code based KPD [8] are clubbed to form the merged blocks results in full communication among the merged blocks.*

*Proof.* Consider any two arbitrary blocks $A$ and $B$. It is evident from the construction that at least node from block $A$ will never lie in the horizontal line as well as the vertical line of either of the two nodes the other block $B$ (refer to Figures 1(a), 1(b) 2(a) and 2(b) for $q = 4, 2^r, 5$ and for general case). This implies that these two nodes will have a common key as discussed in Section 4. Hence the blocks $A$ and $B$ can communicate through this key. As the two blocks were arbitrarily chosen, one is assured of full communication in the new network consisting of blocks constructed by merging two nodes in the manner explained above (see Figures 1(a), 1(b) 2(a) and 2(b) for $q = 4, 2^r, 5$ and for general case respectively).

**Theorem 2.** *The resulting Merged Block Design has a minimum of* one *to a maximum of* four *common keys between any two given pair of (merged) blocks.*

*Proof.* Any two nodes can share at most one key in original RS code based KPD in [8]. So there are at most 4 keys common between two blocks. This situation occurs only if both nodes of the $1^{\text{st}}$ block shares two $(2)$ distinct keys with each node of the $2^{\text{nd}}$ block.




(a) MB Design 1: Special case for $q = p = 5 \implies \mathscr{N} = 25 = 5^2$ nodes.

(b) MB Design 1: General odd prime / prime power $q = p^r, r \in \mathbb{N} \implies q^2$ nodes.

**Fig. 2.** Deterministic Merging Blocks Strategy for all odd prime power cases $q = 2^r, r \in \mathbb{N}$

*Remark 1.* Some important features of the merging block design are as follows:

- Merging does not mean that the two nodes combine physically to become one. Just that they are to be treated as one unit.
- The resultant merged block design has full communication among the blocks through at least one common key between any two given pair of merged block ensuring full communication in resultant network.
- Full communication can not be assured when nodes are merged randomly to form larger blocks. Probably this is the main reason why authors of [3] could not justify several issues in their random merging model.
- The current authors feel that it is mandatory to have inter nodal communication. Any communication received by either of the two constituent nodes of a block can be passed down to the other node and hence make the other node connected. As such, while proposing the merged block design, this consideration was given importance.
- Therefore the total number links in the merged scheme is same as that of the original RS code based KPD of Ruj and Roy in [8]. This fact will be recalled later while discussing resiliency of the system.

## 5 Network Parameters

Some important aspects of the combined scheme like *communication probability, computational overhead, resiliency and scalability* will be present in this section.

### 5.1 Communication Probability; Overhead; Network Scalability

*Communication Probability* or *Connectivity* is defined to be the ratio of number of links existing in the network with respect to the total number of possible links. A *link* is said to exists between two nodes they share at least one common key. Figure 3 presents a comparison between the present scheme v/s existing schemes in terms of $(E(s))$ and connectivity. The graph in Figure 3(a) is plotted with number of nodes in the network in x-axis v/s $E(s)$ in y-axis. It compares the resiliency of Merged Block network with other existing schemes. The graph plotting in Figure 3(b) based on the varying number of keys per node, $k$ in x-axis) v/s connectivity in y-axis. The original KPD [8] is assumed to have $q = 49, \mathscr{N} = 2401$ many nodes.

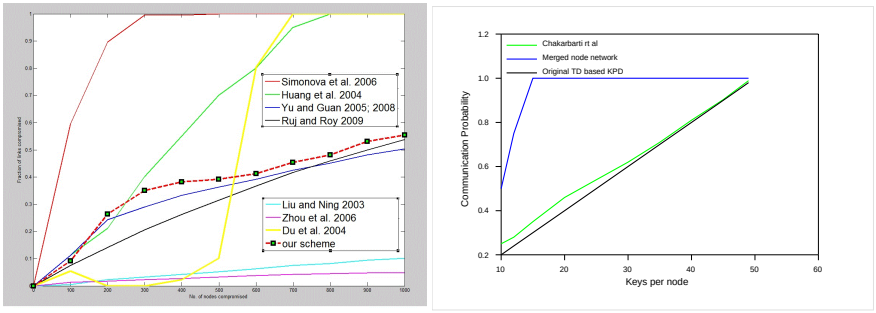
So that the merged network has $\mathscr{N}_{mb} = 1200$ nodes. Clearly the present Merging Block Design provides much better communication than the original RS code based KPD [8] and the random models of [3, 9] (both almost same) even when no. of keys per nodes, $(k)$ decreases. For the RS code based KPD [8], taking the key ring of node $(i, j)$ as $\{(p_{i+jq}(\alpha_c), \alpha_c) : h \leq \alpha_c \leq q-1\}, 1 \leq h \leq q-2$ yields decreasing key rings for increasing $h$. The present Merging Block Design over [8] possesses *full connectivity* $\forall\ k \geq \lfloor \frac{q+7}{4} \rfloor$. This follows from the observations that any 2 block share a min. of $4k-6$ keys, there are $q$ keys in the network and the *pigeon-hole-principle*. Present design connectivity $= 1\ \forall\ k \geq 14$ as $q = 49$.
*Communication overhead* measures the computational complexity of both the key establishment and the message exchange protocols. During *key establishment* polynomials of $(l-1)^{th}$ degree are equated which involves computing inverses over $\mathbb{F}_q$. Since quadratic, cubic and quartic equations can be solved in constant time, this design is valid for for $l = 2, 3, 4$ and 5. However complexity of quadratic, cubic and quartic is quite high, specially for the nodes. Even the resiliency falls drastically with increasing value of $l$. So practically $l = 2$ is considered. For message exchange, complexity is same as that of original RS code based KPD [8]. Of course the complexity depends on the cryptosystem being used. Any *cryptosytem* meant for *embedded systems like AES-128*, is applicable.

### 5.2 Resiliency

Before proceeding with further analysis in the remaining part of the paper, some terminologies need to be introduced. The term 'uncompromised node(s)' associates to node(s) that are not compromised/captured. The *link* between any two uncompromised nodes is said to be *disconnected* if all their shared key(s) gets exposed due to capture of $s$ nodes. Standard resiliency measure, $E(s)$ (recalled below) is considered while analyzing/comparing the scheme with existing KPDs. $E(s)$ measures the ratio of number of links disconnected due to capture of $s$ nodes (here blocks) with respect to the total number of links in the original setup. Mathematically: $E(s) = \frac{\text{number of links broken due to capture of s nodes (here blocks)}}{\text{total number of link in the original setup}}$
One can refer to [8, Section 6.4] for an estimated upper bound of $E(s)$. Construction of the Merging Blocks clearly indicated that a merged network of size $\mathscr{N}$ corresponds to $\approx 2\mathscr{N}$ sized original KPD, while capture of $s$ merged blocks is almost equivalent $2s$ nodes of original. Thus the ratio of links broken $(E(s))$ remains almost the same ($\approx$ old $E(s)$) as observed experimentally. Providing accurate theoretical explanations is rather difficult due to the randomness of node capture both the original and its merged KPDs.

## 6 Simulation and Comparative Results

Run results after 100 runs for each data set are tabulated in Table 2. $\mathscr{N}_{RS}(= q^2)$ denotes the number of nodes of the original KPD scheme in [8]. While $\mathscr{N}_{MB}$ denotes the number of blocks in merged network. Clearly $\mathscr{N}_{MB} = \lfloor \frac{\mathscr{N}_{RS}}{2} \rfloor$. Here, $p$ is a prime number and $q = p^r$ is a prime power. Any given pair of nodes has at most 1 key in common as $l = 2$. Let $s_{MB}$ and $s_{RS}$ be the number of blocks and nodes captured in the original and its merged KPD respectively. Then $E_{RS}(s)$ and $E_{MB}(s)$ denotes the resiliency coefficients in the original and merged blocks respectively. The mentioned tables compares the simulated values of ratio of links disconnected $E(s)$ in the Merging Block model with its original KPD [8].




(a) Comparative E(s) values. (b) Comparative connectivity values.

**Fig. 3.** Graphs showing the comparison between the MB design over RS code based KPD v/s existing schemes with regards to connectivity and resiliency (E(s)) values.

## 7 Conclusions and Future Works

Deterministic merging of nodes of RS code based KPD [8] has been proposed in this paper. As a result full communication between nodes is achieved. Though the merging is being done for the particular chosen KPD scheme in [8], the approach can be modified and generalized to other schemes. This enhances the applicability of the concept. To understand why deterministic is better than its random counterpart, the mentioned design is viewed combinatorially. Algebraic as well as design theoretic analysis of the mentioned KPD paves the logical approach behind the deterministic merging strategy. Remark 1 of Section 4.1 highlight some of the important features of the deterministic merging strategy.
One can readily visualize some immediate future research directions. Application of similar deterministic merging concept to network based on other KPDs which lacks full communication among its nodes like may result in interesting works. The reason of preferring such merging strategy over its random counterpart has been sketched. Deterministic approach can be generalized to other schemes like [6]. Generic survey of deterministic v/s random schemes (like current scheme v/s [3, 9]) yielding fully communicating networks can be future research topics.
The assurance of connectivity with lower keys (refer in Figure 3(b)) paves a direction of achieving fully communicating deterministic schemes having high resiliency. A priori one must look to design scheme having good node support, small key rings, high resilience and scalability. Mathematical solutions to such fascinating problems will be interesting. The deterministic property of the merging technique may enable it to be combined with other deterministic techniques like the one proposed in [10]. This will ensure that the merged design is free of 'selective node attack' which it still suffers from as the original KPD did.

## References

1. S. A. Çamtepe and B. Yener, Combinatorial design of key distribution mechanisms for wireless sensor networks, *In: ESORICS 2004*, Samarati, P.,

| $k=$ $q-1$ | $\mathcal{N}_{RS}$ | $\mathcal{N}_{MB}$ | $s_{MB}$ | $s_{RS}$ | $E_{MB}(s)$ | $E_{RS}(s)$ |
|---|---|---|---|---|---|---|
| 28 | 841 | 420 | 5 | 10 | 0.297007 | 0.297155 |
| 28 | 841 | 420 | 10 | 20 | 0.507596 | 0.509076 |
| 30 | 961 | 430 | 5 | 10 | 0.279395 | 0.280538 |
| 30 | 961 | 430 | 10 | 20 | 0.483049 | 0.484641 |
| 40 | 1681 | 840 | 5 | 10 | 0.219206 | 0.219512 |
| 40 | 1681 | 840 | 10 | 20 | 0.390549 | 0.391530 |
| 48 | 2401 | 1200 | 5 | 10 | 0.186518 | 0.186756 |
| 48 | 2401 | 1200 | 10 | 20 | 0.338435 | 0.338898 |
| 70 | 5041 | 2570 | 10 | 20 | 0.247447 | 0.247348 |
| 70 | 5041 | 2570 | 15 | 30 | 0.346970 | 0.347509 |
| 100 | 10201 | 5100 | 5 | 10 | 0.094725 | 0.094747 |
| 100 | 10201 | 5100 | 10 | 20 | 0.180525 | 0.180588 |
| 100 | 10201 | 5100 | 20 | 40 | 0.328717 | 0.328873 |
| 102 | 10609 | 5304 | 10 | 20 | 0.177422 | 0.177433 |
| 102 | 10609 | 5304 | 30 | 60 | 0.443885 | 0.444061 |

**Table 2.** Simulated $E(s)$ results for MB over RS code KPD: Comparison with RS code KPD.


Ryan, P.Y.A., Gollmann, D., Molva, R.(eds.), LNCS, vol. 3193, pp. 293–308. Springer, Heidelberg, 2004.

2. S. A. Çamtepe and B. Yener, Key distribution mechanisms for wireless sensor networks: A survey 2005. Technical Report, *TR-05-07 Rensselaer Polytechnic Institute, Computer Science Department*, March 2005.
3. D. Chakrabarti, S. Maitra, and B. Roy, A key pre-distribution scheme for wireless sensor networks: merging blocks in combinatorial design, *International Journal of Information Security*, vol. 5, no. 2, pp.105–114, 2006.
4. H. Chan, A. Perrig and D. X. Song. Random key predistribution schemes for sensor networks, *IEEE Symposium on Security and Privacy*, IEEE Computer Society, Los Alamitos, 2003.
5. L. Eschenauer and V. D. Gligor, A key-management scheme for distributed sensor networks, *ACM Conference on Computer and Communications Security*, pp. 41–47., 2002
6. J. Y. Lee and D. R. Stinson, A combinatorial approach to key predistribution for distributed sensor networks. *IEEE Wireless Communications and Networking Conference, WCNC 2005*, New Orleans, LA, USA, 2005.
7. D. Liu and P. Ning, Establishing pairwise keys in distributed sensor networks. *ACM Conference on Computer and Communications Security*, pp. 52–61. ACM, New York, 2003.
8. S. Ruj and B. Roy, Key Predistribution Schemes Using Codes in Wireless Sensor Networks, *Inscrypt 2008*, Lecture Notes in Computer Science 5487, Springer-Verlag Berlin Heidelberg, pp. 275–288, 2009.
9. P. Sarkar and A. Dhar Assured Full Communication by Merging Blocks Randomly in Wireless Sensor Networks based on Key Predistribution Scheme using RS code. *International Journal of Network Security and Its Applications (IJNSA)*, vol. 3, no. 5, pp. 203–215, September, 2011
10. P. Sarkar, A. Saha, M. U. Chowdhury, Secure Connectivity Model in Wireless Sensor Networks Using First Order Reed-Muller Codes. *MASS 2010.* pp. 507–512, 2010.