SIMPLE AND FLEXIBLE RANDOM KEY PRE-DISTRIBUTION SCHEMES FOR WIRELESS SENSOR NETWORKS USING DEPLOYMENT KNOWLEDGE

by

SİNAN EMRE TAŞÇI

Submitted to the Graduate School of Engineering and Natural Sciences
in partial fulfillment of
the requirements for the degree of
Master of Science

Sabancı University
February 2006

SIMPLE AND FLEXIBLE RANDOM KEY PRE-DISTRIBUTION SCHEMES FOR WIRELESS SENSOR NETWORKS USING DEPLOYMENT KNOWLEDGE

APPROVED BY:

Asst. Prof. Albert Levi …….
(Thesis Supervisor)

Asst. Prof. Cem Güneri …….

Asst. Prof. Özgür Gürbüz …….

Asst. Prof. Kemal Kılıç …….

Asst. Prof. Erkay Savaş …….

DATE OF APPROVAL: …….

ABSTRACT

Sensor nodes are tiny, low-power and battery constrained electromechanical devices that are usually deployed for sensing some type of data in different types of areas. Because of their memory and computational restrictions, public key cryptography (PKC) systems are not suited for sensor nodes to provide security. Instead, private key cryptography is preferred to be used with sensor networks and there has been considerable work in this area, but there still exist problems with private key cryptography because of memory restrictions of sensor nodes. Number of keys that can be deployed into a sensor node is determined by the available memory of that node which is limited even private key cryptographic techniques are applied. So, new key distribution mechanisms are required to decrease number of pairwise keys that are deployed into a sensor node.

Random key pre-distribution mechanisms have been proposed to overcome memory restrictions of sensor nodes. These mechanisms are widely accepted for sensor network security. Simply, these schemes try do decrease the number of keys to be deployed in each sensor node in a sensor network and provide reasonable security for the sensor network.

Random key pre-distribution schemes proposed until now have some deficiencies. Some of these schemes are too complicated and too difficult to be applied. Schemes that seem deployable involve unrealistic assumptions when real world scenarios are considered. In this thesis, we propose random key pre-distribution mechanisms that are simple and easily deployable.

In this thesis, we first developed a generalized random key pre-distribution scheme. Then we proposed three random key pre-distribution mechanisms based on this generalized scheme and we provided their simulation results and their comparison to well-known random key pre-distribution schemes in the literature. Our generalized scheme allows different systems to be derived according to deployment needs. It offers simple, easily deployable distribution mechanisms and provides reasonable connectivity and resiliency with respect to its simplicity.

ÖZET

Duyarga düğümleri genellikle değişik alanlara belirli bir tipteki veriyi algılamak maksadıyla dağıtılan küçük, düşük enerjiyle çalışan ve pil gücü zayıf elektromekanik cihazlardır. Hafızaları ve sayısal hesaplama kabiliyetleri kısıtlı olduğundan dolayı açık anahtarlı şifreleme sistemleri (PKC) duyarga düğümlerinin güvenliğini sağlamak için kullanılmaya uygun değildir. Açık anahtarlı şifreleme sistemlerinin yerine özel(tek) anahtarlı şifreleme teknikleri tercih edilmektedir fakat duyarga düğümlerinin hafıza kısıtlarından dolayı hala özel anahtarlı şifreleme sistemlerinin kullanımıyla ilgili sorunlar mevcuttur. Bir duyarga düğümüne yüklenebilecek anahtar sayısı o düğümün eldeki hafıza miktarı tarafından belirlenir ve özel anahtarlı şifreleme yöntemlerinin kullanılmasını da sınırlandırır. Böylelikle bir duyarga düğümüne dağıtılan anahtar sayısını azaltabilecek yeni anahtar dağıtım mekanizmalarına ihtiyaç ortaya çıkmaktadır.

Duyarga düğümlerinin hafıza sorunlarının üstesinden gelebilmek için rastlantısal ön yüklemeli anahtar dağıtım mekanizmaları önerilmiştir. Bu mekanizmalar duyarga ağlarının güvenliğinin sağlanmasında genel kabul görmüşlerdir. Basit olarak bu mekanizmalar her bir duyarga düğümüne yüklenen anahtar sayısını azaltmaya çalışırken aynı zamanda duyarga ağlar için kabul edilebilir seviyede güvenlik sağlamaya çalışmaktadırlar.

Şu ana kadar önerilen rastlantısal ön yüklemeli anahtar dağıtım mekanizmalarının bazı eksiklikleri vardır. Bazıları çok karmaşık, bazılarının ise uygulaması çok zordur. Önerilen mekanizmaların uygulanabilir olanlarının gerçek dağıtım senaryoları düşünüldüğünde gerçek dışı kabullenmeleri mevcuttur. Bu tezde uygulanması ve dağıtılması kolay rastantısal ön yüklemeli anahtar dağıtım mekanizmaları önerilmektedir.

Bu tezde öncelikle genel bir ön yüklemeli anahtar dağıtım şeması önerilmiştir. Daha sonra bu genel mekanizmanın üzerine bina edilmiş üç rastgele ön yüklemeli anahtar dağıtım mekanizması önerilmiş, bunların simülasyon neticeleri sunulmuş ve literatürde iyi bilinen şemalarla karşılaştırmaları yapılmıştır. Genel mekanizma dağıtım ihtiyaçlarına göre farklı şemaların türetilmesine olanak tanır. Ayrıca basit, kolaylıkla dağıtılabilen, kabul edilebilir bağlantı oranı ve dayanıklılık sağlayan mekanizmalar önerir.

To my precious

ACKNOWLEDGEMENTS

I would like to thank my advisor Dr. Albert Levi for his guidance and especially for his patience during this work.

Special thanks are due to Dr. Erkay Savaş and Dr. Özgür Erçetin for their support to this work.

Also, many thanks to Dr. Özgür Gürbüz, Dr. Erkay Savaş, Dr. Kemal Kılıç and Dr. Cem Güneri for their kindness to join my jury.

I must specially thank to my mom for encouraging me and my brother for leading me to have a M.S degree in computer science. Also special thanks to everyone whose names I can't remember.

I have to thank God for giving me a chance with this universe.

TABLE OF CONTENTS

1 INTRODUCTION 1
2 INTRODUCTION TO SENSOR NETWORKS AND SECURITY 3
2.1 Sensor network applications 4
2.1.1 Military applications 5
2.1.2 Environmental applications 5
2.1.3 Health applications 5
2.1.4 Home applications 5
2.1.5 Other commercial applications 5
2.2 Sensor network issues 6
2.2.1 Fault tolerance and security 6
2.2.2 Scalability 7
2.2.3 Production costs 7
2.2.4 Hardware constraints 7
2.2.5 Sensor network topology 8
2.3 Deployment environment 8
2.4 Security background 10
3 PREVIOUS WORK ON SENSOR NETWORK ISSUES 15
3.1 Security issues related to sensor networks 15
3.1.1 Random key pre-distribution schemes without prior deployment knowledge 16
3.1.2 Random key pre-distribution schemes with prior deployment knowledge ..22
3.1.3 Other key pre-distribution schemes 24
3.1.4 Other security schemes 26
3.2 Clustering in sensor networks 29
3.3 Localization in sensor networks 33
3.4 Routing in sensor networks 36
4 PROPOSED RANDOM KEY PREDISTRIBUTION SCHEMES 38
4.1 Design considerations of the proposed schemes 38
4.2 A generalized random key pre-distribution scheme 40

4.3 The first Scheme ABAB 43
4.4 The second scheme ABCD 47
4.4.1 A modification to ABCD scheme: ABCD-Cyclic 52
5 SIMULATIONS AND TEST RESULTS 55
5.1 Definitions 55
5.2 Simulation parameters 57
5.3 Relation between key ring size, connectivity and resiliency 59
5.4 Performance evaluation of proposed schemes 62
5.5 Discussions 72
6 CONCLUSION 75
7 REFERENCES 77

LIST OF FIGURES

Figure 2.1. Integrity provided by hash functions 13
Figure 3.1. Deployment points of batches in the scheme by Du et al 23
Figure 3.2. Key sharing mechanism between zones in the scheme by Du et al 24
Figure 3.3. Security integrated with sensor networks 27
Figure 4.1. Two hundred nodes distributed uniformly random onto a 100x100 deployment area 39
Figure 4.2. Two hundred nodes distributed normally on to a 100x100 deployment area 40
Figure 4.3. Generalized scheme 41
Figure 4.4. Sub key pools in a zone 42
Figure 4.5. Alternating key pool selection of ABAB scheme 44
Figure 4.6. ABAB scheme 45
Figure 4.7. Extending ABAB scheme 46
Figure 4.8. ABCD scheme 48
Figure 4.9. Extending ABCD scheme 51
Figure 4.10. ABCD-Cyclic Scheme 53
Figure 4.11. Extending ABCD-Cyclic Scheme 54
Figure 5.1. Deciding simulation parameters for ABAB scheme 57
Figure 5.2. Deciding simulation parameters for ABCD scheme 58
Figure 5.3. Relation between key ring size and local connectivity 59
Figure 5.4. Relation between key ring size and global connectivity 60
Figure 5.5. Relation between key ring size and resiliency 61
Figure 5.6. Basic scheme and ABAB scheme compared with respect to local connectivity 62
Figure 5.7. Basic scheme and ABAB scheme compared with respect to resiliency by simulation 63
Figure 5.8. Basic scheme and ABAB scheme compared with respect to resiliency analytically 64
Figure 5.9. Key ring size versus local connectivity for four different schemes 64
Figure 5.10. Comparison of analytic and simulation results of ABAB scheme with 33% connectivity 65
Figure 5.11. All schemes are compared with respect to their resiliency 67

Figure 5.12. Resiliency of ABCD scheme with 33% local connectivity.....67
Figure 5.13. Basic scheme and ABAB scheme compared with respect to local connectivity by decreasing variance.....69
Figure 5.14. Basic scheme and ABAB scheme compared with respect to resiliency by decreasing variance.....70
Figure 5.15. Local connectivity examination with different variance samples for ABAB scheme 71
Figure 5.16. Resiliency examinations with different variance samples for ABAB scheme.71
Figure 5.17. Basic scheme and ABAB scheme compared with respect to global connectivity.....72
Figure 5.18. Scheme by Du et al. simulation and analytic results compared 73

1 INTRODUCTION

Wireless sensor networks have a remarkable attention in a few past years. A sensor network involves deployment of a large number of small nodes. These nodes sense data specific to that environment and report them to other nodes over a flexible architecture. Sensor networks are best suited to be deployed in hostile environments and over large geographical regions. In other words, sensor networks are suited to be deployed over unattended areas.

Sensor networks have been useful in various applications such as:

i. Environmental monitoring
ii. Military monitoring
iii. Building monitoring
iv. Healthcare

Sensor networks have broad application areas, and they consist of computationally limited, low-memory and battery constrained microelectromechanical devices. The most important restriction on sensor networks is battery power. The other important restriction on sensor networks is the lack of reasonable amount of memory.

Security that must be provided by sensor network applications is limited because of memory and computational restrictions. Public key cryptography (PKC) techniques are not suited to sensor networks because key sizes of PKC is too big and computation power required is far from an ordinary sensor node can provide. Thus, conventional cryptography (private key cryptography) is more likely to be applied to sensor networks.

Distributing one key to each node requires very little memory but compromise of one node yields compromise of whole network communication. Deploying each node with keys of all other nodes provides very high security but it is not possible for sensor networks with larger number of nodes. The innovation in key distribution for sensor networks is proposed

in [9]. Eschenauer and Gligor proposed a random key pre-distribution scheme that is applicable to sensor networks. Simply a large key pool is generated and each node is loaded with a pre-defined number of keys (key ring) by picking them from the global key pool in uniformly random fashion. All nodes are then disseminated on to the deployment area uniformly. Each node shares some keys with its neighbors with some probability and a securely communicating network can be formed with the key sharing information between sensor nodes. This scheme allows a secure network to be formed by using small number of keys but treats each node to be located at any position with equal probability which is not the case. In [15] Du et al. made use of location knowledge of nodes and a grid-based key distribution scheme is generated. In this scheme a batch of nodes are assumed to be deployed at center points of each cell of a grid. So nodes in the same batch would be close to each other on the deployment area. This simple knowledge enables this scheme to use less number of keys as compared to the one in [9] which is also called as the basic scheme.

The aim of the study in this thesis is to develop a grid-based key distribution scheme which is easily applicable and secure with respect to its simplicity. The scheme is a generalized mechanism that also covers the basic scheme and the scheme proposed in [15]. All these schemes are special cases of our generalized scheme. In other words grid-based key distribution schemes proposed until know can be expressed by our scheme.

Three derivations of our scheme are generated. The first derivation is called as ABAB scheme and makes use of simple location knowledge in order to decrease number of keys deployed in each node. In this scheme deployment simplicity is the main objective. The other scheme is called as ABCD scheme and it makes use of a bit more deployment knowledge as compared to the first scheme and aims further improvement of security. The third derivation is ABCD-Cyclic scheme and it is a variant of ABCD scheme that is specifically designed for allowing enlargement in both directions. These schemes are simulated in order to realize easily applicable and secure key distribution mechanisms.

2 INTRODUCTION TO SENSOR NETWORKS AND SECURITY

Recent advances in wireless communications resulted in development of low power, tiny, microelectromechanical devices. Sensor devices can be described as one sort of those microelectromechanical devices that can be used in the area of environmental, health, battlefield etc applications. One of the best surveys about sensor networks can be found in [1], [2]. These surveys provide valuable information about sensor nodes, sensor networks, and their area of applications, sensor network physical layer aspects, sensor network topologies, and sensor network communication protocols.

In particular a sensor node (sensor node, sensor will be used interchangeably from this point forward) can be described as a low power, tiny, microelectromechanical, computationally restricted device that usually runs on a battery and is capable of sensing information for a specific purpose. A sensor network can be described as a network of several communicating sensor nodes that is deployed for a specific sensing purpose on any area.

Main purpose of a sensor node is sensing, processing and transmission of collected/sensed data. The actual phenomenon of sensor nodes are sensing as the name implies. Sensor networks are prone to failures and because of that reason they are usually densely deployed. Deployment areas of sensor networks can vary from battlefields to state buildings. After this brief introduction to sensor networks and their application areas, more detailed examination regarding sensor network components, sensor network topologies, and usage, deployment areas will be provided.

As a realization of a sensor network application, assume that a greenhouse is being inspected for changes of temperature, water pollutants, and fertilized chemicals. In this application, sensor nodes sense environmental information, in this case, temperature, water pollutants and fertilized chemicals according to a time schedule. After collection of data sensor nodes may determine some statistical information (e.g the highest, the lowest and the mean temperature information) and send it to a controller node (also known as a sink).

The staff responsible for the greenhouse can take necessary actions according to the information sent by different sensor nodes in different locations.

Taking into account the greenhouse scenario above, it is obvious that sensor nodes require wireless communications and networking capabilities. Ad hoc networking techniques may not be well suited to sensor networks because of the differences between ad hoc and sensor networks. Mentioning the differences between ad hoc networks and sensor networks can be a good lead for a better understanding of sensor networks. Differences between these two types of networks can be listed as:

a- Number of nodes in a sensor network may be much more than an ordinary ad hoc network.

b- Sensor nodes are prone to failures.

c- Because of a) and b) sensor networks are densely deployed as compared to ad hoc networks.

d- Sensor nodes are limited in terms of power, computational capabilities and memory.

e- Sensor nodes use broadcast communication mechanism in order to communicate with their neighbors and also communication ranges of sensor nodes are shorter than nodes in ordinary ad hoc networks.

2.1 Sensor network applications

Nodes that are forming a sensor network may be capable of sensing different sorts of data such as temperature, humidity, pressure, movement, soil makeup, etc. Since sensor nodes are manufactured with some sensing capabilities sensor networks are used in very different applications [1]. Some of them are described below.

2.1.1 Military applications

Usage of sensor networks in military applications can be combined as: Monitoring friendly forces, equipment and ammunition, battlefield surveillance, reconnaissance of opposing forces and terrain, targeting, battle damage assessment; nuclear, biological and chemical attack detection and reconnaissance.

2.1.2 Environmental applications

Environmental applications of sensor networks can be combined as tracking the movement of birds, small animals and insects, monitoring environmental conditions that affect crops and livestock, irrigations, forest fire detection, flood detection, bio-complexity mapping of the environment, and pollution study.

2.1.3 Health applications

Some of the health applications for sensor networks provide interfaces for integrated patient monitoring, diagnostics, drug administration in hospitals, tracking and monitoring doctors and patients in a hospital.

2.1.4 Home applications

Sensor networks can be effectively used in home automation. Sensor networks are well suited to home users in order to manage home devices locally and remotely.

2.1.5 Other commercial applications

Some of commercial applications that sensor networks can be used are: Environmental control in office buildings, detecting and monitoring car thefts, managing inventory control, and vehicle tracking and detection.

2.2 Sensor network issues

Issues on sensor networks can be various, because of their low power, communication and computational resources designing a sensor network requires more effort that must be put in contrast to other types of networks. Issues regarding sensor networks can be listed as [2]: Fault tolerance, scalability, production costs, operating environment, sensor network topology, hardware constraints, transmission media, power consumption, and security. The main objective of this work is to design a simple and applicable random key pre-distribution mechanism so the focus of this section will be mainly on sensor network security.

2.2.1 Fault tolerance and security

Sensor nodes may fail due to lack of power and fault of some sensor nodes in a sensor network should not preclude the sensor network fulfilling its main duty.

Actually the level of fault tolerance depends on the purpose of the sensor network. For instance, considering a battlefield deployment sensor network must be much more reliable than any other deployment purpose. This issue can be defined as reliability of fault tolerance. In other words, fault tolerance is the ability to sustain sensor network functionalities without any interruption due to sensor node failures [3, 4, 5].

Fault tolerance is also an important factor in security issues of sensor networks. Security is a fundamental service in many areas of applications. From the security point of view, fault tolerance defines the sustentation of sensor network communication in a secure way without interrupting its main functionalities.

When security comes to mind, physical capture of the nodes is one of the main problems. Sensor networks must be resilient against the physical capture of the nodes. That means, compromise of sensor nodes should not affect the secure communication of sensor nodes and sensor network should sustain secret information to some acceptable degree. There exist many security related problems regarding sensor networks. The acceptable

degree changes accordingly to deployment area, deployment purpose, number of nodes, manageability and security desired. In section 1.5, sensor network security issues will be examined in detail.

2.2.2 Scalability

The number of nodes deployed in a sensor network may be in the order of thousands according to the purpose of deployment. Sensor network schemes must be able to work with that amount of nodes. When the number of nodes increase dealing scalability becomes a real problem. Scalability is not only the problem of managing with such number of nodes but also dealing with extension of the sensor field while providing same level of security and manageability.

Scalability cannot be determined in any measurement without considering security issues. When security is involved scalability becomes a more complicated issue to handle. In this thesis, the proposed scheme aims to improve scalability while keeping security concerns in mind.

2.2.3 Production costs

Sensor networks consist of a large number of sensors as compared to traditional sensors. So it is very important to determine the cost of a sensor network before deployment and if the sensor network is not cost affective, there is no point is deploying a sensor network instead of traditional sensors. The cost of a sensor node must be kept low so that the realization of sensor networks is feasible [6, 7].

2.2.4 Hardware constraints

A sensor node is mainly made up of four basic components.

i. a sensing unit for sensing data

ii. a processing unit for processing received data
iii. a transceiver unit for wireless communications
iv. and a power unit

The most important constraints on sensor networks are battery power, processing power and memory size. While security in mind, constraints on processing power and memory are the most important determiners of the security schemes that are to be deployed. For instance, memory size is very important to determine the key size and number of keys to be deployed. Moreover because of processing power constraints, traditional cryptography is more suitable to be applied as compared to public key cryptography.

2.2.5 Sensor network topology

There is no predefined network topology for sensor networks. In other words, there is no particular infrastructure specially designed for sensor networks. After deployment of sensors onto the target area a properly communicating network is formed usually in a hop by hop fashion. Each node communicates with the nodes in its neighborhood (one hop neighbors) and communication with other sensors is achieved by the help of neighboring sensors. Such networks can be called as “infrastructureless”.

2.3 Deployment environment

With respect to the purpose of sensor network applications, their deployment areas would change. Except for deployment schemes done by hand, usually deployment areas are unattended. As a list of sample deployment areas [1], please see below

- The bottom of a sea or an ocean,
- On the surface of a sea or an ocean,
- In a building or a warehouse,
- In a drain or river moving with current,
- In a biologically or chemically contaminated field,

- In a battlefield beyond the enemy lines,
- Attached to animals,
- Attached to fast moving vehicles.

Since deployment areas are different, sensor node properties should also be different. For instance, a node that is deployed behind an enemy line should be capable of communicating even if the communication lines are noisy. In another case, the nodes under the water should be resistant to high pressure and water proof. Various kinds of sensor nodes can be manufactured to be used in very different applications. The term “sensor node” does not refer to a single type of device but it refers to a device that can be manufactured for different purposes. The only generalization that can be made about sensor nodes is that they are manufactured for sensing data, as the name implies. Any scheme to be used with sensor networks such as routing, security, etc. should consider those aspects of deployment environments. Assume that deployment takes place on habitat of some insects, deployment schemes proposed until now may not be suitable and new schemes may be needed. For instance, aerial scattering may not be suitable for deployment and sensors should be disseminated from a moving vehicle such as a truck. In this case the density of nodes and the path of deployment must be determined by a different scheme. As a result it can be said that, since there are different application areas for sensor networks and there are very different areas to deploy, it is obvious that there should be different schemes for routing, security etc.

There exist different aspects of sensor networks such as transmission media, power consumption, communication protocols, protocol layers and data processing. These concepts are all in relation to sensor networks but not too much concerned with the idea in this thesis. Data processing is an important concept in sensor networks. If a few words needed on data processing; energy consumption in data processing is much less than consumption in data communication. Any scheme (routing, security, etc) should be able to decrease the communication among sensor nodes in an efficient manner such that sensor nodes can sustain functioning properly for a longer time. Extensive information is provided in [1] and [2].

2.4 Security background

In order to explain some concepts about security issues regarding sensor networks a simple introduction to security primitives is needed. In this part, a brief explanation of some security concepts is provided.

Main security services can be listed as:

- Authentication: Authentication can be simply described as proof of identity. Assume that two parties are communicating with each other, if one party can assure that the other party is the really the one it claims to be, and then authentication is provided. As a realization of the concept, assume that Alice wants to open the door of a laboratory that is protected by a fingerprint mechanism. If Alice is an authorized one then her fingerprint must have been registered and she should be able to open the door with her fingerprint. In other word, Alice authenticates herself with her fingerprint.

- Data Confidentially: Data Confidentiality can be described as protection of data from unauthorized disclosure. For instance, assume that Alice wants to send a message to one of her friends Bob. Alice should make sure that no one other than Bob can read her message. So she puts her message in a box and locks the box with a key. She gives an identical key to her friend Bob and nobody other than Alice and Bob has an identical key. Bob opens the box with the key and reads the message. He is sure that nobody other than himself can read the message.

- Data Integrity: Data integrity is the assurance that data is received exactly as sent. For instance, assume that Alice sends a message to one of her friends Bob and she must make sure that the message she sent was not altered on the way to her friend Bob.

Cryptography is the term that refers to “act of secret writing”. Writing in a secret way can be achieved by use of a secret key. Secret keys are nothing other than sequence of bits

that is known only to authorized parties. Keys are used in different cryptographic algorithms such that a plain text is converted in such a form that it cannot be read without the reverse operation with the same key applied to the cipher text. Data confidentiality can be provided by secret keys as long as the secret key is not known to any unauthorized party which explains why this cryptographic technique is defined as "secret". For instance, assume that Alice and Bob share a secret key. Alice sends a message to her friend Bob encrypted by a secret key, and no one other than Alice and Bob can open the message and cannot read the message, so that confidentiality is provided. In a formal way:

K : Private Key
P : Plain Text
C : Cipher Text
E : Encryption
D : Decryption

$$E_K(P) = C$$
$$D_K(C) = P$$

Public key cryptography is another technique that uses two different but mathematically related keys for encryption and decryption. These keys are a public key that is freely distributed to everyone and a secret key that is known only to the owner. A plain text that is encrypted under the public key can only be decrypted by the corresponding private key, so no one other than the owner of the private key can read the message. Data confidentially is achieved. If a plaintext is encrypted under the private key than encrypted message can be freely disclosed by anyone who owns the public key. This technique proves that the message is originated from the owner of the private key since the private key is known only that person. Actually this technique is known as "digital signature"; the message is signed by the owner of the private key that proves his/her identity, so authentication is provided if the message is not a replay.

Key sizes of public key cryptography technique are larger as compared to key sizes in secret key cryptography. Also computational overhead of public key cryptography is greater than secret key cryptography, and public key cryptography is not suitable for bulk

encryption. Secret key cryptography and public key cryptography are not substitutes of each other. Public key cryptography is usually utilized in exchanging secret keys, and signing messages (digital signatures). Large key sizes and computational overhead of public key cryptography makes it inefficient to use with sensor nodes, so conventional key cryptography has to be preferred to be used with sensor networks.

Until now, examples to confidentiality and authentication are given. In order to give an example to data integrity hash functions should be explained. A hash function is a function that converts any length of input to a fixed size unique output. Actually the output is a fingerprint of the input. Whenever a message is to be sent to another party, a hash of message is calculated, the original message is encrypted under the key. The hash is appended to the original message and sent to other party. The receiver decrypts the message under the key and calculates the hash of the decrypted message, this hash and the hash appended to the message are compared, if they match then the message is not altered in some way and integrity is provided. Figure 2.1 depicts the way integrity is provided by hash functions.

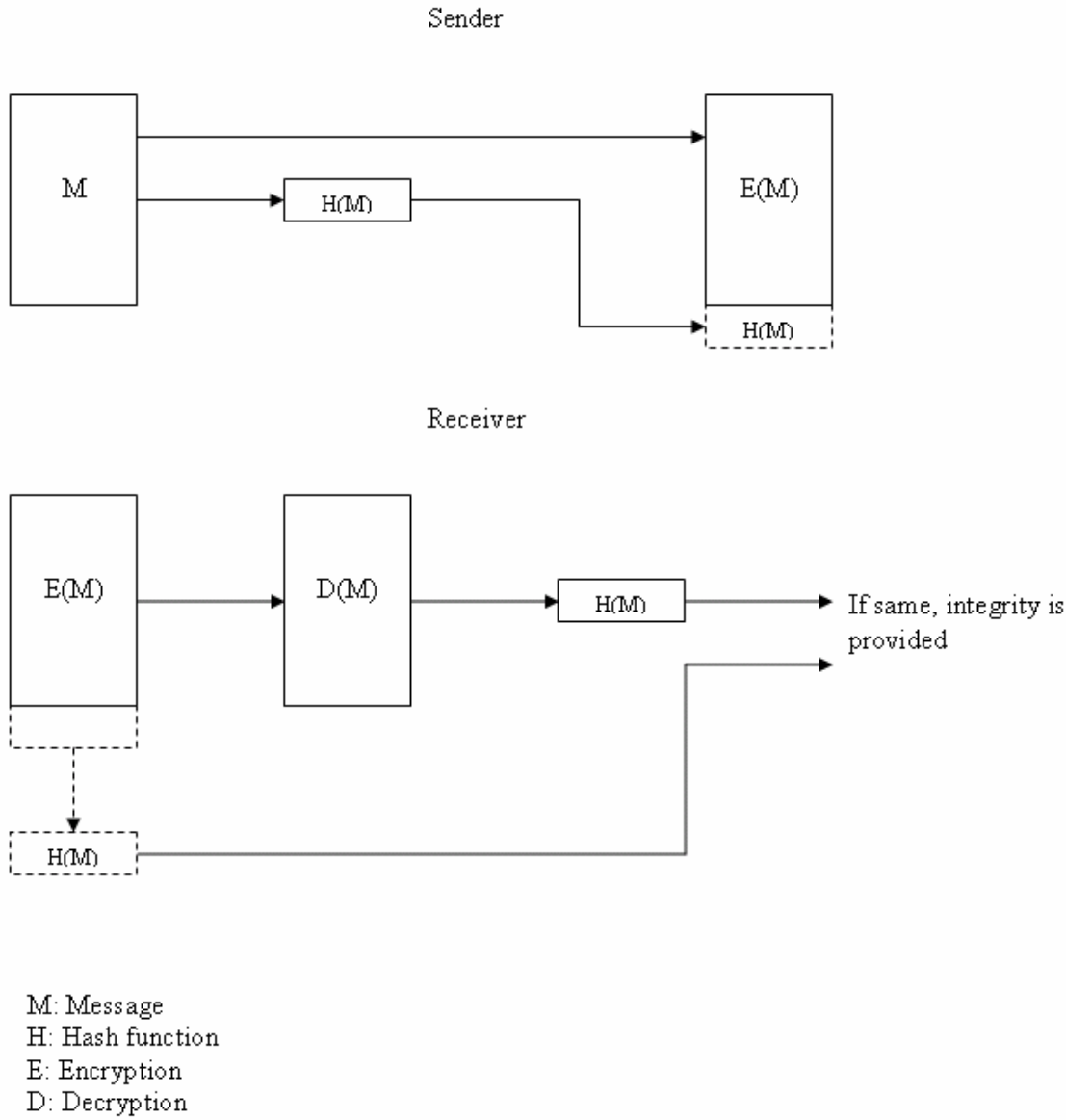

Figure 2.1. Integrity provided by hash functions

Key agreement is another fundamental issue in security. Actually key exchanging protocols based on public key cryptography is not suitable for sensor networks, widely accepted approach is pre-distributing symmetric keys in sensor nodes before deployment.

Previous work on sensor network applications can be grouped in many different areas but, here, focus will be on security issues, routing issues and clustering issues. These topics are all the major concerns related to sensor networks as in many other types of networks.

The main objective of this thesis is to propose a random-key deployment scheme so the main emphasis will be on security issues regarding sensor networks. This part is intended to give a deep understanding of these three concepts in sensor networks. One step forward is the description and detailed explanation of the proposed approach by the author.

3 PREVIOUS WORK ON SENSOR NETWORK ISSUES

In order to explain all the concepts in this thesis a brief explanation of security concepts frequently referred and an overview of the previous work should be presented. After providing vital security concepts most of this chapter is dedicated to previous work on random key pre-distribution schemes and other security mechanisms for wireless sensor networks.

3.1 Security issues related to sensor networks

Security is a fundamental service in many applications and sensor network applications are not exceptions, so solutions to this fundamental issue will be examined and discussed throughout this section.

Resurrecting duckling proposed in [8] refers to ad hoc sensor wireless networks, and it is useful to realize the innovations presented in this work because it covers security issues regarding devices with short range radio coverage. The main idea is that “a duckling emerging from its egg will recognize as its mother the first moving object it sees that makes a sound, regardless of what it looks like: this phenomenon is called imprinting“. When this reality is applied to a transceiver it becomes: When a transceiver initially boots it will belong to the first device it communicates and will stay imprinted to that device until the imprinted device tells it to die, also it can accept a key from just the imprinted device until the duckling dies (e.g. it is out of service). When the ducking boots again it is ready for finding another device to imprint. In this approach, devices contact each other in a close distance such that no cryptography occurs during the transfer of the secret.

This idea is an innovation in the area of short range wireless communications because it offers a scheme that is easy, applicable and cheap. There still exist problems with this scheme such that temper-proofness or temper-evidentness. The idea is based on physical contact and a natural fact “imprinting”. Even it seems applicable to sensor networks, physical contact of sensor nodes on an unattended area is not possible but on attended and

relatively small areas it seems possible and applicable. So, new schemes are needed to deploy sensor networks especially on unattended and large areas with large number of sensor nodes.

3.1.1 Random key pre-distribution schemes without prior deployment knowledge

The most important innovation in key distribution regarding sensor networks is proposed by Eschenauer and Gligor in [9]. This scheme is called as the basic scheme and it was subject to various improvements. From this point forward, this scheme and its consequences and affects on key distribution will be examined.

Most of cryptographic techniques cannot be applied to sensor networks because of computational capabilities, and memory restrictions of sensor nodes. For instance, public key cryptography is not suitable to be applied to sensor nodes. Many sensor node applications restrict the cryptography limited to conventional cryptographic (private key cryptography) techniques. Because of this reason, key distribution becomes a very hard problem to solve. A KDC (Key distribution center) may not solve these problems effectively because sensor nodes are usually deployed on unattended areas which makes key distribution task of KDCs harder. However KDC and PKC (public key cryptography) based solutions are not applicable to sensor networks, pre-deployment of keys to sensor nodes seems quite applicable while remembering the idea behind the sensor networks. Pre-distribution of keys to sensor nodes before deployment still has problems. Distributing a global key is not suitable since capture of one node will compromise the whole network. Distributing one key for each sensor node is not possible even for other types of networks that are well-equipped in terms of memory. So, another key distribution mechanism is needed that requires less memory and still secure. Randomization seems to be a way of achieving this task.

In basic scheme there exists a large key pool P which a pre-defined number of keys (key ring) k are picked from to be loaded into each sensor node. Remember that P is generated offline. In other words, k numbers of keys are picked in a uniformly random fashion from a large key pool P and pre-loaded into each sensor node. This is the first step

and called key pre-distribution phase.

The second phase is called shared key discovery and takes place just after deployment. Each node on the deployment area discovers its neighbors in its wireless communication range. An easy way of achieving it is to broadcast key identifiers to all neighbors in plain-text. Another mechanism that is secure for broadcasting key identifiers is described below:

Each node broadcasts a list of key identifiers $\alpha, E_{K_i}(\alpha), i = 1,...,k$, where α is a challenge. The decryption of $E_{K_i}(\alpha)$ with the proper key by a recipient would reveal the challenge α and establish a shared key with the broadcasting node. If two neighboring nodes share a secret key a link is established between these two nodes. A key that is used to secure the communication between any two nodes can also be used to secure the communications between other pair of sensor nodes. Compromise of a key requires the revocation of this key over the whole network, making the links unusable secured by that key.

The third phase is path-key establishment. In this phase each node tries to establish a link between its neighbors that are in communication range but does not have at least one link. Path-key establishment phase is done via the links of each node, in other words a node tries to establish a link with its neighbor by the help of its secure neighbors in two or more steps. Its secure neighbors may share a key with that node and send one of its keys over those links. Shared-key discovery phase has to be finished in order to begin path-key establishment phase.

DSN (Distributed sensor network) connectivity has the major importance in this scheme. After deployment all sensor nodes must be able to find a secure neighbor, and all these secure neighbors must form a connected graph. In this case, the network is connected but, it is not needed to be fully connected (each node can establish links with its all neighbors in its communication range after completion of shared key discovery phase, without needing the path-key establishment phase) since path key establishment phase is mainly aims to generate a fully-connected network.

Let p be the probability that a shared key exists between two sensors, n be the number of network nodes, and $d = p*(n-1)$ be the expected degree of a node (the average number of edges connecting that node with its graph neighbors). In order to establish the desired connectivity two important measures must be examined carefully.

- expected degree of a node, d, such that a DSN of n nodes is connected
- given d and the neighborhood connectivity constraints imposed by wireless communication, values of k, and pool P must be determined to have a connected network of size n.

Random graph theory helps to determine d stated in the first entry. A random graph $G(n, p)$ is a graph of n nodes such that the probability that a link exists between two nodes is p. When p is zero there is no edge in the graph, whereas when p is one, the graph is fully connected. The value p must be such that $G(n, p)$ is connected.

Erdos and Renyi [10] showed that, for monotone properties, there exists a value of p such that the property moves from nonexistent to certainly true in a very large random graph. The function defining p is called the threshold function of a property. Given a desired probability P_c for graph connectivity, the threshold function p is defined by:

$$P_c = \lim_{n\to\infty} \Pr[G(n, p) \text{ is connected}] = e^{e^{-c}}$$

where

$$p = \frac{\ln(n)}{n} + \frac{c}{n}\ ,\ c \text{ is any real constant.}$$

Therefore, given n, p and $d = p*(n-1)$ can be found with desired probability P_c.

Wireless communication constraints may limit neighborhoods to $n' << n$ where n' is

number of nodes in a neighborhood. This implies that the probability sharing a key between any two nodes in a neighborhood is $p' = \frac{d}{(n'-1)} >> p$

So the probability that two nodes share at least one key in their key rings of size k chosen from a given pool of P keys to p' and then derive P as a function of k. To derive the value of P, given constraint k for a p' that retains DSN connectivity with an expected node degree d note that $p' = 1 - \Pr[\text{two nodes do not share any key}]$ and thus

$p' = 1 - \frac{((P-k)!)^2}{(P-2k)!P!}$ since P is very large, using Starlings' approximation

for $n!$

$n! \approx \sqrt{2\pi} n^{n+\frac{1}{2}} e^{-n}$ to simplify the expression of p' and obtain:

$$p' = 1 - \frac{(1-\frac{k}{P})^{2(P-k+\frac{1}{2})}}{(1-\frac{2k}{P})^{(P-2k+\frac{1}{2})}}$$

For examples, the reader may refer to [9].

Various schemes based on the basic scheme have been developed so far. From this point forward, some necessary information about those schemes will be provided, the basic scheme is inspected in detail because it is the basic of the whole work so far.

There still exists an important problem with the basic scheme. Only large random graphs are considered of which nodes are uniformly distributed over a deployment area. Such an assumption is not realistic and realization of such distribution can only be possible in deployment areas on attended areas, done by humans or robots. In other words, the

scheme itself is assumed to be too uniform to be realized.

Since the memory resources of sensor nodes are restricted, large key rings are needed for networks with large number of sensor nodes which does not seem to be applicable to real sensor nodes when their capabilities are considered nowadays.

In [11] there are three schemes proposed based on the basic scheme. The first scheme proposed is "q-composite scheme" which imposes q as a security parameter to the network in the following way: Sensor nodes must establish a secure link when they share at least q number of keys. If neighboring nodes share less than q keys than a secure link is not established between these sensor nodes. In this scheme the idea is to make the network resilient against node capture, but it is obvious that in order to apply q-composite scheme, neighboring nodes should share more keys as compared to the basic scheme if the global key pool size P is the same for the both of the schemes. In other words, key ring size k must be increased to realize the q-composite scheme, so q-composite scheme is only applicable when small number of nodes is assumed to be captured. When large number of nodes is captured, this scheme is not applicable since capture of one node reveals more keys as compared to the basic scheme as already stated in [11]. The other scheme proposed is called as "Multipath Key Reinforcement". When the basic scheme is considered a key that is used to secure the communication between two nodes can also be used to secure various communication links through the network which spreads the compromise through the network. In order to overcome this situation "Multipath key reinforcement" is proposed assuming that enough routing information is available. Assume that a node A knows all the *disjoint* paths to node B. Specifically, $A, N_1, N_2, ..., N_i, B$ is path created during the initial key setup if and only if each link $(A, N_1), (N_1, N_2), ...(N_{i-1}, N_i), (N_i, B)$ has established a link during the initial key setup using the common keys in the nodes' key rings. Let j be the number of such paths that are disjoint (Do not have any links in common). A then generates j random values $v_1, ..., v_j$. A then routes each random value along a different path to B. When B has received all j keys, then a new link key can computed by both A and B as:

$$k' = k \oplus v_1 \oplus v_2 \oplus ... \oplus v_j$$

In that way, the link is secured by contribution of j random values. In order to overcome the security threats that eavesdropping imposes over the network is lowered by this way. But the communication overhead that this scheme imposes is not insignificant, both the network topology and the communication overhead are significant drawbacks of this scheme. Even as stated in [11] 2-hop multipath key reinforcement may be applicable since discovering disjoint paths more than two hops is infeasible and q-composite scheme should not be applied at the same time with multipath key reinforcement since compounds both schemes' weaknesses. Small key ring size requirement of q-composite scheme weakens the multipath key reinforcement scheme.

The last scheme proposed in [11] is "Random-pairwise keys scheme" that introduces node to node authentication. A key can be used to secure various communication links, so a node should be certain of communicating with the right node. In order to achieve authentication a node identifier is created for each node and each node identifier is matched up with k other randomly selected distinct node identifiers. Also a pairwise key is generated for each pair of nodes and stored in the key rings of both nodes along with the identifier of the other node. This idea is to ensure that the other node is a legitimate node and also this scheme does not allow reuse of the same key by multiple pairs of sensors.

Until now, key distribution schemes designed for sensor networks have been mentioned. Most of these schemes are based on the idea presented in the basic scheme. Keeping the same idea in mind, there are other schemes proposed. Especially the mathematical structure of keys is prone to changes. There are schemes that mainly focus on the key structure, and try to improve the basic scheme. From this point forward, a brief look at these schemes is necessary to give a better understanding of the concept.

In the basic scheme there is no information about the structure of the keys such that a key is just a piece of secret information to secure the communication between sensor nodes. Mathematical structure of these keys affects the key size, key ring size, global key pool size, local connectivity and resiliency against node capture.

Liu and Ning [12] proposes a scheme that is a generalization of the basic scheme and resilient against node capture. This scheme is called "polynomial pool-based key predistribution". As the name implies, there exists a polynomial pool and there are no keys deployed in the sensor nodes, instead polynomial shares of a set of bivariate t degree polynomials are deployed in sensor nodes. Mathematically:

A set F of randomly generated s bivariate t-degree polynomials over the finite field F_q. For each sensor node, the setup server randomly picks a subset of s' polynomials from F and assigns polynomial shares of these s' polynomials to the sensor node.

Sensor nodes discover whether they own a share of the same polynomial and generate the key to be used to secure to communication between them. This scheme is resilient against node capture, since in order to compromise the network, t number of sensor nodes must be captured which is not easy to achieve since the polynomial shares are distributed randomly. So the network size is limited to $\frac{(t+1)s}{s'}$ nodes. Number of nodes in that sensor network cannot exceed that number of nodes. This scheme can be scalable since new nodes can be added dynamically to the network as long as the limit on the number of nodes is not exceeded. Also another scheme "A Pairwise Key Pre-distribution Scheme for Wireless Sensor Networks" [13] is based on Blom's Key Pre-distribution scheme [14] which resembles to the idea presented in [13].

3.1.2 Random key pre-distribution schemes with prior deployment knowledge

Schemes that briefly examined until now do not assume any deployment knowledge. All sensor nodes are assumed to be deployed on a field with no prior deployment knowledge. Most of the time this assumption is not the case since even the nodes are deployed via aerial scattering, there exist knowledge where a sensor node approximately resides which can be utilized to decrease the key ring size of a sensor node carries. The most remarkable one is proposed by Du et al in [15]. This scheme assumes a grid deployment scheme such that nodes are deployed on a grid and distribution of nodes in

each zone is Gaussian.

Nodes are assumed to be deployed in the center of each zone in the form of a batch. Those batches of nodes are distributed over each zone normally. Normal distribution is assumed to best fit the real world deployment scenarios (e.g. aerial scattering). Keys are distributed to each node uniformly by selecting from the key pool of the corresponding zone. But the key distribution mechanism is quite complicated and does not scale. Each zone shares some percent of keys with its neighbor zones and all that key sharing computation is offline. With respect to those complications the scheme provides high security and resiliency against node capture. Figure 3.1 depicts the deployment points on a 5x5 grid.

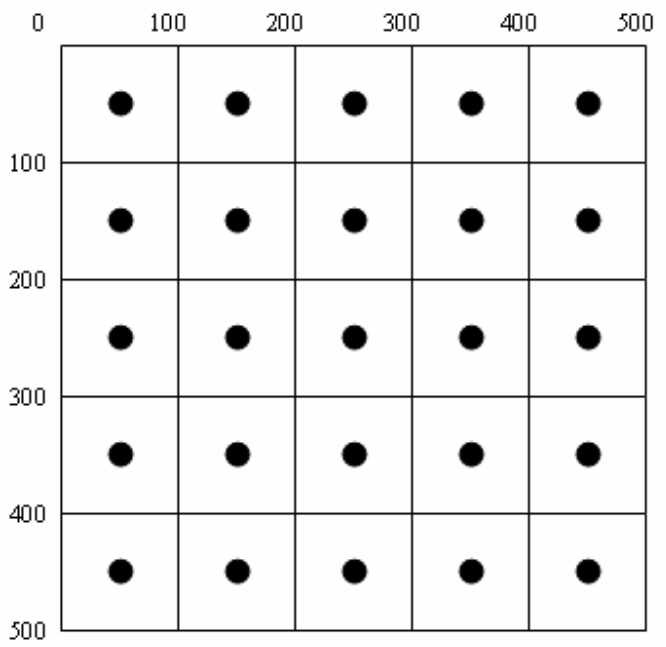

Figure 3.1. Deployment points of batches in the scheme by Du et al

Dots in Figure 3.1 are target deployment points of batches and nodes in each batch are normally distributed over that zone.

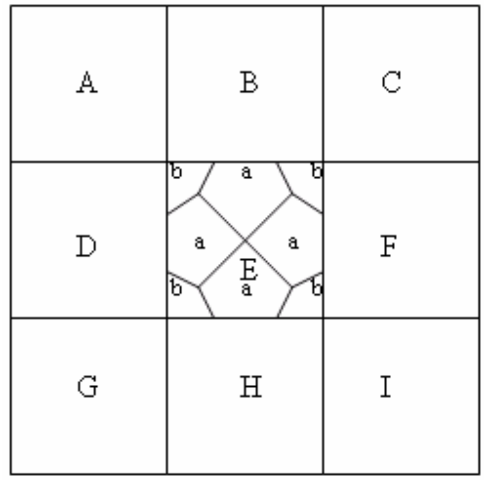

Figure 3.2. Key sharing mechanism between zones in the scheme by Du et al

Figure 3.2 depicts key sharing scheme between zones. Zone "E" shares its "a" percent of keys with zones B, D, H, F and "b" percent of its keys with zones A, C, G, I. Zones that are not neighbors do not share any keys so intuitively the number of links that are secured via the same key are decreased. This scheme offers improved resiliency against node capture and decreases the key ring size remarkably but it is too complicated in terms of pre-computation steps. Also the ability of a sensor network to scale heavily depends on the pre-allocation of keys for zones on the corners and residing on the edges which is not proposed in this scheme. A generalized scheme that also covers the capabilities of this scheme will be proposed in this thesis. For a more detailed explanation of key distribution and key pool generation please refer to [15]. Another scheme that makes use of deployment knowledge is presented in [16] which assumes a uniform distribution of keys and nodes in a zone, and pairwise pre-deployed key sharing knowledge. This scheme provides remarkable security and resiliency against node capture but distribution of nodes and keys in a zone does not seem applicable when real deployment scenarios are considered and also scaling the sensor network is not possible.

3.1.3 Other key pre-distribution schemes

Another key pre-distribution scheme is proposed in [39]. This scheme proposes an innovative approach for key pre-distribution. In this scheme different keys are logically mapped to a two dimensional space and position of each node is output of a probability

density function. In other words, each node is assumed to be located at a position from a pdf (probability density function) thus each node's positions are restricted to a sub-area and so that the number of keys that should be deployed in each node is aimed to be reduced. In other group based deployment models nodes are distributed according to a pdf but there is no key position mapping. The distribution mechanism is executed as follows:

i. Deployment area is divided in to cells and each cell is mapped to a key.
ii. The expected distribution position P of a node is computed from the probability density function.
iii. A circle with radius r is drawn and a node Q that resides in that circle is picked in a random fashion.
iv. The key that is owned by Q is assigned to P.
v. If this key is already contained in P then go to step ii and continue.

Such an approach gives better results than [15] and ours but there exists other problems with this scheme. For instance, if large sensor networks are considered such deployments do not seem possible because for each node that probability density function must be computed which means each node is assumed to be deployed by hand in order to be realize the deployment with the proposed scheme. Schemes that assume deployment of nodes in batches do not differentiate the nodes in deployment manner but this scheme allows each node to be placed according to the pdf which means that in real deployment scenario each node must be treated individually or this approach is not stated clearly by the authors of the scheme. Another issue regarding this scheme is that iterating the distribution algorithm for each node in order to load all keys into sensor nodes before deployment is not an easy task especially when it is compared to batch based distribution schemes. But the idea presented in this scheme is an innovation and may be applicable to sub-group of sensor nodes in each batch. This way, instead of positioning each node according to a pdf, sub-group of batches may be deployed by applying the algorithm presented in this scheme.

Another deployment scheme is proposed by Mao and Wu [40]. In this scheme square and hexagon lattice deployment models are proposed for deployment of nodes on to a target area. The contribution of this study is that it proposes a sensor location update

mechanisms to optimize sensing coverage and secure connectivity. Square and hexagon lattice deployment models aim improving the sensor coverage but in reality these deployment models does not seem to be applicable even with few number of nodes. The contribution of this work is the proposal of a location update scheme in order to both improve the node coverage and secure connectivity under the assumption that sensor nodes are mobile in some manner. For further details please refer to [40].

Key pre-distribution schemes for wireless sensor networks can be divided into many categories. But current approaches mainly focus on group based deployments as in our case. Group deployment models enable schemes to have more chance of increasing local connectivity with deploying less number of keys since grouped keys are more likely to be neighbors on the deployment area. Schemes presented in [15], [16] are both group based deployment models and offer considerable security with less number of keys used in each node. Zhou et al. proposed another key establishment mechanism for group based deployments in [46]. This scheme proposed an approach such that each node in a group of nodes shares one key with every other node in the same group. Also inter group key establishment is achieved via some agents. There exists pair of agents in two neighboring groups such that neighboring sensors from these two groups can establish pairwise keys using these pair of agents as intermediaries. This scheme offers high resiliency because it is a scheme mainly based on pairwise rather than random key pre-distribution. Actually, pairwise key distribution has a drawback. In this type of deployment each node has to carry the keys of other nodes in the same group. Number of nodes contained in a group is a major determiner of the applicability of these schemes. Groups with large number of nodes are not suitable to be deployed in this manner because of the memory constraints of sensor nodes.

3.1.4 Other security schemes

Distributing keys to sensor nodes in a sensor network is not the only problem to be solved from the security point of view. Authentication, data confidentiality and integrity are other security issues to be solved. There exist two schemes that are novel in sensor network security area. SNEP (Secure Network Encryption Protocol) and μTESLA (the micro

version of Timed, Efficient, Streaming, Loss-tolerant Authentication) are proposed in [17]. SNEP is a protocol that provides data confidentiality, two-party data authentication, integrity and freshness. µTESLA is a protocol that is based on delayed key disclosure and provides broadcast authentication based on TESLA [19]. TESLA is not originally designed for sensor networks µTESLA is a modified version of TESLA that is applicable for sensor networks. Actually key distribution is the first building block of the security service that should be provided by the security architectures proposed for sensor networks. Authentication, data confidentiality, integrity are the other building blocks that should be based on key distribution. After key distribution, appropriate authentication and data confidentially mechanism can be applied. This idea is best represented in [20] and a depiction of the hierarchy is available in Figure 3.3.

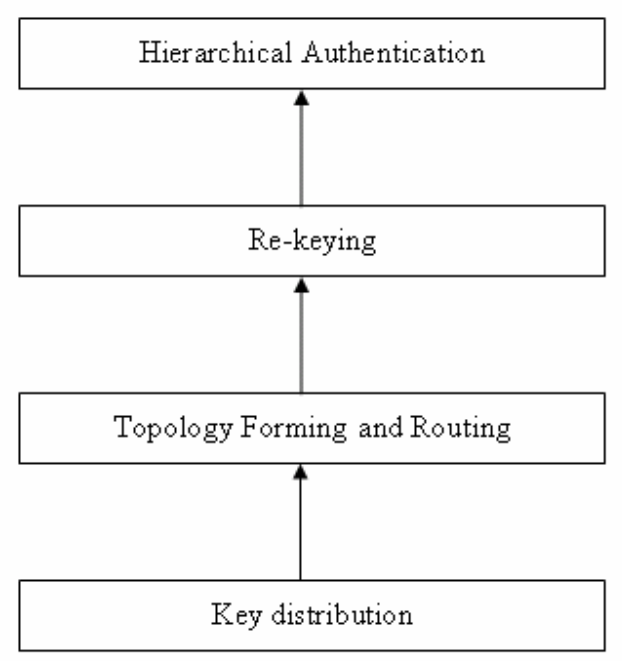

Figure 3.3. Security integrated with sensor networks

After key distribution, keys are pre-loaded into sensor nodes in this case, a topology forming algorithm is executed and in the formed topology a re-keying algorithm can be run. As the last step µTESLA can be used to provide hierarchical authentication service.

There exist other protocols to manage keys that are based on super nodes as in [18]. There are only two keys to be deployed in each sensor node and assumption is the existence of

super nodes a main controller located on an attended and secure area. Such protocols can be applied for relatively small sized networks but is not applicable for large networks (e.g. sensor networks with thousands of nodes). Another mechanism proposed is LEAP (Efficient Security Mechanisms for Large-Scale Distributed Sensor Networks) [21]. This scheme explains passive participation and introduces four types of keys. The motivation is that, there are different types of messages available so there should be different type of keys to be used. Types of keys used are:

- Individual key: Each node has a key that is only shared with the base station to communicate in a secure way.

- Pairwise Key: Each node has a pairwise key shared with its each neighbor in order to communicate securely.

- Cluster Key: Cluster key is shared with a group of neighboring nodes in order to make passive participation available such that a node overhearing the message of one of its neighbors can use it without receiving the same information in another secure message.

- Group Key: A key shared by all nodes in the network and used for network-wide messages.

The idea is novel, but LEAP suffers from an expensive bootstrapping phase. In LEAP starting from a master key every node creates a cluster key that distributes to its immediate neighbors using pairwise keys that shares with each one of them. This scheme offers deterministic security and broadcast of encrypted messages. In [47], a new key management scheme is proposed in order to provide deterministic security for wireless sensor networks. This scheme assumes three types of keys Node Key, Cluster Key and Master Key. Each node shares a node key with the base station; a cluster key is shared with each cluster head and with the base station. Master key is a key that is shared among all nodes throughout the whole network. A simple clustering algorithm is also proposed in this study. Both schemes try to come up with a deterministic key management scheme such that

no random key pre-distribution is necessary. Such schemes mainly base their assumptions on base stations and clusters. Clustering yields good results in key management since key management task is handled as sub tasks, association of keys with base stations in strict manner may not be so correct though. Pairwise keys are important to communications between sensor nodes which is not taken into consideration in [47]. Pairwise key management is taken into consideration in LEAP but, as it is mentioned in this section, it suffers from an expensive bootstrapping phase which is not the case in random key pre-distribution schemes. Key management mechanism that is to be used heavily depends on the aim of the deployment.

3.2 Clustering in sensor networks

Clustering in sensor networks is an important issue to be resolved. Clustering is a well-known problem that is studied in the area of distributed computing in order to solve different problems. Especially clustering is an important area of study in sensor networks that is to solve communication overhead problems. Optimization of communication bandwidth is an important issue since sensor network communication bandwidths are limited and also battery constraints of sensor nodes make clustering an area such that considerable effort must be put in. These two constraints, battery power and communication bandwidth, lead to development of clustering schemes that try to prolong the network life time.

Proposed schemes in this thesis employ group based distribution of sensor nodes over a target deployment area. All these groups actually form clusters over the deployment area. In other words, even a specific clustering algorithm is not run after deployment because of the deployment scheme itself there exist clusters in the whole network. Before deployment, a piece of location knowledge is bound to sensor nodes and on the deployment area a cluster formation can be assumed because of the nature of group based deployment. In this thesis there is no specific clustering algorithm employed but since clustering plays an important role in sensor network applications, an examination of clustering algorithms are provided.

Since sensor nodes are deployed on to unattended areas and because of the distributed

nature of the sensor networks, clustering schemes should work in a distributed manner. Cluster head selection and further management of clusters should be achieved without intervention of a third party computationally powerful device which means a cluster head is an ordinary sensor node. Cluster heads should not be assumed as nodes that have extensive computation and battery power.

In [41], LEACH (Low energy adaptive clustering hierarchy), an application-specific protocol architecture is proposed. The aim of this protocol is to prolong the network life time and evenly distribute the energy load to each sensor. LEACH is a distributed clustering algorithm that uses a probabilistic model to select a cluster head. LEACH also makes use of a slotted algorithm to reduce the energy consumption of sensor nodes and the algorithm operates in rounds. Basically the protocol works as follows:

- Each node probabilistically determines whether it will be a cluster head or not.
- Nodes that elect themselves as cluster heads announce that they are cluster heads.
- Each node-cluster head node waits for the cluster head announcements and send join request to the cluster head that requires the lowest-communication energy.
- Cluster heads create TDMA schedule and send to cluster members.

The above algorithm iterates in rounds so that cluster heads are chosen with high energy and energy load of sensor nodes are evenly distributed throughout the sensor network. Each cluster head is elected according to its remaining battery power formally according the formula below:

$$P_i(t) = \min\{\frac{E_i(t)}{E_{total}(t)}k,1\}$$

Probability that node i elects itself as a cluster head in a round, at time t is computed as above where k is the number of clusters in the sensor network and $E_i(t)$ stands for the energy of node i and $E_{total}(t)$ stands for the total energy of the network. TDMA schedule gives each node a slot to transmit its data to its cluster head and then the node goes to sleep to reduce the energy consumption. Cluster heads send their processed data to BS (Base

Station). The most important contribution of this algorithm is to distribute energy dissipation evenly throughout the whole network thus it increases the system life time. The proposed scheme does not include a mechanism to choose the percentage of cluster heads for a network. Another issue with this scheme is when to invoke re-election of cluster heads. No explanation is given about this issue. Also BS in the scheme can be considered as an aggregation point, and cluster heads that are away from BS will dissipate more energy than the cluster heads that are closer to BS, so life time of cluster heads that are away from BS will be shorter. An approach to solve this problem may be to use a higher level of cluster hierarchy as in [36] other than involving only one cluster head.

Another probabilistic distributed clustering scheme is proposed by Younis and Fahmy [38]. This scheme is a hybrid algorithm that periodically selects cluster heads according to a hybrid of their residual energy and a secondary parameter, such as node proximity to its neighbors or node degree. The algorithm is as follows:

- Each node determines whether it will be a cluster head probabilistically.
- Each cluster head candidate announces itself as a candidate or a final cluster head according to the probability in the first round.
- An ordinary node that receives cluster head announcements determines its candidate cluster head according to the power needed to communicate with the candidate cluster head. If power levels are the same for each candidate then the candidate cluster head with the lowest node degree is chosen as the candidate. If a final cluster head message is received then the node ordinary node joins that cluster.
- Candidate cluster heads doubles their cluster head election probability ands goes to the first step.

After execution of above steps if a node is left uncovered then it announces itself as a final cluster head.

Probability of becoming a cluster head is determined as:

$$CH_{prob} = C_{prob} \times \frac{E_{residual}}{E_{\max}}$$

C_{prob} stands for the initial percentage of cluster heads among all nodes. C_{prob} is only used to limit the initial cluster head announcements where $E_{residual}$ is the estimated current residual energy in the node, and $E_{\max}$ is a reference maximum energy (corresponding to a fully charged battery)

An energy efficient hierarchical clustering algorithm for wireless sensor networks is proposed by Bandyopadhyay and Coyle [36]. The proposed algorithm is not much different from the previously proposed clustering algorithms in terms of probabilistic election of cluster heads and decrease in the energy used. In this algorithm there are two types of cluster heads, *volunteer* cluster heads and *forced* cluster heads. Each sensor in the network becomes a cluster head with probability p and advertises itself as a cluster head to the sensor nodes in its wireless communication range. These nodes are *volunteer* cluster heads. Each advertisement is forwarded to sensors that are k hops away. No advertisement is forwarded more than k hops. Any node that receives the forwarded cluster head advertisement that is not a cluster head joins the cluster of the closest cluster head. A node that does not receive a cluster head advertisement becomes a *forced* cluster head. The algorithm proposed in this paper such simple. An important part of this algorithm is to determine the parameters p and k. For further details of determining those parameters with different number of sensor network please see [36]. The second part of this algorithm proposes a hierarchy of clusters. There exist level 1, level 2, level h clusters. Each node senses data and sends collected data to level – 1 cluster heads and level – 1 cluster head processes and forwards data collected from level – 1 cluster heads to level– 2 cluster heads. The algorithm continues in that way. Election of higher level cluster heads is not much different from election of level – 1 cluster heads. Each level – 1 cluster head elects itself as a level – 2 cluster head with certain probability p_2 and advertises itself. Any level – 1 cluster head that receives the cluster head advertisements joins the closest level – 2 cluster head. For the formation of level – n cluster heads different probabilities are determined. This algorithm focuses on formation of clusters based on minimization of the energy used to communicate information from all nodes to the processing center, it does not propose a mechanism for dynamic change of cluster heads instead the algorithm is run periodically to

give a change every node to be a cluster head. Hence load balancing between sensor nodes is tried to be achieved. Energy savings increase while the level of clusters increases. This algorithm is one of the earliest algorithms and proposes a novel approach to energy based distributed clustering.

Basagni proposed a distributed clustering algorithm for ad hoc networks in [37]. The proposed algorithm (DCA) is well suited for *quasi-static* ad hoc networks. Since in many sources sensor networks are defined as static, DCA is suitable for sensor networks also. DCA is a weight based algorithm that elects cluster heads according to a weight parameter. Even there is no information provided on determination of weights of nodes, this algorithm can be considered as a generalized approach in clustering of *quasi-static* networks. The algorithm is simple. The node with the greatest weight sends a cluster head message to its neighbors stating that it is a cluster head. A node that receives cluster head messages joins the cluster with the greatest weight. If some cluster heads have the same weight then a node decides to join the cluster of which cluster head with the lowest id (Each node is assumed to be deployed with a unique id). If a node has not received a cluster head message than it announces itself as a cluster head. Lowest node id is used to break ties.

3.3 Localization in sensor networks

Localization in ad hoc networks is an important area of study. Especially localization mechanisms that can be applicable to quasi-static ad hoc networks are generally applicable to sensor networks. Localization techniques may lead reduction in power consumption in multi-hop wireless networks and also it is obvious that localization is an important issue for routing in wireless sensor networks.

In this thesis, prior location knowledge is used to decrease key ring size of each sensor node. Nodes are deployed in groups and location knowledge is already bound to each sensor node. Localization mechanisms try to determine position of each sensor network as accurate as possible. Our proposed schemes do not make use of exact positions of each sensor node but makes use of being a member of a specific group. In other words, each sensor node carries the location knowledge it needs to establish key sharing information. Since localization mechanisms play an important role in sensor networks and

there exists considerable attention in this area an examination of localization schemes is presented.

Well-known localization techniques may not be suitable for sensor networks. For instance GPS [45] is a publicly available service but because of resource constraints of sensor networks it does not seem to be applicable to sensor networks also the location determined by GPS may deviate 10-20 meters which can be larger than the deployment points of two sensor nodes. GPS-less approaches for localization are needed and there have been such algorithms proposed. GPS-less localization techniques are mainly based on locally available information to determine relative positions of nodes in the network. Ad hoc positioning system (APS) is proposed in [42]. APS resembles to GPS in the way that it works. It does not work with satellites but landmarks. In APS landmarks are connected in hop by hop fashion which is different from GPS. A node that have acquired distances to at least 3 landmarks can estimate its position in the plane. One hop neighbors of landmarks can estimate the distance by direct signal strength measurement. Using some propagation method 2 hop neighbors can estimate the distance to the landmark. The propagation techniques that could be used:

- "DV-Hop" propagation method
- "DV-distance" propagation method
- "Euclidean" propagation method

DV-Hop propagation method estimates the distances according to the hop count. Each landmark computes a correction (proximate 1-hop distance) and each node computes its distances to the other landmarks by multiplying the correction by the hop count to that landmark. All those values are plug into the triangulation procedure in [43]. This is the simplest propagation method. The second propagation method is DV-distance. In this model distance between neighboring nodes is measured using radio signal strength and propagated in meters rather than in hops. DV-distance method is sensitive to measurement errors. The last method is Euclidean propagation method. In this method true Euclidean distance to the landmark is propagated. In order to compute the Euclidean distance a node needs at least two neighbors. Euclidean method is the one that is closest to GPS.

GPS-less low-cost outdoor localization for very small devices [44] is another approach proposed for localization in sensor networks. This scheme assumes an idealized radio model and proposes a simple connectivity based localization method. A fixed number of nodes in the network with overlapping regions of coverage serve as reference points and transmit periodic beacon signals. All those reference points form a regular mesh structure. Nodes use a simple connectivity metric to infer proximity to a given subset of these reference points and then localize themselves to the centroid of the selected reference points.

Another algorithm proposed for localization in sensor networks is GPS-free positioning in mobile ad hoc networks [44]. This scheme tries to come up with a network coordinate system for ad hoc networks, since sensor networks are usually assumed as static it is applicable to sensor networks also. Distances between nodes are assumed to be measured with any method such as time arrival. Each node locates itself as the center of its local coordinate system and by triangulisation the angles between the nodes are computed. It is obvious that a node needs at least two other 1-hop neighbors for triangulisation. This algorithm is executed at each node and after this step all the local coordinate systems of each node is rotated and mirrored according to one of the local coordinate system of one node. The algorithm proposes an approach for generating a network wide coordinate system without help of GPS.

Localization schemes that do not use GPS use different types of methods to determine positions of nodes in the network. For instance APS use landmarks that resemble to satellites in GPS. It is not wrong to mention that APS is an adaptation of GPS to ad hoc networks. Another approach is to generate a coordinate system with the help of at least two other sensor nodes and triangulisation. All those techniques require beaconing and also localization techniques make use of some distance measurement methods such as time of arrival method or signal strength method. Distance measurement on different areas may lead to different problems. For instance, obstacles in outdoor implementations of sensor networks may lead to wrong distance measurements. Also the shape of the area is another determiner of the health of those measurements. A localization system that is designed

should consider this realization issues and of course the application of those localization system should tolerate such sort of errors.

3.4 Routing in sensor networks

Routing in sensor networks is another issue to be resolved. There have been various schemes designed for routing in sensor networks, but none of them is designed taking security issues into consideration. Directed diffusion proposed in [31] is a data centric routing algorithm for collecting data from a sensor network. Base stations in the network propagate interest for named data. Nodes able to satisfy the interest disseminate information along the reverse path of interest propagation. Directed diffusion is a simple routing algorithm designed for collecting named data out of the whole sensor network. GEAR (Geographical and energy aware routing) [32] and GPSR (Greedy perimeter stateless routing for wireless sensor networks) [33] are examples to geographic routing protocols designed for sensor networks. GEAR uses energy aware neighbor selection to route a packet towards the target region. GPSR resembles GEAR but they are different in choosing a path to forward a packet. GPSR chooses the next hop according to its distance to the destination. The closest path to the destination is chosen to forward a packet. Such routing algorithm results in uneven distribution of energy between sensor nodes. GEAR tries to solve this problem by weighting the hopes according to remaining energy and distance to the destination. This way energy is more evenly distributed among the nodes in the sensor network. Rumor routing proposed in [35] offers an energy efficient alternative when high cost of flooding cannot be justified. Rumor routing works with events and agents. When a source observes an event it sends an agent on random walk through the network. Agents simply carry information about events and at each node it visits, it informs the node with the information. All agents have a TTL (time to live) field, list of events and list of visited nodes in order not visit the whole network in a cycle. When a base station wants to gather information it sends an agent in the same way to collect information previously disseminated by the previous agents.

In [22] different types of attacks to routing algorithms are inspected and important design considerations are proposed regarding most well-known routing protocols [31], [32], [33], [34], [35]. Various algorithms that are based on public key cryptography are proposed

to secure ad hoc networks. Algorithms that are proposed for ad hoc networks [23], [24], [25], [26], [27], [28], [29], [30] are not suitable for sensor networks, so schemes for securing sensor networks should be based on private key cryptography. In this thesis there is no routing protocol proposed but for whom they work on secure routing in sensor networks should be aware of that previous work.

4 PROPOSED RANDOM KEY PREDISTRIBUTION SCHEMES

The aim of this thesis is to develop random key pre-distribution schemes that are easily deployable, scalable and resilient against node capture attacks. There exists a generalized scheme that can be considered as a basis for all zone based distribution schemes. There are three schemes proposed that are derivations of the generalized scheme. These schemes are analyzed in detail according to scalability, connectivity and resiliency they provide.

4.1 Design considerations of the proposed schemes

Many key distribution schemes for other types of networks are not suitable for sensor networks. The innovation regarding key distribution for sensor networks begins with the basic scheme [9]. The basic scheme led to many other schemes to be developed based on its idea. Background information on the basic scheme is given in Chapter 2. The basic scheme itself assumes the uniform distribution of nodes on the deployment which is unrealistic when the real world deployment techniques are taken into account. In other words, the basic scheme does not seem to be applicable for real world scenarios. When nodes are deployed as stated in the basic scheme the appearance of the sensor nodes on the deployment area is like in Figure 4.1. This deployment sample is extracted from a simple simulation of the basic scheme over a 100×100 deployment area. Since the basic scheme assumes a uniform distribution of nodes onto the deployment region the appearance of the distribution is like the distribution depicted as in Figure 4.1.

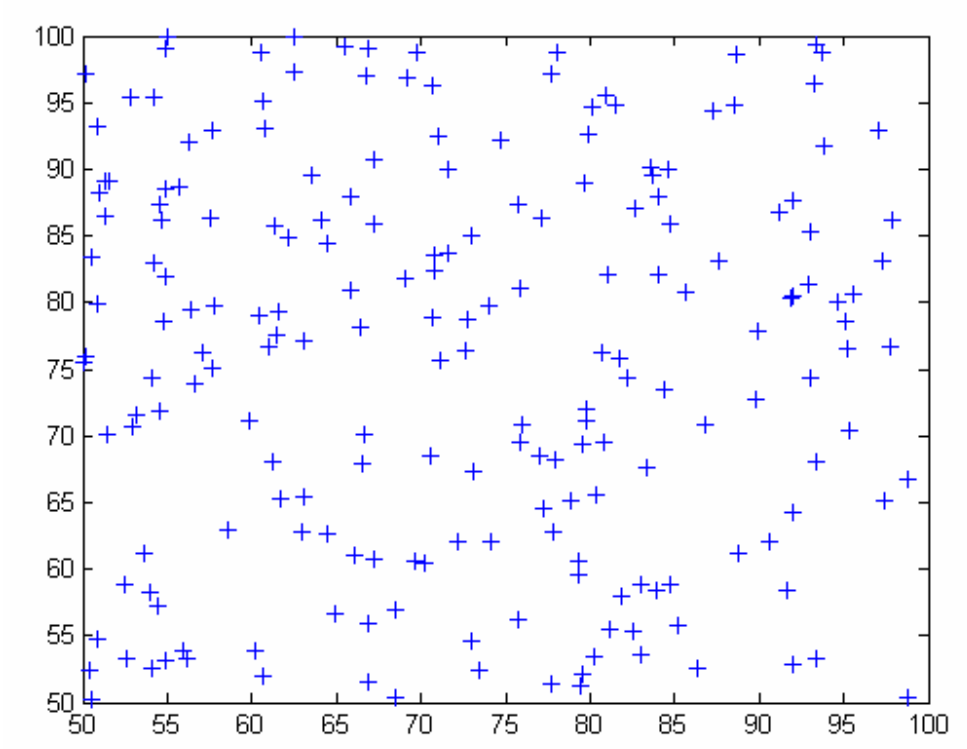

Figure 4.1. Two hundred nodes distributed uniformly random onto a 100x100 deployment area

If the real world deployment techniques are applied such appearance of a sensor network is not possible. So first of all, the technique of the node distribution must be picked carefully that would fit the real world techniques. One of the most applicable node deployment techniques is dropping a batch of nodes onto the deployment area from a moving vehicle, for instance from an airplane. In such a case, nodes are expected to concentrate on the deployment center and from that deployment center nodes can be assumed to be spreading out. The distribution that fits this real world scenario is Gaussian distribution. On the same area, when nodes are distributed normally the final appearance of the sensor network is expected to be like in the Figure 4.2. As it can be easily seen from Figure 4.2, Gaussian distribution is a better way of representing distribution of sensor nodes.

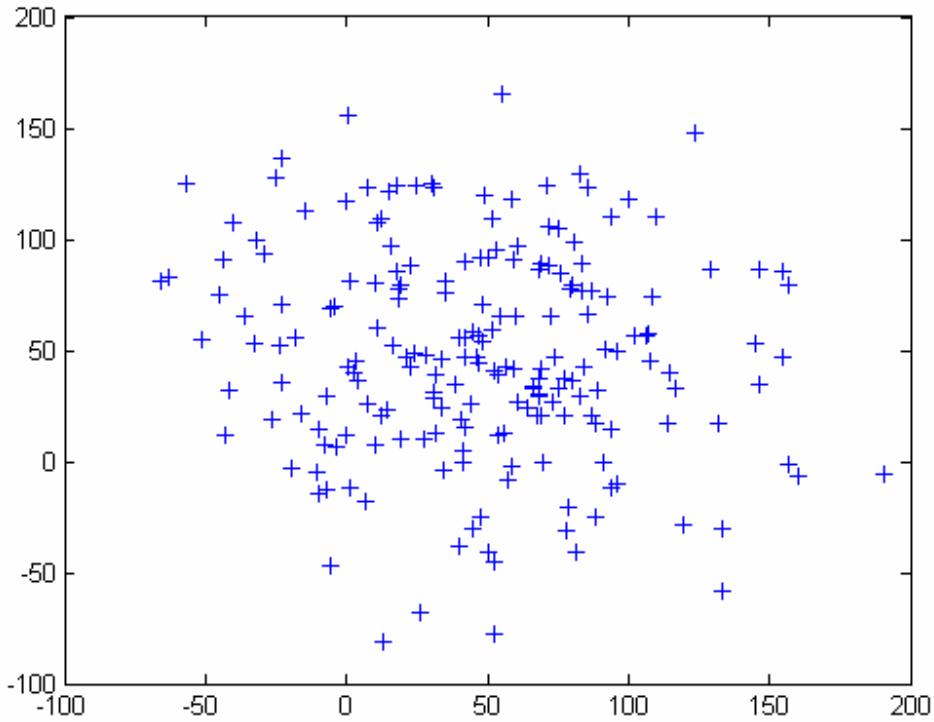

Figure 4.2. Two hundred nodes distributed normally on to a 100x100 deployment area

While designing the key distribution schemes instead of distributing all the nodes on the same area, the deployment area is assumed to be a grid and each deployment takes place in each zone of this grid as if distributing all the nodes on to the same area. This type of grid distribution allows the usage of location knowledge for the keys and disallows distributing some keys that will not be used in some zone. So, keys and zones stick together in order to decrease the number of keys deployed in each node.

4.2 A generalized random key pre-distribution scheme

Many schemes proposed until now assume zone based distribution, but a framework that includes and enhances all those schemes is not present. A generalized scheme that will cover the schemes proposed now in terms of key distribution should be proposed. A general zone based key distribution mechanism is provided in this thesis in order to propose a generalized key distribution mechanism that is scalable and secure. The idea can be reflected as in the Figure 4.3.

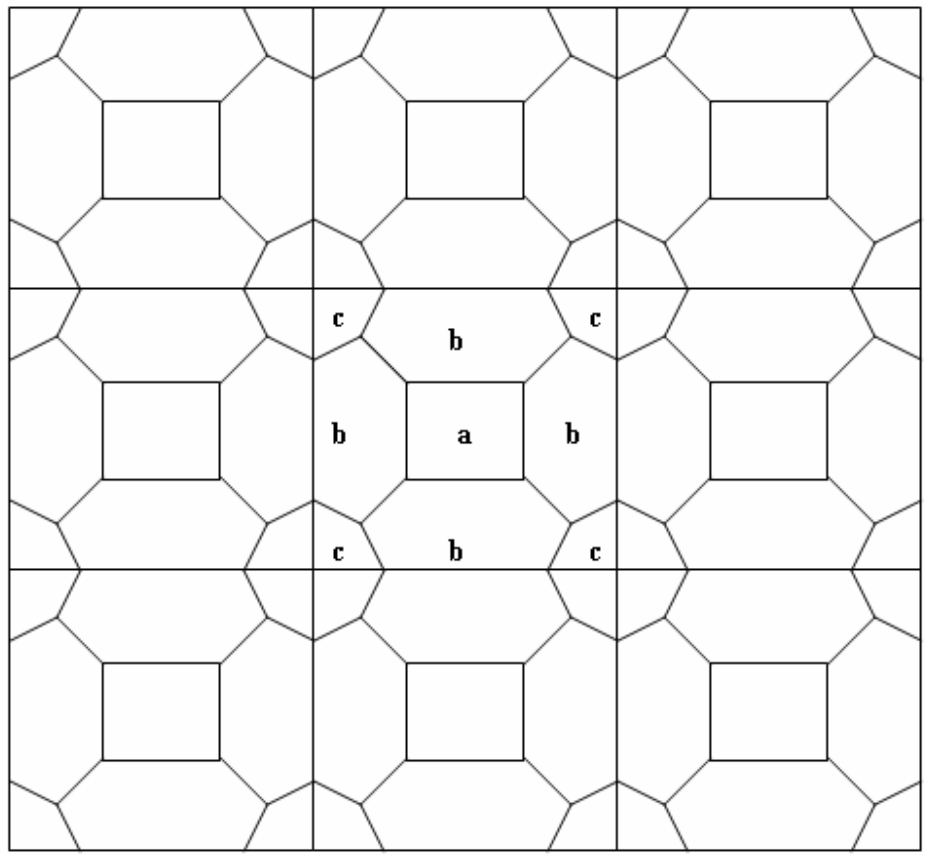

Figure 4.3. Generalized scheme

In this generalized scheme each zone shares some percent of keys with each of its neighbors. For further deployments even there is no neighbor zones to some zone keys are reserved thus further deployments without any security compromise can be achieved.

Each zone shares *b* or *c* percent of keys with its neighbors and no zones that are not neighbors share no keys. Different implementations can be provided according to the distribution and security issues. This scheme can be treated as a framework for zone based key distribution mechanisms. Such distribution mechanism can also be used as it is, even it seems complicated. The motivation is to base key distribution schemes on a scalable framework.

Since each zone consists of different key pools even though there still exists a global key pool, sub key pools for each zone should be stored for the sake of distribution simplicity. Also, in order to extend the network some key pools are to be stored. With the

following approach all of the keys can be distributed and the whole network can be deployed. Following approach is not the only way, but during the implementation of this scheme this technique is applied for the sake of simplicity.

Whole deployment area is divided into a $m \times n$ grid and each zone i, j is assigned a key pool $G_{i,j}$ such that $G_{i,j}$, $i = 1,...,m$ and $j = 1,...,n$ consists of sub key pools $G_{i,j}.NE, G_{i,j}.NW, G_{i,j}.SE, G_{i,j}.SW$, NE stands for northeast, NW stands for northwest, SE stands for southeast and SW for southwest. In the same way, $G_{i,j}.E, G_{i,j}.W, G_{i,j}.N, G_{i,j}.S$ are the other key pools and there exist the central key pool $G_{i,j}.M$, M stands for the central key pool. For a depiction please see Figure 4.4.

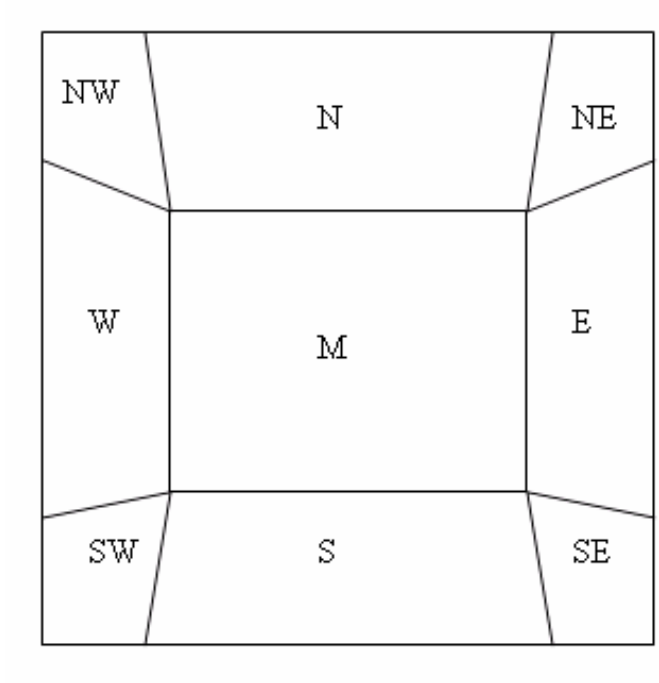

Figure 4.4. Sub key pools in a zone

For all groups $G_{i,j}$ $i = 1,...,m$ and $j = 1,...,n$,

1) If $j - 1 = 0$ then select $b|S|$ keys from the global key pool, assign them to $G_{i,j}.N$ and remove those keys from the global key pool else $G_{i,j}.N = G_{i,j-1}.S$.

2) If $i-1=0$ then select $b|S|$ keys from the global key pool, assign them to $G_{i,j}.W$ and remove those keys from the global key pool else $G_{i,j}.W = G_{i-1,j}.E$.

3) If $i+1 \le m$ and $j-1 \ge 1$ then $G_{i,j}.NE = G_{i+1,j-1}.SW$ else select $c|S|$ keys from the global key pool, assign them to $G_{i,j}.NE$ and remove those keys from the global key pool.

4) If $i-1 \ge 1$ and $j-1 \ge 1$ then $G_{i,j}.NW = G_{i-1,j-1}.SE$ else select $c|S|$ keys from the global key pool, assign them to $G_{i,j}.NW$ and remove those keys from the global key pool.

5) Select $x|S|$ keys from the global key pool where $x=a$ for $G_{i,j}.M$, $x=b$ for $G_{i,j}.E$ and $G_{i,j}.S$, $x=c$ for $G_{i,j}.SW$ and $G_{i,j}.SE$.

4.3 The first Scheme ABAB

In this scheme, there exist two key pools A and B. These two key pools share a common key pool of s. m (key ring size) keys are picked in a uniformly random fashion from the key pool A or B according to the target deployment zone. After that, nodes collected as batches and deployed on to the center of each target zone. The motivation behind this is to design a simple key distribution scheme that is suitable for most of the sensor node deployment purposes. Actually the idea is to make use of that simple location knowledge while keeping the distribution as simple as possible. The ABAB scheme is depicted for a 2×2 zone in Figure 4.5.

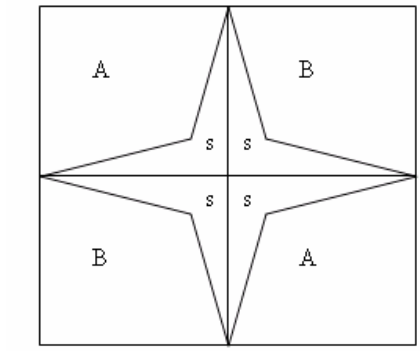

Figure 4.5. Alternating key pool selection of ABAB scheme

This scheme is a derivation of the generalized scheme. In this scheme, the percentage c in the generalized scheme is set to zero. So, there is no key sharing information between key pools that are diagonal neighbors. Only key pools that are horizontal and vertical neighbors share keys. The percentage b is set to $\frac{d}{4}$ where d represents the desired key sharing percentage between zones. In the generalized scheme, in steps 1, 2 and in step 5 for the sub key pools S and E, keys that are selected to be shared are not removed from the key pool and in each of these steps keys that are selected previously are selected again. Keys for the central key pool are selected according to the simple algorithm below.

If $i + j \bmod 2 = 0$ then select the keys from key pool A else select those keys from key pool B.

Zone based distribution puts location knowledge and key sharing information together in order to decrease m. Regardless of the application, grid structure can be considered as a basis for many types of applications and key distribution is an example to this technique. Figure 4.6 depicts a larger application of this approach. Even in the basic scheme, distributing large number of sensor nodes is not an easy task. For instance, distributing 10000 nodes onto a region should be divided into subtasks which make our ABAB scheme applicable. Scattering process should be done part by part. So with a simple extension to scattering process our scheme becomes applicable and since it makes use of location knowledge, it is obvious that it will decrease the number of keys that a sensor node should have. This scheme can be implemented in a few steps as below.

Step1: Generate key pool A.

Step2: Pick $|s|$ keys from key pool A.

Step3: Generate key pool W consisting of $w = |A| - |s|$ keys.

Step 4: Merge key pools W and s such that B is formed.

Step 5: For each zone uniformly select m keys from key pool A or key pool B accordingly and deploy into the nodes.

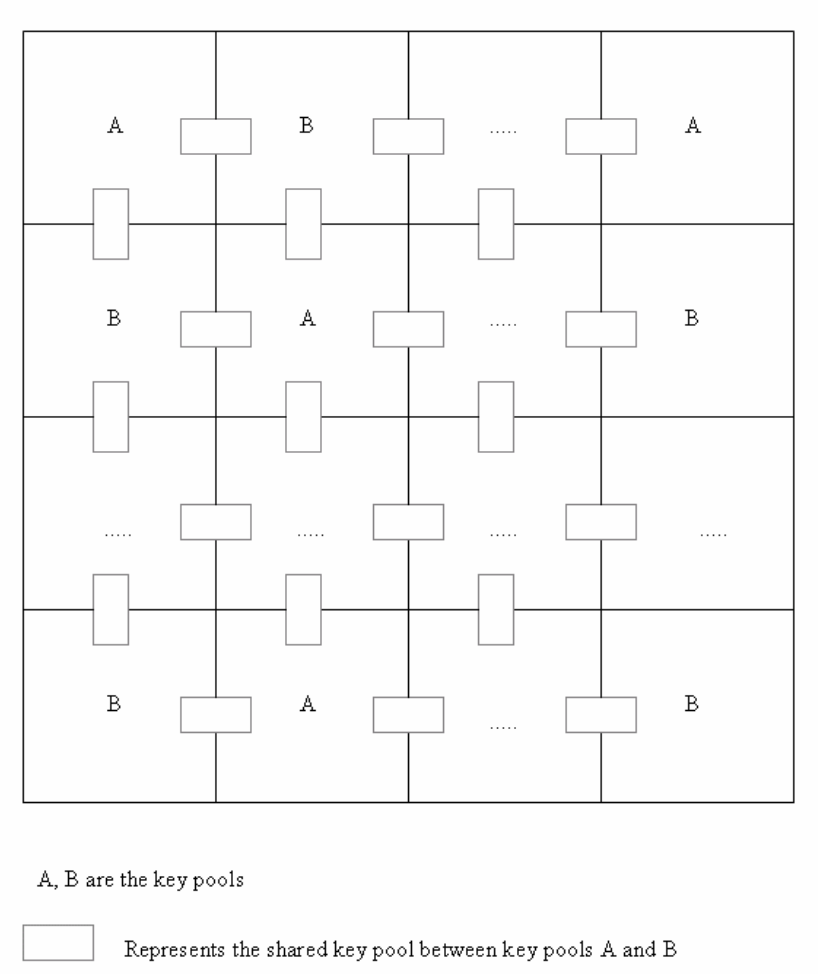

Figure 4.6. ABAB scheme

This scheme can enlarge as new nodes are added to the network in both directions. Adding nodes preloaded with keys from the pools A and B. This scheme gives place to simplicity and easiness of deployment. As the place for simplicity and easiness of

deployment are decreased security provided is increased. Even with this basic scheme a network that is scalable in advance can be deployed. For a depiction of extension to ABAB scheme please see Figure 4.7.

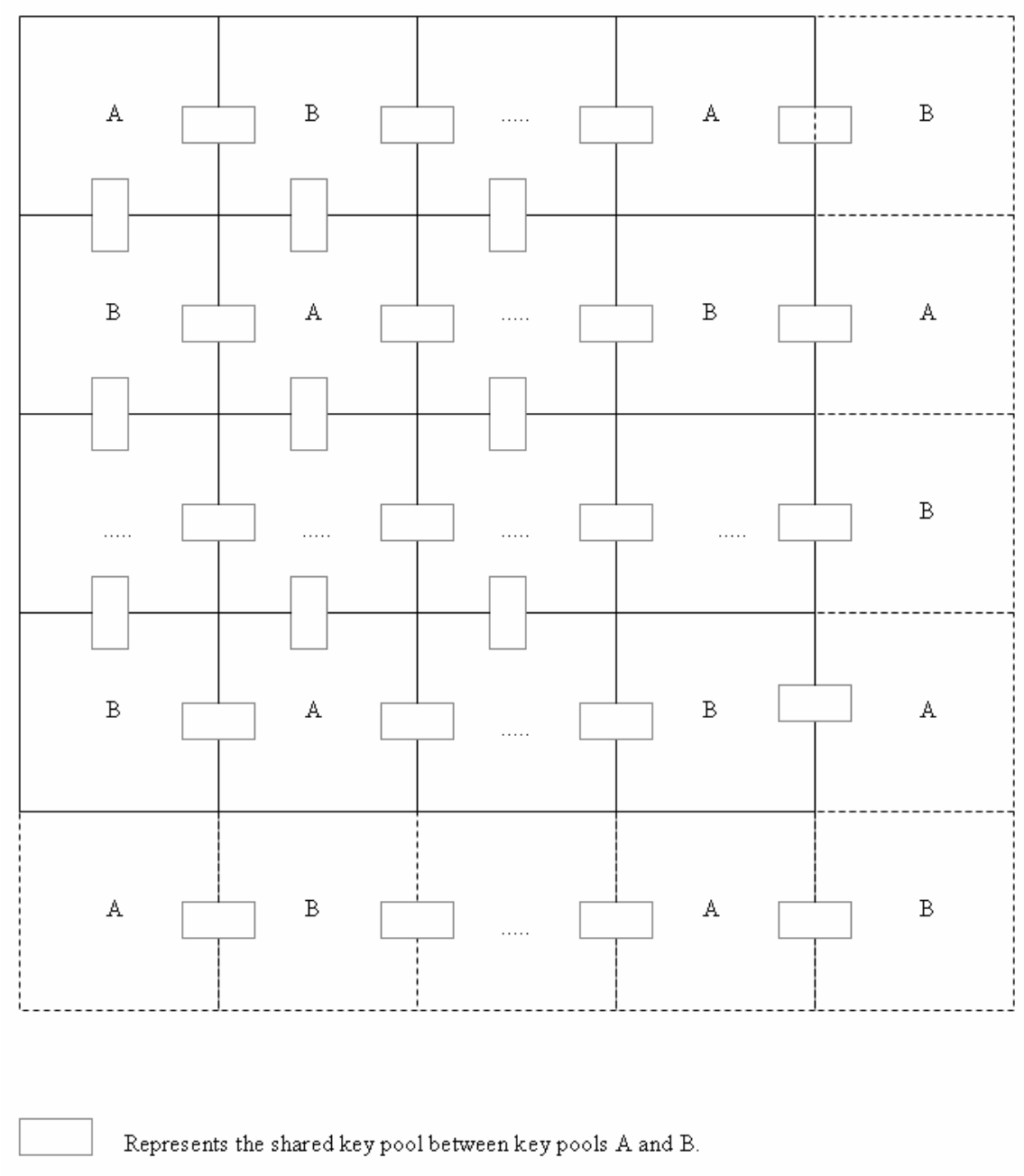

Figure 4.7. Extending ABAB scheme

4.4 The second scheme ABCD

ABAB scheme is easily applicable in sensor networks but it has a resiliency problem since same keys are used in different zones several times. Capture of a node causes compromise of keys that are used in other zones. In order to solve this problem, another scheme named ABCD scheme is proposed.

Decreasing the number of keys in each zone thus increasing the local connectivity is a way of increasing the security. While implementing such a solution distribution of keys must be as simple as possible in order to keep the scheme applicable. For such a scenario, a new scheme called ABCD scheme, is proposed as shown in Figure 4.8. In ABCD scheme two different key pools are generated for each line of deployment. These two key pools share some number of keys with its neighbors both vertically and horizontally. For instance, assume that key pools A and B are generated for the first line of deployment. Pool A and B share some number of keys. Key pools C and D that are generated for the second line of deployment share the same number of keys but key pool C shares the same amount of keys with key pool A and key pool D shares the same amount of keys with key pool B. After generation of all key pools are deployed in alternating manner as depicted in Figure 4.8.

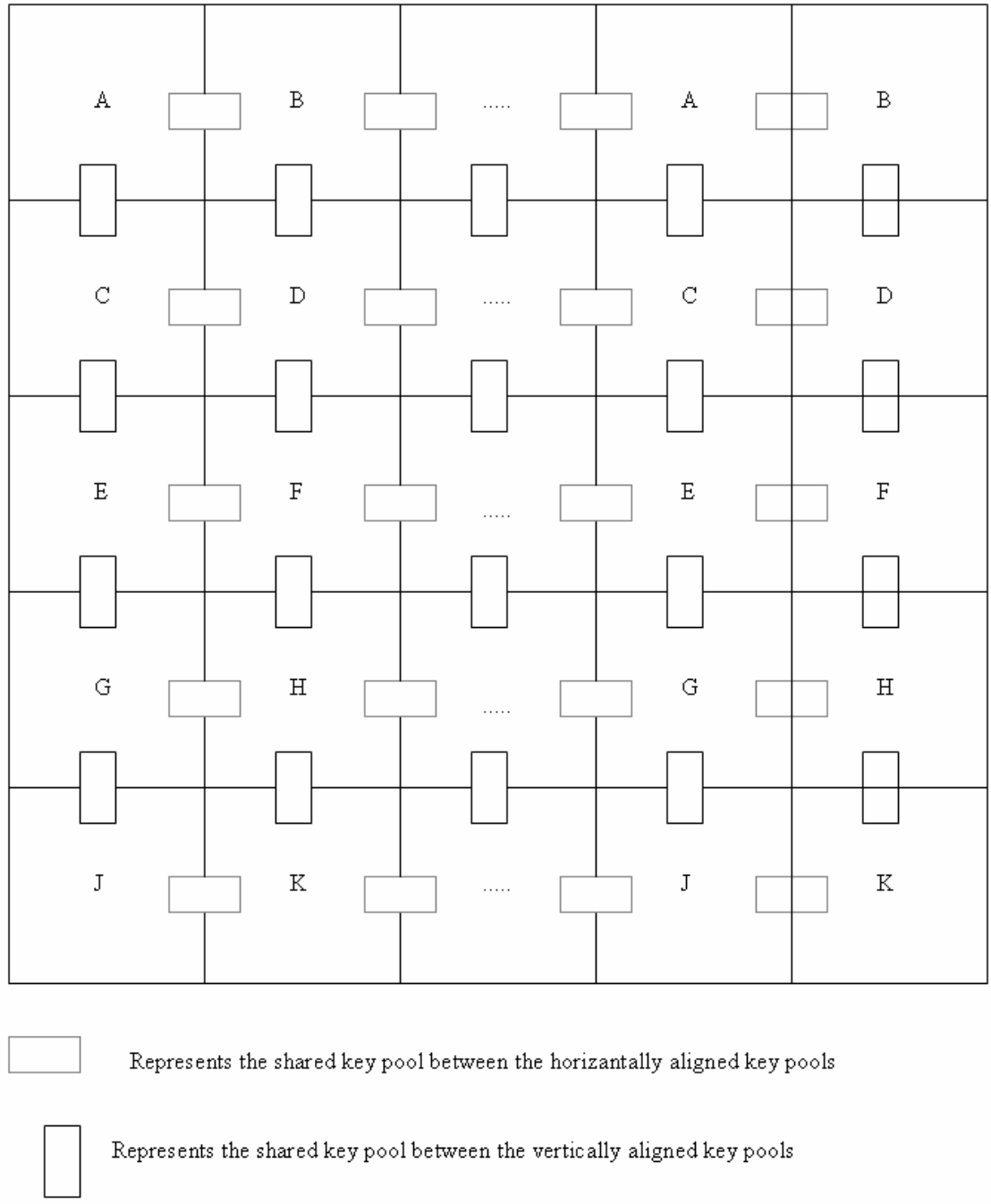

Figure 4.8. ABCD scheme

ABCD scheme is also a derivation of the generalized scheme and can be expressed in terms of it. Key sharing percentage c is set to zero which implies that diagonally neighboring key pools share no keys. In the generalized scheme, in steps 1, 2 and in step 5 for sub key pools E and S, keys that are selected to be shared between key pools are not

removed from the global key pool and for each line of key pool generation, in other words for each i they are stored and used again. For each line of key pool generation, key pools are assigned to each zone in the same alternating manner as in ABAB scheme.

The new scheme conveys the idea of the first scheme but it aims to come up with a more scalable and resilient scheme making a concession of simplicity. Even, ABCD scheme seems more complicated actually it is much simpler than previously designed schemes. Even with little previous deployment knowledge and assumption ABCD scheme provides considerable connectivity and resiliency comparable to previously designed schemes. During all those discussions, the affect of simple deployment knowledge on the distribution schemes is inspected.

There is a tradeoff in this scheme. The tradeoff here is deployment simplicity versus local connectivity and resiliency. Increasing the key pool size for a zone makes the deployment simple but there exists a simple problem that is the security. The simple scheme that assumes the simplest deployment knowledge, actually no prior deployment knowledge, is the basic scheme. Since the basic scheme assumes no prior knowledge of deployment, a large key pool must have to be used and the key ring size of each node must be deployed with more number of keys in order to keep the network connected. For 33 % connectivity with a global key pool (S) of 100000 keys each node should be deployed with 200 keys that causes the compromise of 200 keys as a result of capturing one sensor node. So the main idea should be to decrease the number of keys that should be deployed in each node. Prior deployment knowledge of sensor nodes allows the designer of the scheme to decrease the number keys to be deployed in each sensor node. Since the number of keys that can be deployable into a sensor node in real world applications is limited and also resiliency is increased in that way, decreasing the key ring size of sensor nodes should be the main target of key distribution techniques. In this scheme alternating distribution of keys for each zone on the horizontal line is a simple task; also arrangement of key pools vertically is a simple task. Key distribution process can be described as follows.

Key pools that are to be deployed for each zone are described as $G_{i,j}$ where i, j are the coordinates of each zone. v is the number of keys that is to be deployed in each zone

and w is the number of keys that are shared between key pools both vertically and horizontally.

1) For group $G_{i,1}$ select $v-2w$ keys from the global key pool S then remove those keys from the global key pool and assign those keys to $G_{i,1}$. Also for group $G_{i,2}$ select $v-2w$ keys from the global key pool S then remove those keys from S and assign those keys to $G_{i,2}$. For group $G_{i,1}$ and $G_{i,2}$ select w keys from S then remove those keys from S and assign them to both $G_{i,1}$ and $G_{i,2}$.

2) For groups $G_{i,j}$ where $i=1,2$ and $j=2,...,n$ select $v-2w$ keys from S and for the same groups select w keys from S, assign all selected keys to $G_{i,j}$ and remove them from S.

3) For groups $G_{i,j-1}$ and $G_{i,j}$ where $i=1,2$ and $j=2,...,n$ select w keys from S then remove them from S and assign those keys to both $G_{i,j-1}$ and $G_{i,j}$.

Following the simple procedure above all the key pools for all zones are arranged. After uniformly picking m keys for each node from the corresponding key pool and nodes are ready for deployment.

This scheme can be enlarged horizontally or vertically without any extra arrangement. Adding new nodes deployed keys from the appropriate key pool in the alternating manner in ABCD scheme allows the extension of network. A depiction is given in Figure 4.9.

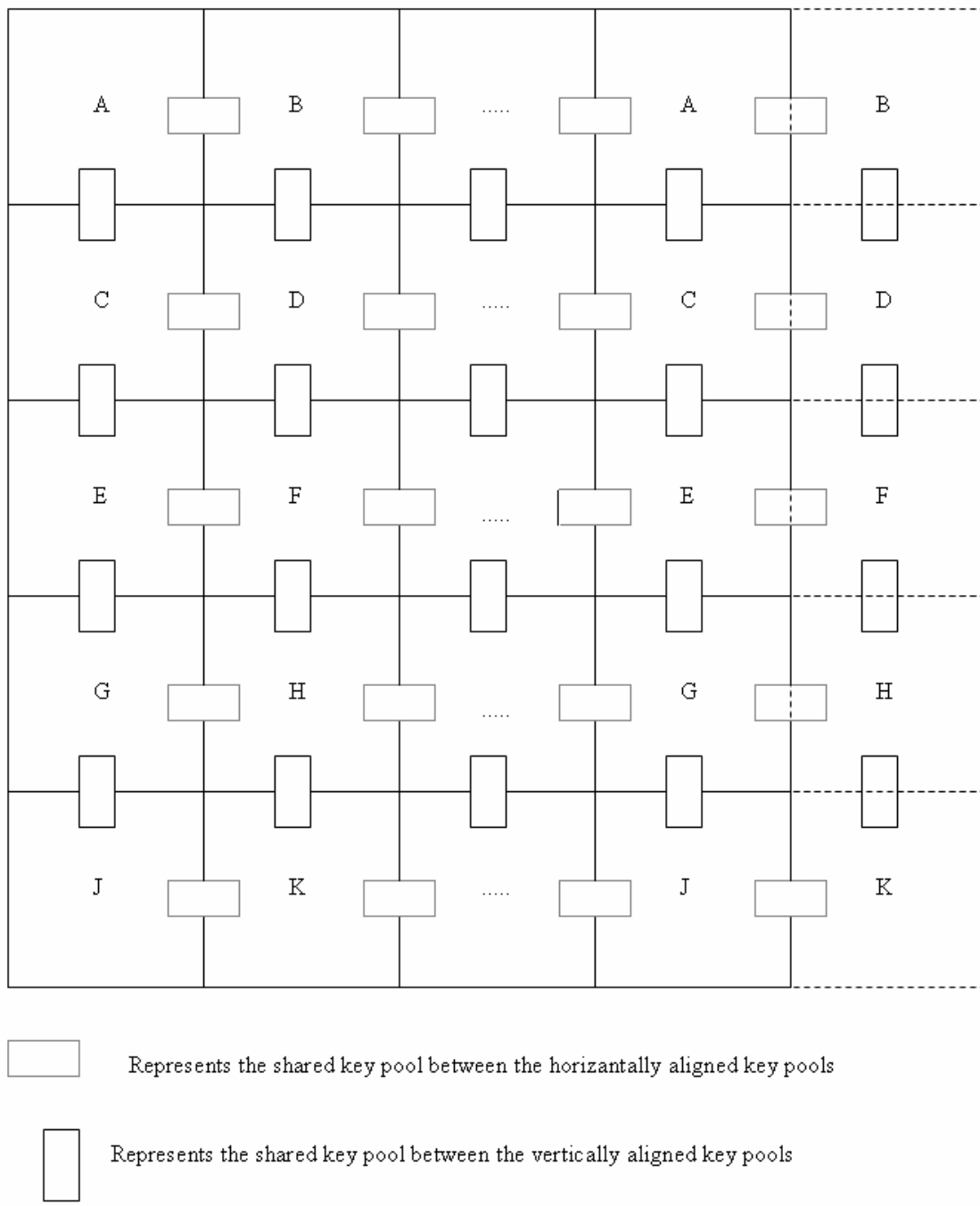

Figure 4.9. Extending ABCD scheme

4.4.1 A modification to ABCD scheme: ABCD-Cyclic

ABCD scheme can enlarge in one direction (horizontally or vertically) without needing generation of new key pools. In order to allow the scheme to be enlarged in both directions, a variant of ABCD scheme, ABCD-Cyclic scheme is proposed. This scheme is an extension to ABCD scheme such that first two key pools used in the first line of deployment share some number of keys with the key pools used in the last line of deployment. Such an approach allows the scheme to enlarge in the other direction. Assume that the deployment area is a $n \times n$ grid. In order to enlarge the network $n+1$th line is deployed again as line 1, the $n+2$th line is deployed as line 2 and line $n+x$ is deployed as line x.

ABCD-Cyclic scheme allows enlargement in both directions without needing generation of new key pools. In Figure 4.10 a depiction of ABCD-Cyclic scheme is presented. Enlarging the scheme in vertical direction is an easy task. Nodes that are to be deployed are loaded with keys as the nodes in the first line of deployment. In this way, generation of new key pools is avoided. In order to further extend the scheme new nodes are deployed as in the second, third, etc. lines. For a depiction please see Figure 4.11.

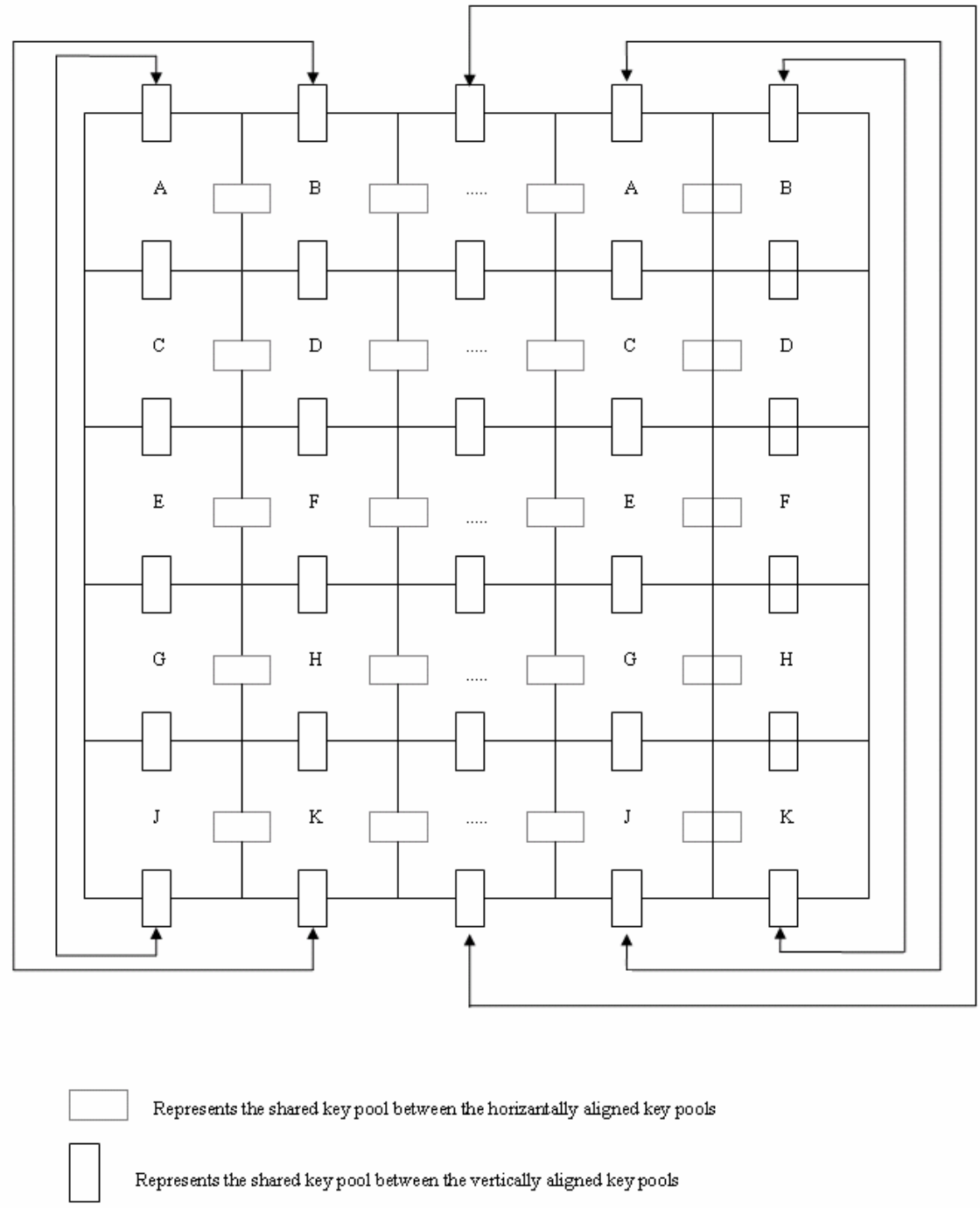

Figure 4.10. ABCD-Cyclic Scheme

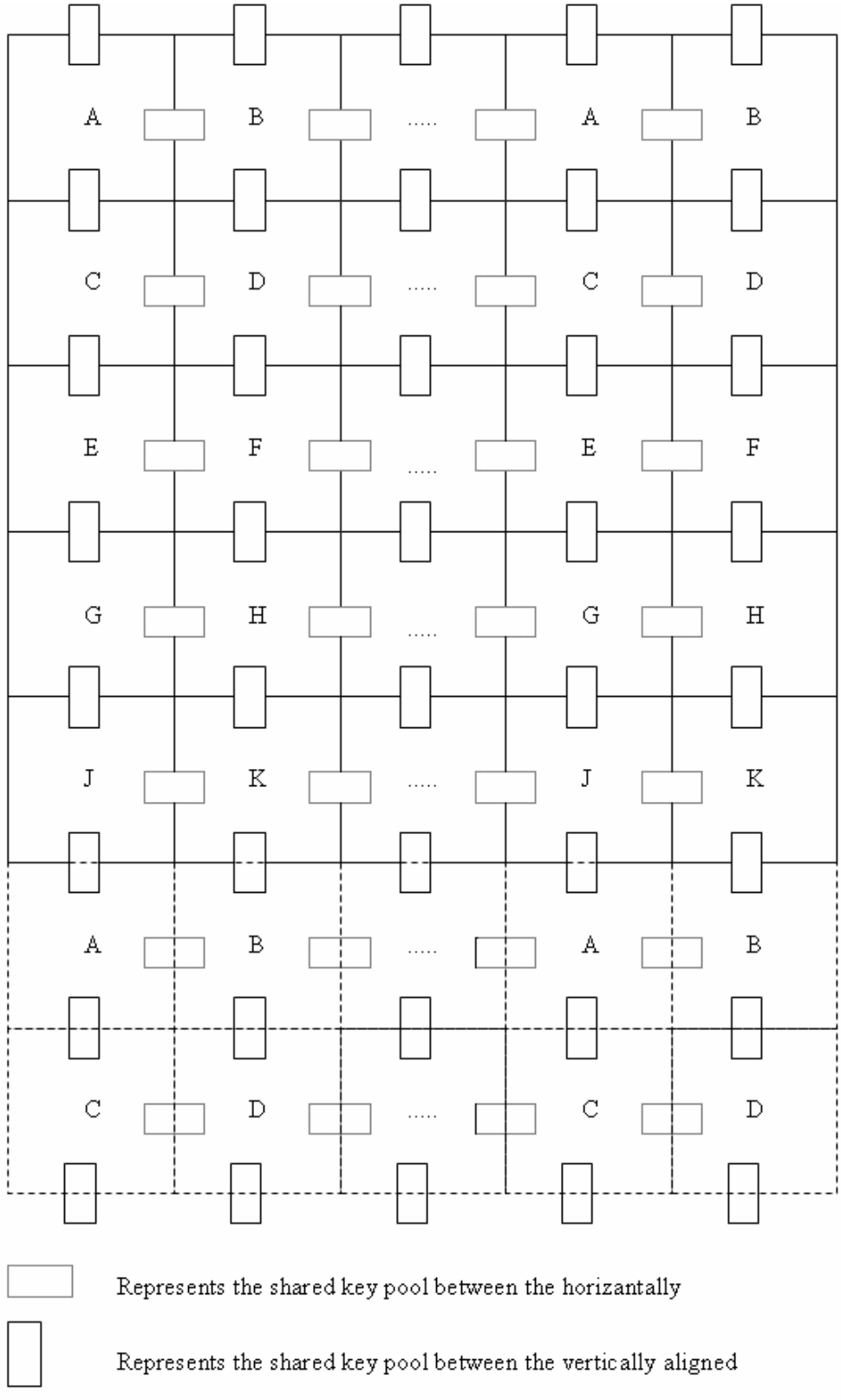

Figure 4.11. Extending ABCD-Cyclic Scheme

5 SIMULATIONS AND TEST RESULTS

In this section a detailed examination and discussion of our schemes are provided, also comparison of our schemes with previously designed ones (basic scheme and scheme by Du et al.) is included. In order to be able to provide a proper examination and comparison of the basic scheme and our schemes, our schemes need to be distributed as in the basic scheme, in other words distribution of nodes in each zone should be uniform to be able to compare it with our schemes. Actually without such an arrangement comparison between our schemes and other schemes with different distribution assumptions is not correct. There is no such arrangement for the scheme proposed in [15] since normal distribution of nodes in each zone is assumed.

All simulations are carried out using Matlab. Simulations mainly focus on the analysis of key relations and connectivity issues. All simulations results are again converted to easily readable format, in other words, they are reflected as graphs again by the use of Matlab. Well-known schemes and basic scheme are all implemented in order to be able to compare them with the designed ones in this thesis. An $1000\,\mathrm{m}\times 1000\,\mathrm{m}$ deployment area is assumed because many schemes designed until now base their simulations and analysis on that deployment area. Also since proposed schemes in this thesis are based on grid based deployment, $1000\,\mathrm{m}\times 1000\,\mathrm{m}$ zone is mapped to a 10×10 grid so, each zone is an $100\,\mathrm{m}\times 100\,\mathrm{m}$ area. Size of the global key pool ($|S|$), with another saying, total number of keys used is 100000. Each node is assumed to have a communication range of 40 meters.

5.1 Definitions

There exist some terms that are frequently used throughout this chapter. Local connectivity, global connectivity and resiliency are the most important terms that must be examined.

Local connectivity is the probability that two neighboring nodes share a key; with the aid of this key they can establish a secure communication link. In simulations this

probability is estimated as below.

For each node N_i, $i = 1,...,n$ where n is the total number of nodes in the sensor network. $p = \frac{\sum_{i=1}^{n} \sum K_i}{\sum_{i=1}^{n} \sum E_i}$ where p is the estimated local connectivity, K_i is the event that node i shares a key with one of its neighbors in its communication range and E_i is the event that node i has a neighbor in its communication range.

Global connectivity can be defined as $\frac{|G_s|}{|N|}$ where G_s refers to the largest isolated component that can securely communicate and N refers to the whole graph such that nodes that cannot communicate with any other node is excluded. Nodes that cannot communicate with any other node are excluded from N because this is caused by Gaussian distribution not by our schemes.

In a sensor network whenever a sensor node is compromised all keys stored in this sensor are also compromised. These compromised keys can be used to secure some other secure communication links in the whole network. The ratio of all compromised communication links to all secure communication links gives the fraction of the communications compromised. Resiliency is the fraction of remaining secure communication links and can be computed by subtracting fraction of communications compromised from 1. In simulations resiliency is computed as follows.

Some nodes are randomly selected and the number of links secured by all those keys in those nodes is divided by the number of all secure links in the network. This test is evaluated for 10 times and mean of the communications compromises are computed. In the end, for a specific number of compromised nodes resiliency is estimated.

5.2 Simulation parameters

The deployment area is a 10×10 grid and each zone in this grid is an $100\,\text{m}\times100\,\text{m}$ area. Total number of keys used (global key pool size) is 100000. For each zone nodes are assumed to be deployed airborne as batches from a moving vehicle such as an airplane. These batches are deployed targeting the center of each zone. However, deployed nodes are diversified from the center of that zone according to Gaussian distribution where $2\alpha = 100\,\text{m}$. Communication range of a sensor node is 40 meters. Please note that Scheme by Du et al [15] also assumes the same deployment parameters. These parameters are selected to be compatible with previously proposed schemes If these parameters are changed in the simulations for some reason this fact will be explicitly stated.

In our schemes key sharing percentage between key pools is an important parameter. In ABAB scheme there exist two large key pools sharing a percent keys. In order to decide on a, the following test is performed.

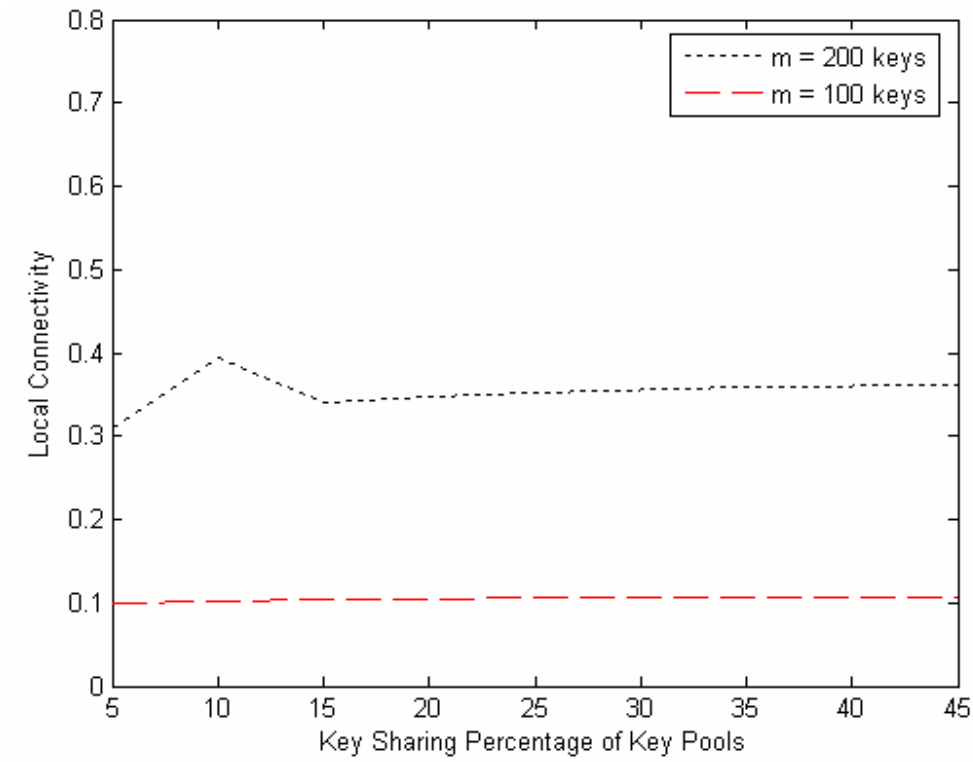

Figure 5.1. Deciding simulation parameters for ABAB scheme

In figure 5.1 ABAB scheme is simulated with different key sharing percentages and with two different key ring sizes. For $a = 10$ and $m = 200$, local connectivity increases and for remaining key sharing percentages local connectivity is almost same. For $m = 100$ case the curve is almost flat which means that for small key ring sizes, the effect of key sharing percentage is not significant. Thus based on these observations to keep the network more connected $a = 10$ is chosen as the key sharing percentage and for all ABAB simulations.

For ABCD scheme, simulations are done to determine key sharing percentage between key pools. Key sharing percentage versus local connectivity is depicted in Figure 5.2.

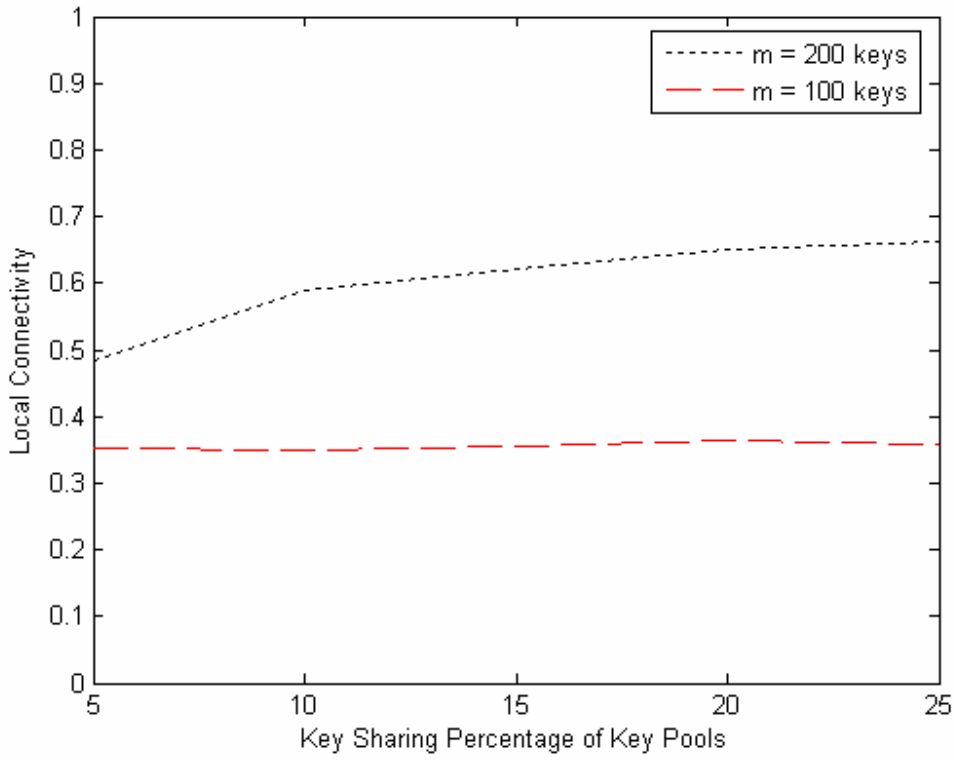

Figure 5.2. Deciding simulation parameters for ABCD scheme

In this case, for $m = 100$ the local connectivity curve is almost flat. For the other case where $m = 200$, the rate of local connectivity starts to reduce when $a = 10$. Increase in local connectivity continues until $a = 25$ but $a = 10$ is selected as the key sharing percentage. This is because, when this scheme is enlarged for some reason since key pools are very small as compared to ABAB scheme, same keys will be used very frequently which decreases the resiliency. Also in order to unify the simulations, key sharing percentage for

ABCD scheme is set to 10 percent.

5.3 Relation between key ring size, connectivity and resiliency

In order to examine the relationships between key ring size connectivity and resiliency, some simulation results are provided. Intuitively, when the number of keys stored in a sensor node increases, local and global connectivity increase. On the other hand, since number of keys stored in each sensor increases if a node is compromised then more keys are compromised. In other words, resiliency against node capture decreases. The most important point is to increase local and global connectivity without increasing the key ring size. Basic scheme does not make use of deployment knowledge. In our proposed schemes, we make use of simple deployment knowledge to increase connectivity without increasing the key ring size.

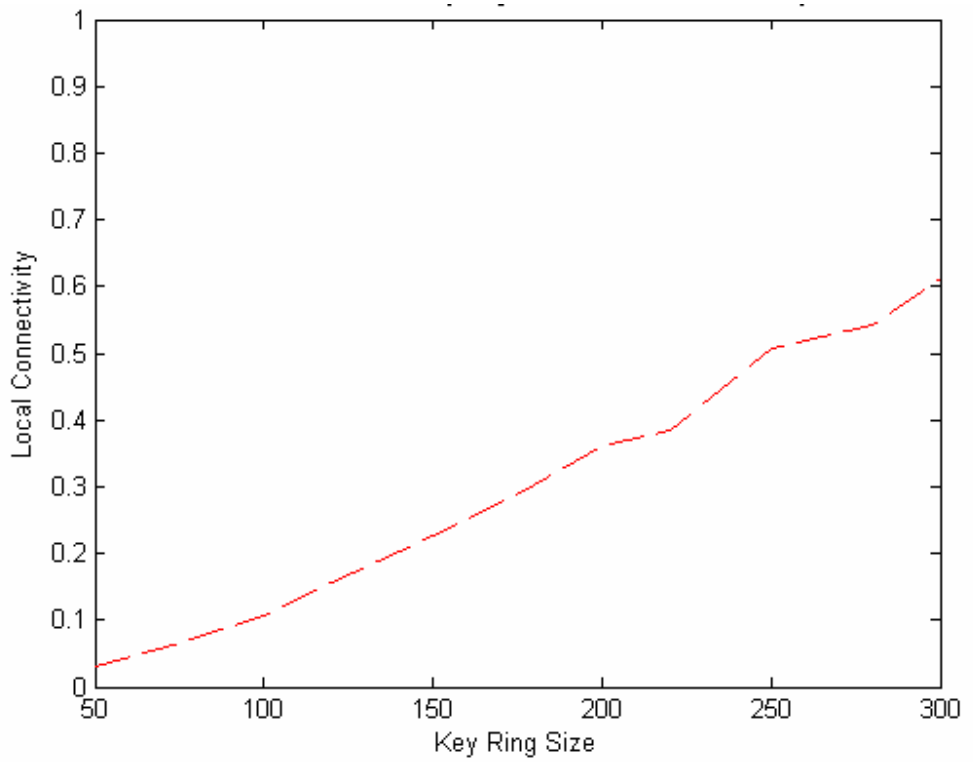

Figure 5.3. Relation between key ring size and local connectivity

In Figure 5.3 a depiction of relation between the key ring size and local connectivity is provided. Local connectivity increases as number of keys deployed in each node

increases. Global connectivity is determined by local connectivity and it also increases as the key ring size increases, as shown in Figure 5.4.

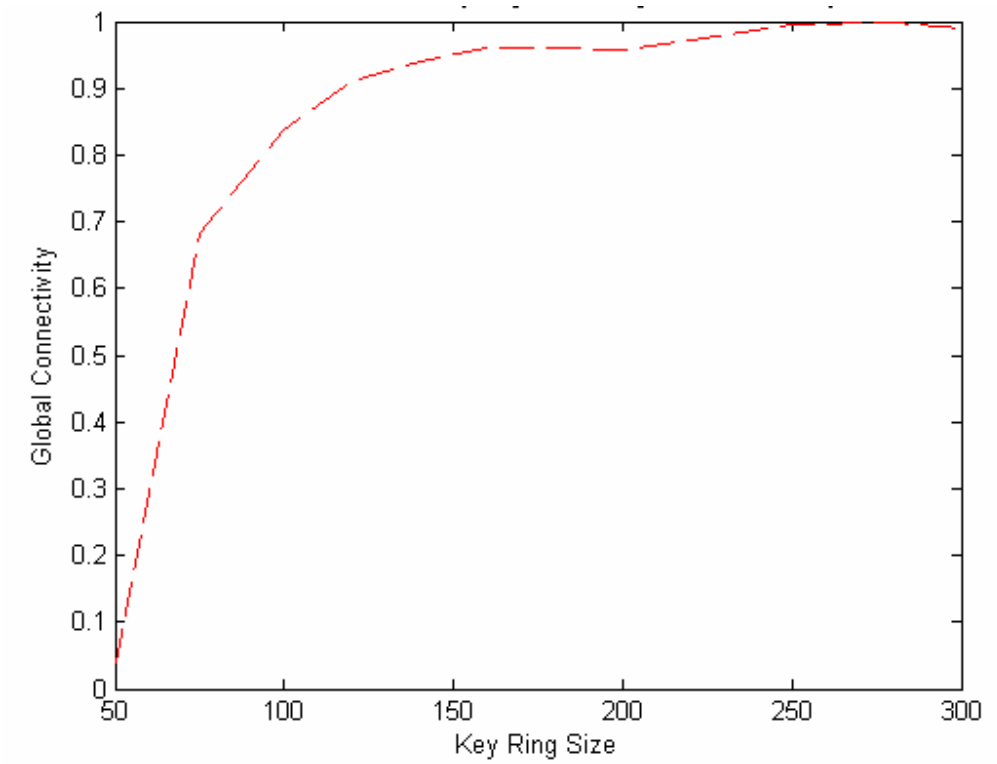

Figure 5.4. Relation between key ring size and global connectivity

Resiliency is also determined by the key ring size. When key ring size increases, local and global connectivity both increase but resiliency decreases since whenever a node is compromised more keys are revealed and more secure communication links are compromised. As mentioned before, the point is to increase local connectivity using less number of keys. In proposed schemes, location knowledge is used to decrease the key ring size to provide reasonable connectivity in that network and the network can be still resilient against node capture attacks. In Figure 5.5, relation between key ring size and resiliency is shown.

Simulations in this section assume a network of 400 nodes deployed on a 4×4 grid. There exist 100 nodes deployed for each zone and variance of those simulations is set to $2\sigma = 100\,\text{m}$. Network size in these simulations is kept small because the motivation is to exemplify the relation between the key ring size, connectivity and resiliency.

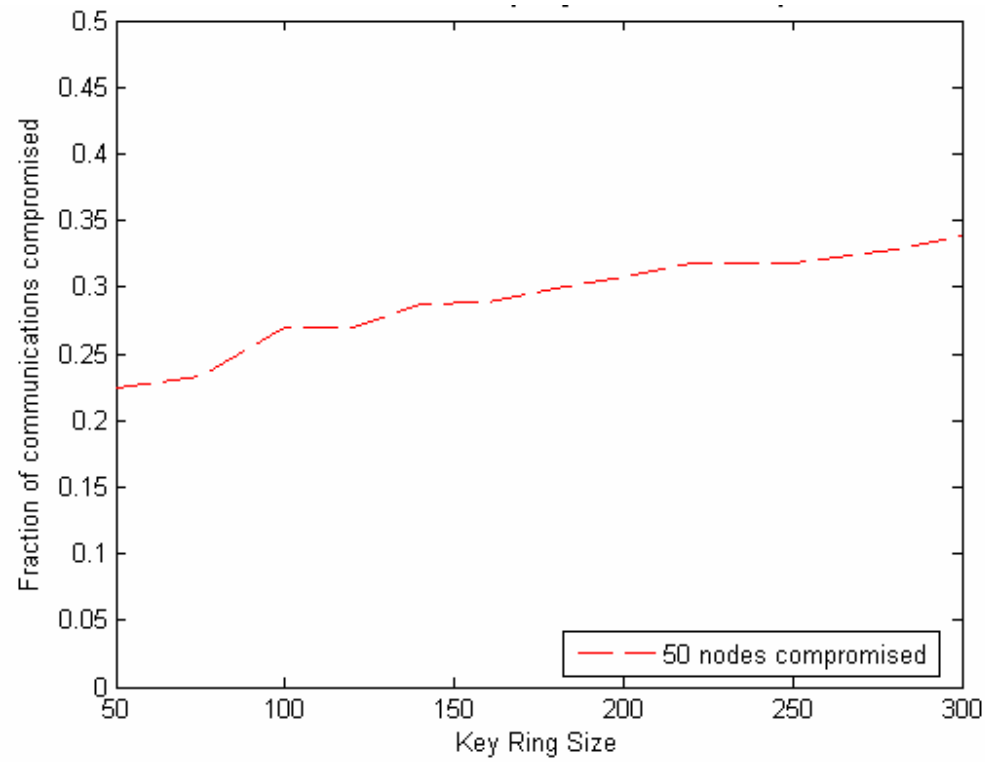

Figure 5.5. Relation between key ring size and resiliency

5.4 Performance evaluation of proposed schemes

Basic scheme is the scheme in which a novel approach to key distribution in sensor networks is proposed. Basic scheme does not assume any prior deployment knowledge. In ABAB scheme a little prior deployment knowledge is assumed and with the aid of little piece of knowledge, local connectivity and thus resiliency against node capture are improved.

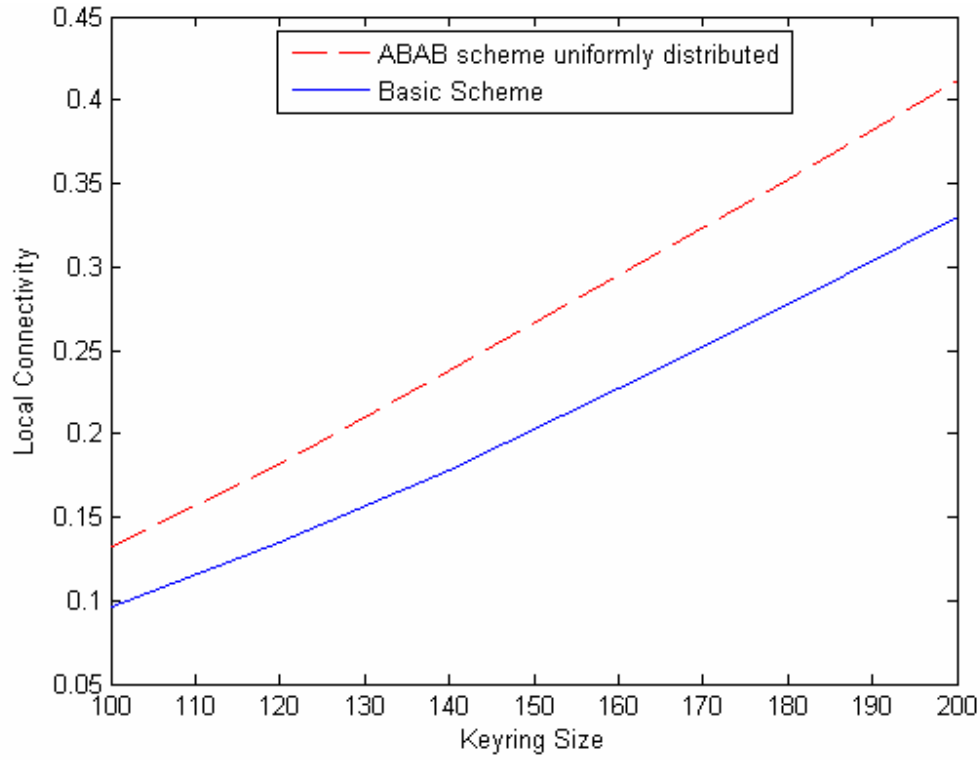

Figure 5.6. Basic scheme and ABAB scheme compared with respect to local connectivity

Local connectivity is a measure that enables comparison of key distribution schemes. With less number of keys deployed in sensor nodes if better connectivity can be achieved then the scheme designed would be more resilient to node captures. In Figure 5.6, a comparison of the basic scheme and ABAB scheme is provided. The main idea is that the basic scheme offers a novel approach to key distribution but its deployment assumption would yield unrealistic results that are satisfactory as compared to scenarios that have assumptions close to the real world. The basic scheme is not suitable for large sensor node

deployments; its application areas can be restricted to small deployments over attended areas. Local connectivity and resiliency are closely related concepts and the aim of the designed schemes should be increasing the local connectivity using less number of keys thus capturing nodes yields compromise of less number of keys and compromise of less number of communication links Since ABAB scheme performs better than the basic scheme in terms of local connectivity; it is obvious that it will also perform better in terms of resiliency according to the close relations between those concepts. Analytic and simulation results from the comparison of the basic scheme and ABAB scheme are depicted in Figure 5.7 and Figure 5.8 respectively.

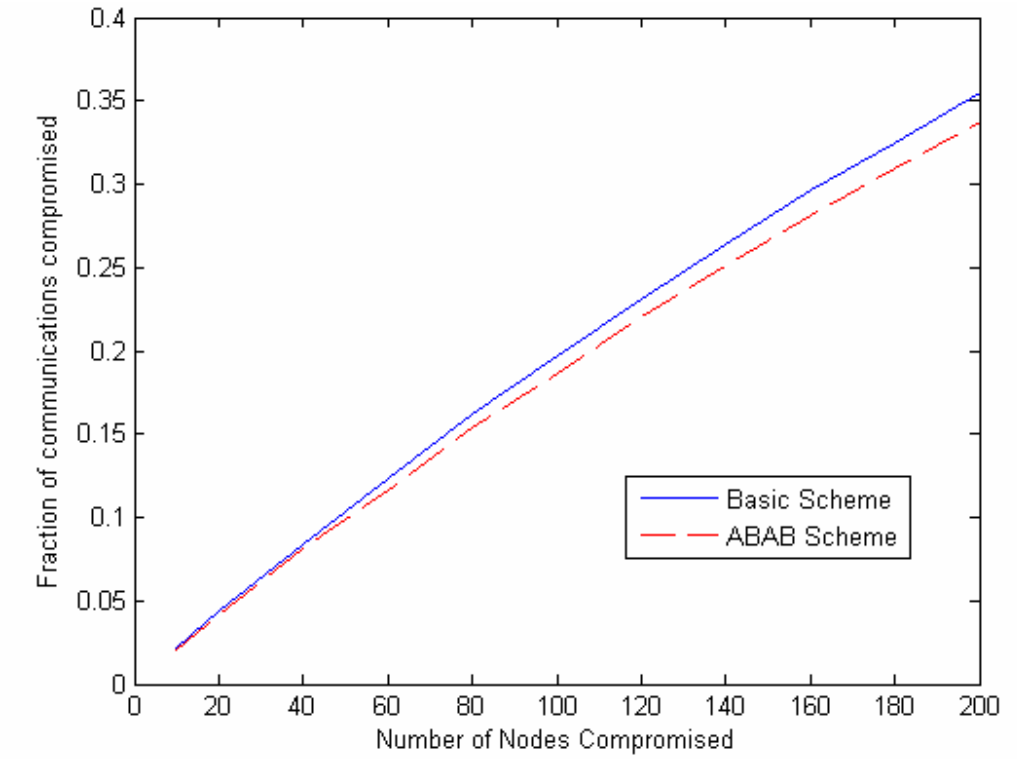

Figure 5.7. Basic scheme and ABAB scheme compared with respect to resiliency by simulation

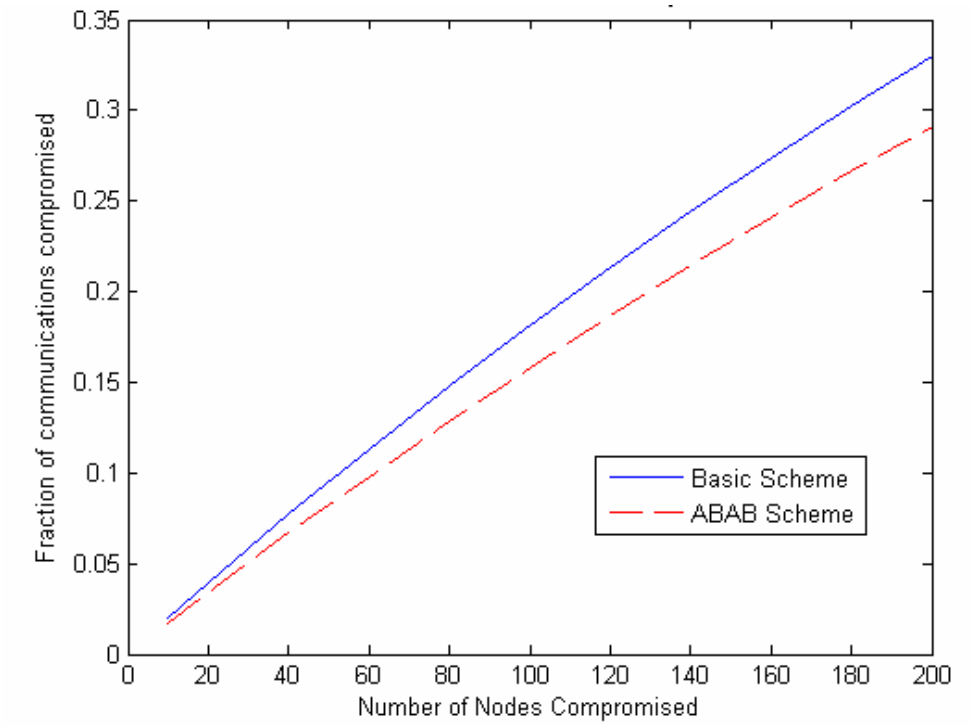

Figure 5.8. Basic scheme and ABAB scheme compared with respect to resiliency analytically

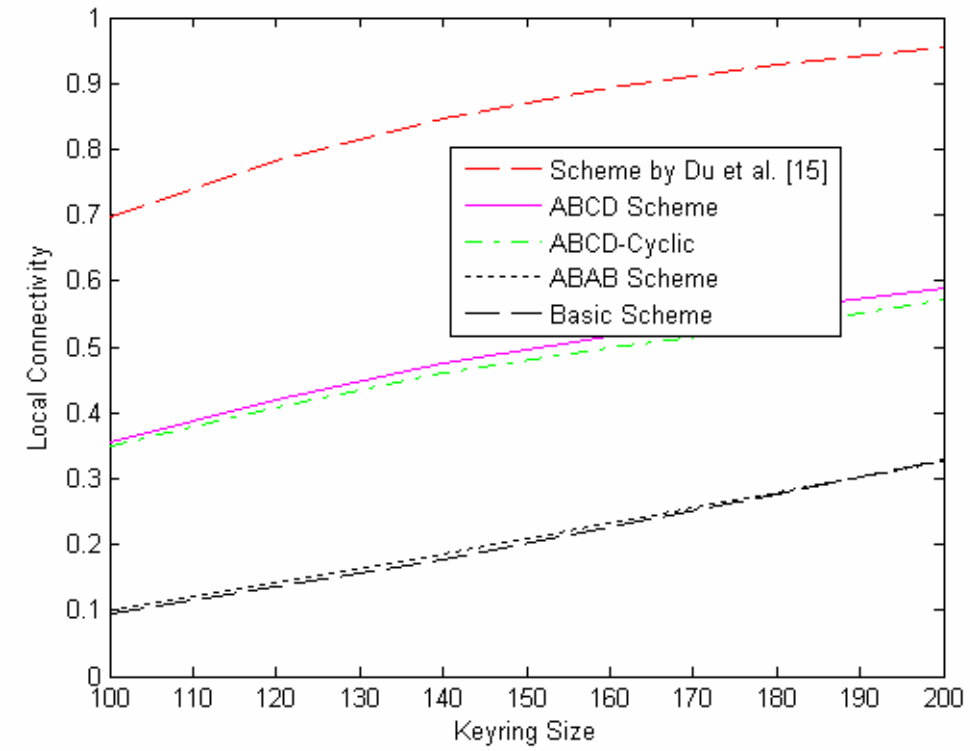

Figure 5.9. Key ring size versus local connectivity for four different schemes

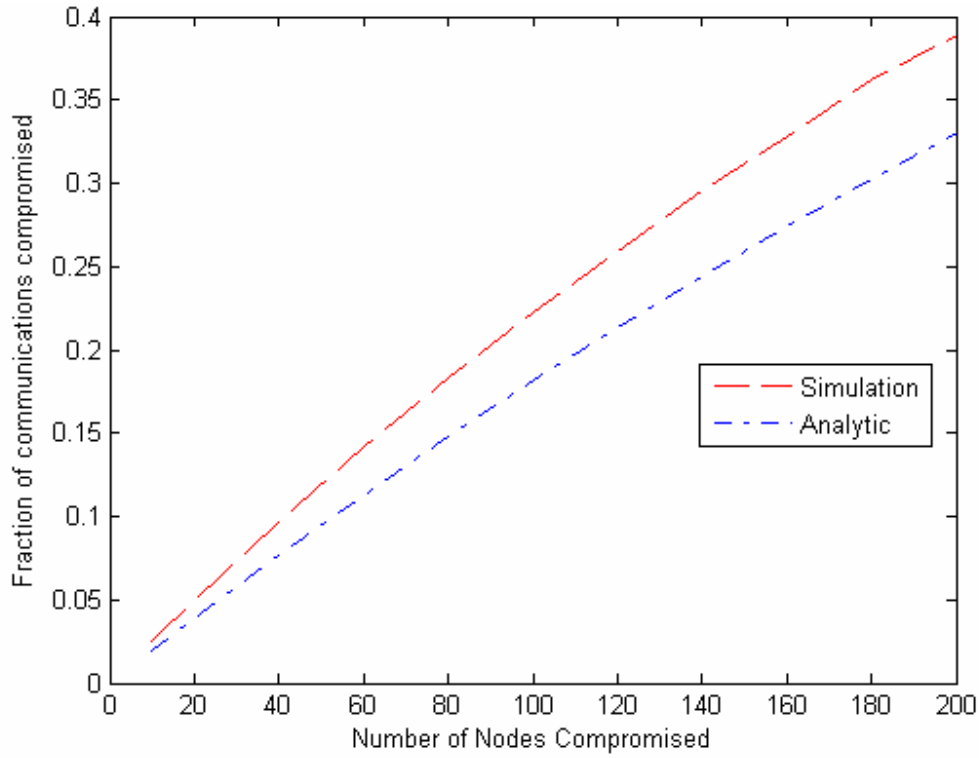

Figure 5.10. Comparison of analytic and simulation results of ABAB scheme with 33% connectivity

In Figure 5.9 local connectivity provided by different schemes is presented. The highest local connectivity with fewer keys is provided by the scheme that is presented in [15] because of the usage of location knowledge. Despite its complication it provides high local connectivity with less number of keys since each zone acts as a little deployment region over the whole deployment region. The main idea behind it can be described as a generalization of whole deployment over a grid deployment region. ABCD scheme performs well as compared to the basic scheme since again it makes use of deployment knowledge even it is much less complicated than the one in [15] (Scheme by Du et al). ABCD-Cyclic scheme performs almost same as the original ABCD scheme. Because ABCD-Cyclic scheme is deployed with the same number of keys with ABCD scheme and the effect of two key pools shared between the first and the last line of deployment is insignificant. Thus ABCD scheme and ABCD-Cyclic schemes perform almost same in terms of local connectivity. ABAB scheme and the basic scheme seem to perform almost the same according to the simulations. Actually this is not the case because comparing the

basic scheme to other three schemes yields results such that the basic scheme performs well because all nodes in the network are deployed uniformly on the deployment area which is not applicable when real deployment scenarios are considered. So even ABAB scheme seems to perform no good than the basic scheme, actually any deployment scheme regarding sensor networks should be considered keeping this in mind and the performance of the designed scheme should be treated according to this practical fact. Comparison of all those schemes both with the assumption of the uniform node deployment of the basic scheme and without that assumption are provided to give a better understanding of the concept. Further explanation and analysis of the performance of ABAB scheme and the basic scheme is provided at the beginning of this chapter.

ABAB scheme is analyzed in two ways, both analytical and simulation. The difference between these two examinations is reflected to the graph in Figure 5.10. Results of the simulations indicate a higher portion of the communications of the network to be compromised as the number of compromised nodes increases because of the distribution of nodes on the deployment area is not uniform (remember that distribution of nodes in each zone is normal). Basically fraction of the compromised keys can be estimated easily as

$$1-(1-\frac{m}{|S|})^{x} \quad . \tag{1}$$

where m is the key ring size, $|S|$ is the size of the global key pool, and x is the number of compromised nodes.

Supplementing the analytical analysis of network compromise including the distribution criteria is beyond the scope. Mainly, all the simulations consist of 10000 numbers of nodes and a global key pool of 100000. Even schemes other than the basic scheme are tested with a global key pool that includes slightly more number of keys, since the global key pool size cannot be set exactly 100000 when those schemes are simulated. The difference between those two analyses is because of taking the node distribution into account. In those simulations local connectivity is 33 percent and required number of keys to achieve that connectivity is 200. When the number of keys in key ring of each node decreases local connectivity and therefore the resiliency against node capture can be improved which is obvious.

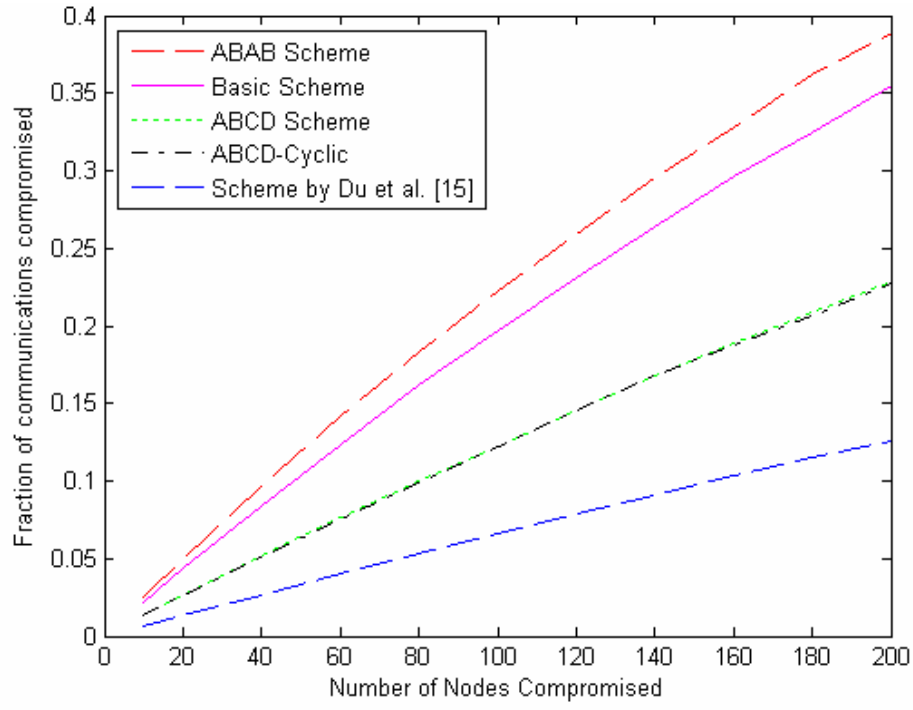

Figure 5.11. All schemes are compared with respect to their resiliency

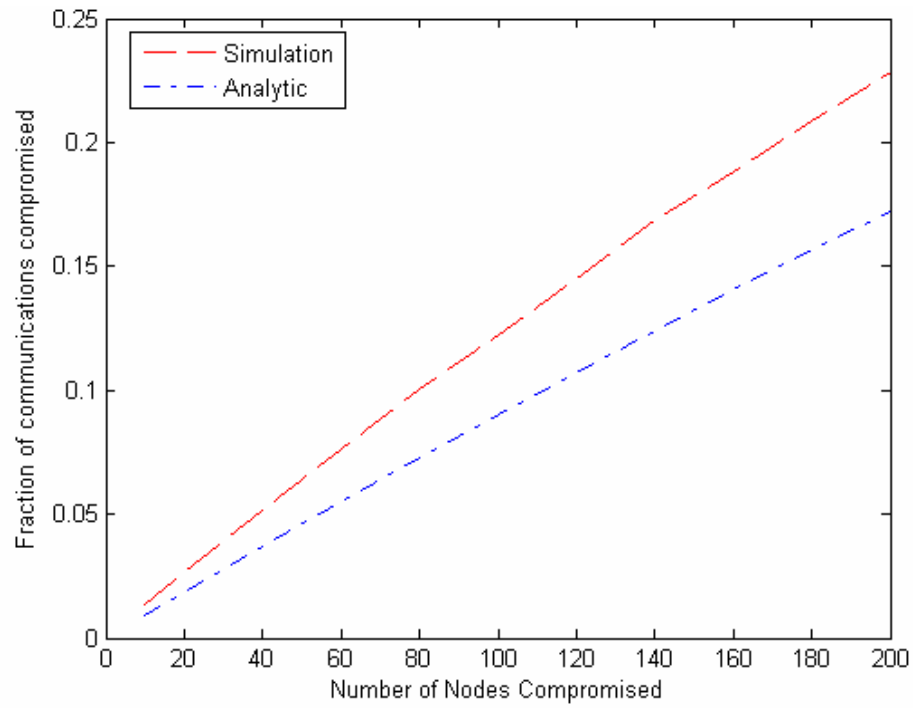

Figure 5.12. Resiliency of ABCD scheme with 33% local connectivity

ABCD scheme provides better resiliency than both the basic scheme and ABAB scheme. A depiction is presented in Figure 5.11. ABCD scheme needs less keys, in other words in order to provide the same local connectivity it needs a smaller key ring size (95 keys for providing 33% local connectivity). ABAB scheme seems to be less resilient against node capture but this point is examined at the beginning of this chapter, but it is obvious that ABCD scheme performs well even it is much less complicated than the scheme proposed by Du et al. in [15]. ABCD and ABCD-Cyclic schemes perform almost same in terms of resiliency. In local connectivity discussion, it is mentioned that they almost provide same local connectivity with the same key ring size. So, it is obvious that they provide almost same resiliency against node capture. According to the simulation results the scheme proposed in [15] offers the best resiliency against the node capture. It is because this scheme decreases the number of keys to be used for each node. In other words, nodes that are deployed on the same zone needs to have much less number of keys to provide the same connectivity as compared to the other schemes. ABCD scheme has also a substantial decrease in the number of keys to be deployed in each node. Even it is not as resilient as the scheme in [15] since it is much applicable when real deployment scenarios are considered, it is obvious that it provides a substantial decrease (as depicted in Figure 5.9) in the number of keys needs to be used and thus a great improvement in local connectivity and resiliency. Analytic and simulation results for ABCD scheme are also provided in Figure 5.12.

Another issue about the distribution is the variance and just applicable to ABAB scheme and the basic scheme. During all the simulations of ABAB scheme and ABCD scheme, $2\sigma = 100\,\text{m}$ is assumed. Deployment points for each batch are $100\,\text{m}$ apart. If σ is decreased than an increase in local connectivity and resiliency is expected. In simulations of which results are depicted in Figure 5.13 and Figure 5.14 $2\sigma = 60\,\text{m}$ is assumed.

When the variance is decreased both the local connectivity and the resiliency against the node capture are improved. This exemplifies the effect of distribution, and depicted in Figure 5.13 and in Figure 5.14.

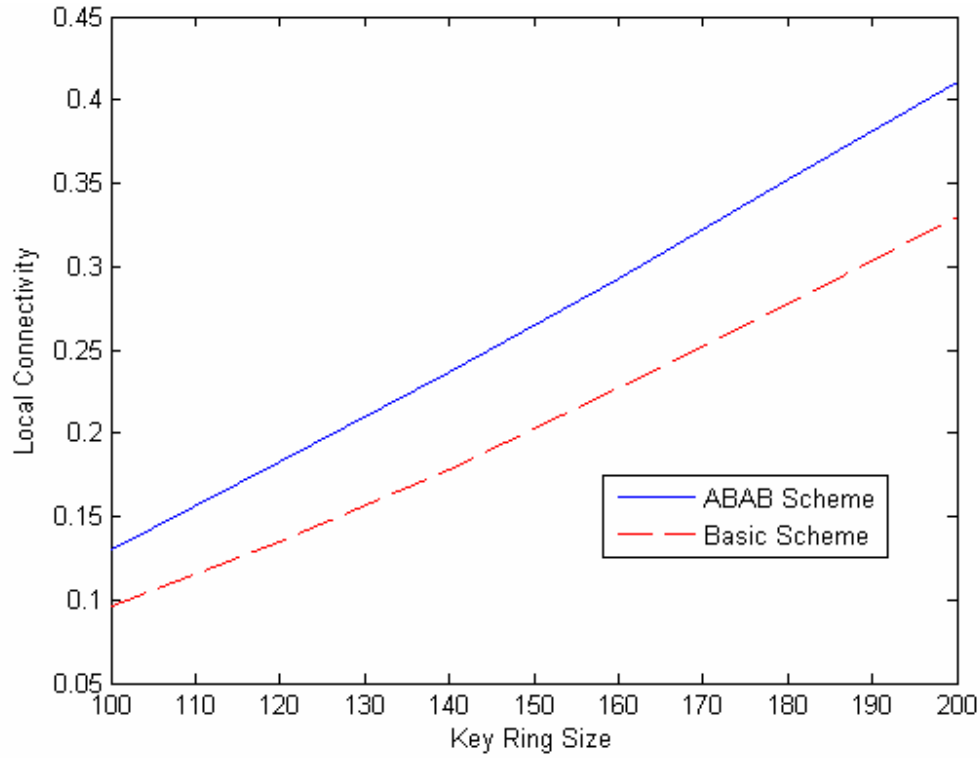

Figure 5.13. Basic scheme and ABAB scheme compared with respect to local connectivity by decreasing variance

When variance is decreased nodes can find neighbor nodes they share a common key because simply the number of neighbors is increased where the other parameters are kept constant. Also without changing the distribution of nodes in each zone, ABAB scheme yields improvement in resiliency just by simply decreasing the variance as depicted in Figure 5.13.

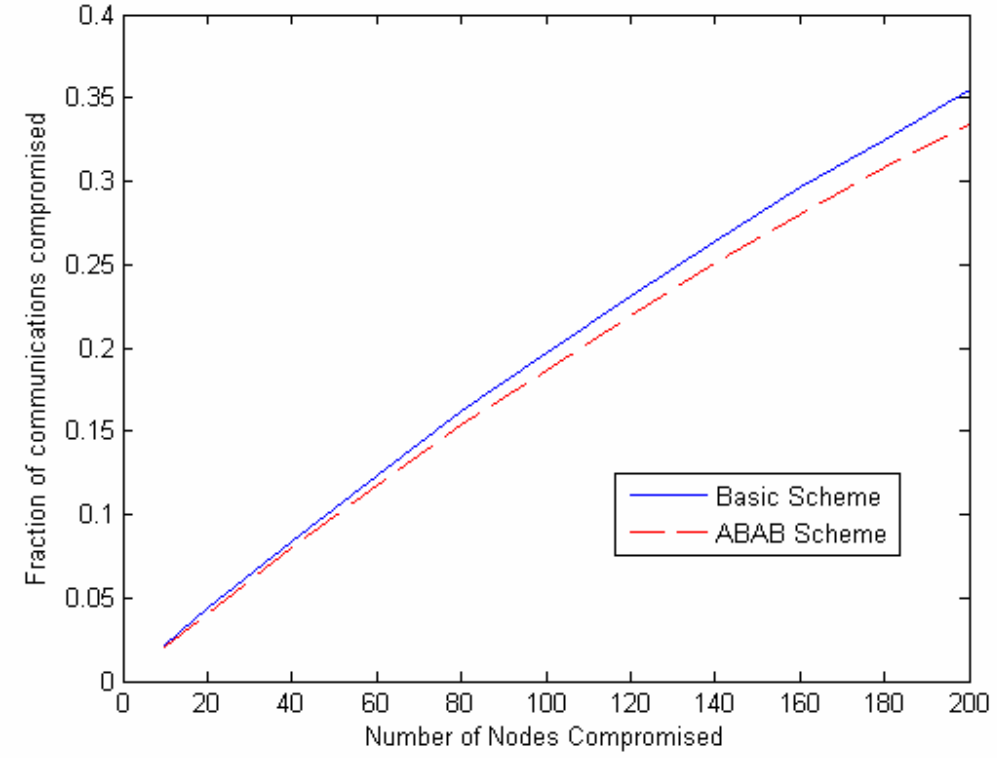

Figure 5.14. Basic scheme and ABAB scheme compared with respect to resiliency by decreasing variance

ABAB scheme is simulated with different variance values where, $2\sigma = 50\,\text{m}$, $2\sigma = 60\,\text{m}$, $2\sigma = 80\,\text{m}$ and $2\sigma = 100\,\text{m}$. For these variance samples ABAB scheme is examined in terms of local connectivity and resiliency. Results are shown in Figure 5.15 and Figure 5.16 respectively. In the cases where variance decreased better local connectivity is achieved with less number of keys. Since better local connectivity is provided with less number of keys it is obvious that resiliency achieved will be better as compared to the case where $2\sigma = 100\,\text{m}$.

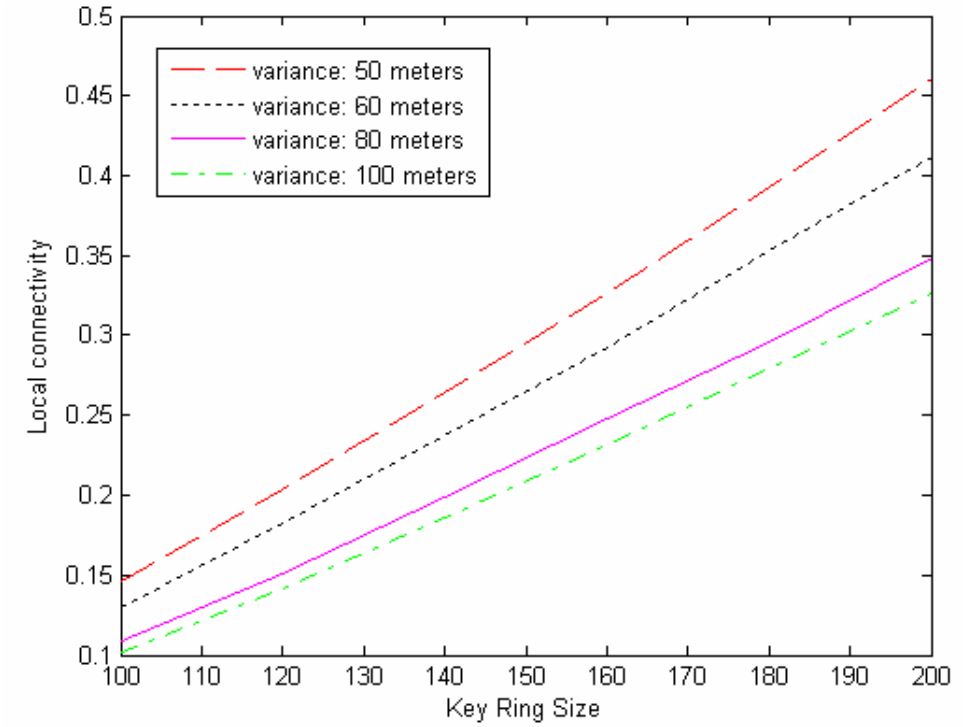

Figure 5.15. Local connectivity examination with different variance samples for ABAB scheme

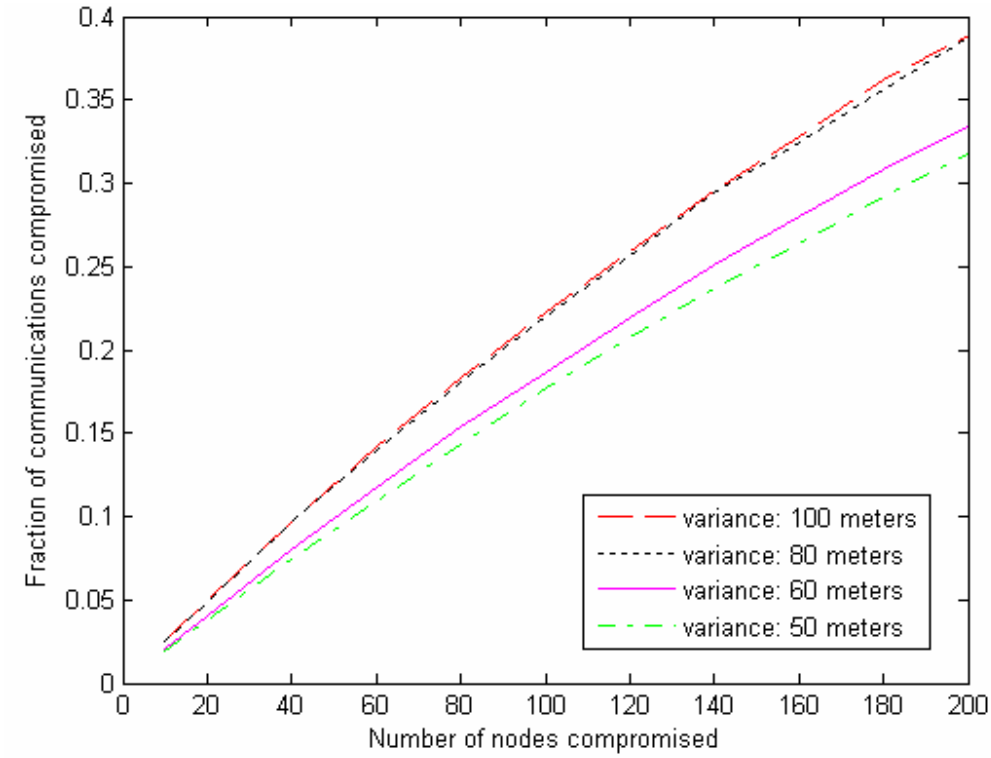

Figure 5.16. Resiliency examinations with different variance samples for ABAB scheme

Global connectivity is another determiner of the key distribution schemes and it is explained in Section 5.1. Scheme by Du et al. provides the highest global connectivity but slightly better than ABCD and ABCD-Cyclic schemes. Actually even the global connectivity that is provided by ABAB scheme with 100 keys may be more than enough when the real world scenarios are considered. A depiction of global connectivity of these schemes is provided in Figure 5.17.

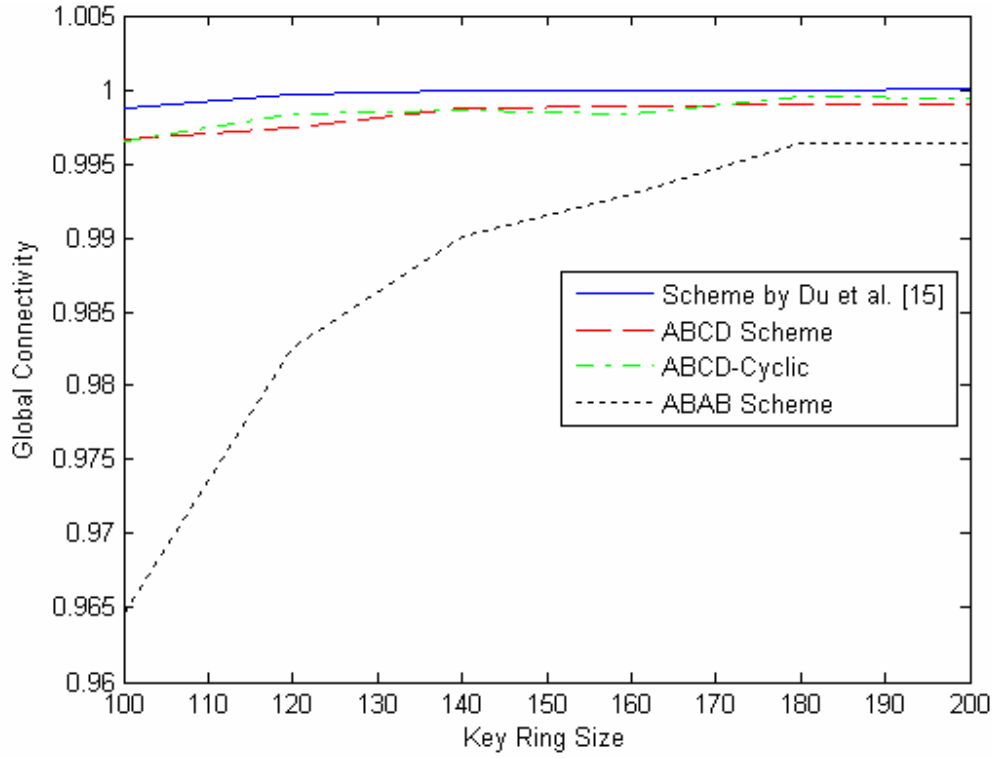

Figure 5.17. Basic scheme and ABAB scheme compared with respect to global connectivity

5.5 Discussions

In this section, further examination of schemes is provided especially comments on inconsistency of simulation and analytic results of the scheme by Du et al. are remarkably important. A detailed inspection of this scheme is provided and some concluding remarks about scheme by Huang [16] are provided.

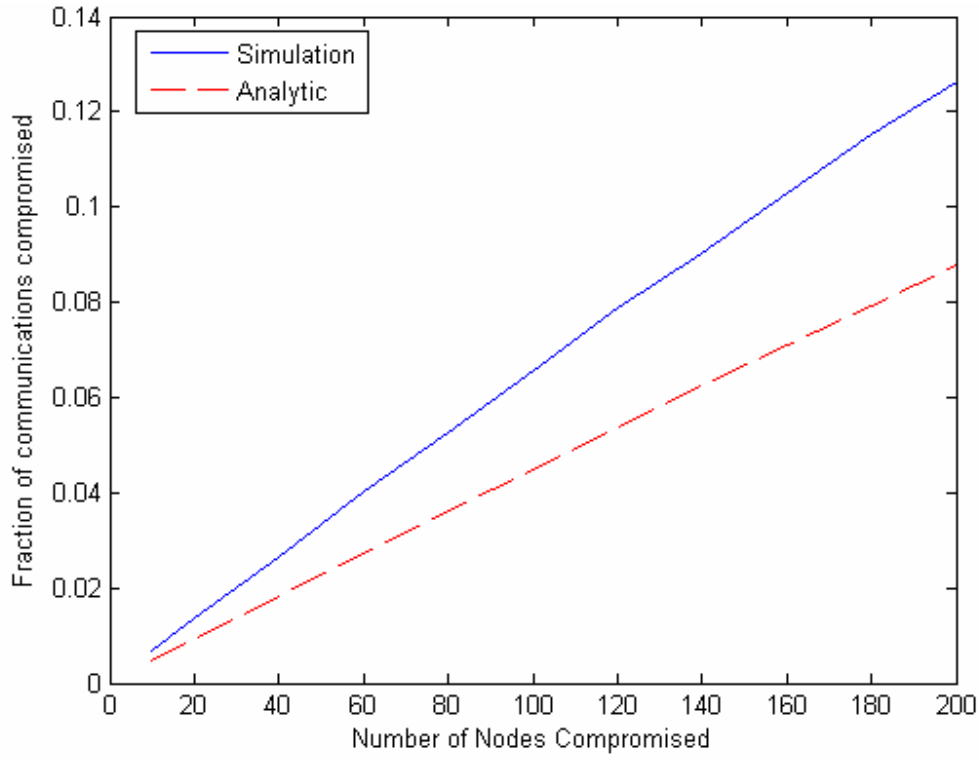

Figure 5.18. Scheme by Du et al. simulation and analytic results compared

The scheme in [15] did not give simulation results for their schemes regarding resiliency. Actually their results are based on the analytic Formula (1). Analytic results for their scheme are more than satisfactory, but actually this is not the case. Formula (1) is used to calculate the probability that a key K is compromised where x nodes are compromised. It does not yield the result that the percentage of secure communications compromised when x nodes compromised. Actually Formula (1) defines an upper bound on resiliency of the schemes. It does not consider the effect of distribution on deployment area. So it should be explicitly mentioned that analytic results are upper bounds on resiliency of the proposed schemes and must be treated in this way. In simulations, percentage of communications compromised can be estimated and that estimation differs from the analytical results. The difference between these two results is not provided by the authors of the paper, but it is implemented in this thesis in order to have a better idea about this scheme. The simulation results indicate that the scheme itself is not that much satisfactory as presented in the original paper. A depiction of these results is presented in Figure 5.18.

Another issue while computing the resiliency of these schemes is that: How should be

the nodes captured? Throughout this thesis all resiliency simulations are based on uniform capture of sensor nodes. Selective capture of nodes includes involving the knowledge of key distribution which is quite possible. Huang offered a scheme [16] that involves selective capture of nodes. Huang mainly focuses on the capturing all the communications of whole network. Also, critics of random key deployment schemes are provided in [16] such that their scheme is secure until a predefined number of nodes have been compromised. In other words, it mentioned that their scheme is perfectly secure against capture of some number of nodes, but after that point whole network is compromised however which is not mentioned. Also that scheme is not scalable in any way, which is not suitable for cases in which the network should enlarge for some reason. Huang has another objection to assumption of disseminating nodes normally onto the deployment zone since data that is to be sensed may be distributed uniformly such that it must be sensed uniformly. So, it is assumed that nodes are deployed uniformly onto each zone. Actually, if this is the case, nodes must be deployed onto on such an area that can be easily accessible all the time, in other words, the deployment area must be attended. But in this thesis the deployment area is assumed to be unattended, so, in order to be realistic such a distribution technique is assumed in this thesis.

6 CONCLUSION

In this thesis, we aimed to develop a random key pre-distribution scheme for sensor networks. For this purpose, we examined some of the previously designed random key pre-distribution schemes. Then we presented a generalized random key pre-distribution scheme for sensor networks. We derived three special cases of the designed scheme called ABAB, ABCD, ABCD-Cyclic and examined them with respect to previously designed and widely accepted schemes. In ABAB scheme there exist two key pools A, B for all sensor deployment field and in ABCD and ABCD-Cyclic scheme for each line of deployment there exist two key pools. The idea is to propose random key pre-distribution schemes that offer simplicity in deployment phase; with respect to such simplicity in deployment phase we try to achieve considerable connectivity and resiliency. We simulated these three schemes and compared their resiliency, global and local connectivity to other widely accepted schemes.

It is evident that, the scheme we proposed can be used to further derive new random key pre-distribution schemes according to security, and simplicity to be provided. The scheme proposed in this thesis can be considered as a base and a generalization of random key pre-distribution schemes proposed until now. The three derivations of the scheme present random key pre-distribution schemes that provide considerable security while keeping key distribution task as much as simple.

Available memory is one of the most important restrictions of sensor nodes. Since the available memory is limited as compared to many other electronic devices, schemes that achieve better connectivity and resiliency with smaller key ring sizes are needed. In order to provide 33% connectivity, our proposed ABAB scheme requires 172 keys while the basic scheme can achieve the same local connectivity with 200 keys. Also our proposed ABCD and ABCD-Cyclic schemes both provide 36% local connectivity with 100 keys where the basic scheme can only provide 1% connectivity, and the scheme by Du et al. [15] provides 70% connectivity. ABCD and ABCD-Cyclic schemes provide 33% local connectivity with only 95 keys where the basic scheme requires 200 keys, and the scheme

by Du et al. requires 46 keys. When 100 nodes are assumed to be compromised fraction of communications compromised in the basic scheme is 20% , in ABCD and ABCD-Cyclic schemes fraction of communications compromised is 12% and in the scheme by Du et al. it is 7% . All these results prove that our schemes provide promising connectivity and resiliency according to their applicability, and deployment simplicity.

The idea presented in this work is based on making a tradeoff between deployment simplicity and security provided while assuring that the security provided is reasonable with respect to that level of simplicity provided in key distribution phase. We do not come up with a unique solution to key distribution problem in sensor networks, but we come up with a simple, applicable and promising solution to key distribution in sensor networks.

7 REFERENCES

[1] I. F. Akyildiz et al., “Wireless sensor networks: a survey”, Computer Networks, Vol. 38, pp. 393-422, March 2002.

[2] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, “A survey on sensor networks”, IEEE Communications Magazine, vol. 40, no. 8, pp. 102–114, August 2002.

[3] G. Hoblos, M. Staroswiecki, A. Aitouche, “Optimal design of fault tolerant sensor networks”, IEEE International Conference on Control Applications, Anchorage, AK, September 2000, pp. 467–472.

[4] D. Nadig, S.S. Iyengar, “A new architecture for distributed sensor integration”, in Proceedings of IEEE Southeastcon’93, Charlotte, NC, April 1993.

[5] C. Shen, C. Srisathapornphat, C. Jaikaeo, “Sensor information networking architecture and applications”, IEEE Personal Communications, August 2001, pp. 52–59.

[6] J. Rabaey, J. Ammer, J.L. da Silva Jr, D. Patel, “Pico-Radio: Ad-hoc wireless networking of ubiquitous low-energy sensor/monitor nodes”, in Proceedings of the IEEE Computer Society Annual Workshop on VLSI (WVLSI’00), Orlando, Florida, April 2000, pp. 9–12.

[7] J..Rabaey, .J. Ammer, J.L. da Silva Jr, D. Patel, S. Roundy, “PicoRadio supports ad hoc ultra-low power wireless networking”, IEEE Computer Magazine (2000), pp. 42–48.

[8] F. Stajano and R. Anderson, “The resurrecting duckling: Security issues for ad-hoc wireless networks”, in 7th International Workshop on Security Protocols, vol. 1796, 1999, pp. 172–194, lNCS Volume 1796, Springer-verlag.

[9] L. Eschenauer and V. D. Gligor, “A key-management scheme for distributed sensor networks”, in Proceedings of the 9th ACM conference on Computer and communications security, Washington, DC, USA,November 18-22 2002, pp. 41–47.

[10] J. Spencer, “The Strange Logic of Random Graphs, Algorithms and Combinatorics”, Springer-Verlag 2000, ISBN 3-540-41654-4.

[11] H. Chan, A. Perrig, and D. Song, “Random key predistribution schemes for sensor networks”, in IEEE Symposium on Security and Privacy, Berkeley, California, May 11-14 2003, pp. 197–213.

[12] D. Liu and P. Ning, “Establishing pairwise keys in distributed sensor networks”, in Proceedings of the 10th ACM Conference on Computer and Communications Security (CCS), Washington, DC, USA, October 27-31 2003, pp. 52–61.

[13] W. Du, J. Deng, Y. S. Han, and P. K. Varshney, “A pairwise key pre-distribution scheme for wireless sensor networks”, in Proceedings of the 10th ACM Conference on Computer and Communications Security (CCS), Washington, DC, USA, October 27-31 2003, pp. 42–51.

[14] R. Blom, “An Optimal Class of Symmetric Key Generation System”, in Advances in Cryptology - Eurocrypt’84, LNCS vol. 209, p. 335-338, 1985.

[15] W. Du, J. Deng, Y. S. Han, S. Chen, and P. K. Varshney, "A key management scheme for wireless sensor networks using deployment knowledge", in IEEE INFOCOM, March 2004.

[16] D. Huang, M. Mehta, D. Medhi, L. Harn, "Location-aware key management scheme for wireless sensor networks", in Proceedings of the 2nd ACM workshop on Security of ad hoc and sensor networks, pp. 29-42, October 2004.

[17] A. Perrig, R. Szewczyk, V. Wen, D. Cullar, and J. D. Tygar, “Spins: Security protocols for sensor networks”, in Proceedings of the 7th Annual ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom), Rome, Italy, July 2001, pp. 189–199.

[18] G. Jolly, M. C. Kuşçu, P. Kokate, and M. Younis, “A Low-Energy Key Management Protocol for Wireless Sensor Networks”, in Proceedings of the eighth IEEE International Symposium on Computers and Communications (ISCC'2003).

[19] A. Perrig, R. Canetti, D. Song, and J. D. Tygar, “Efficient and secure source authentication for multicast”, in Proceedings of the Symposium on Network and Distributed Systems Security (NDSS 2001).

[20] F. Hu, J. Ziobro, J. Tillett, N. K. Sharma, “Secure Wireless Sensor Networks: Problems and Solutions”, in Journal of Systemics, Cybernetics and Informatics, volume 1, number 4.

[21] S. Zhu, S. Setia, S. Jajodia, “LEAP: Efficient Security Mechanisms for Large-Scale Distributed Sensor Networks”, in the 10th ACM Conference on Computer and Communications Security (CCS '03), Washington D.C., October, 2003, pp. 62-72.

[22] C. Karlof and D. Wagner “Secure routing in wireless sensor networks: Attacks and countermeasures”, in Proceedings of the 1st IEEE International Workshop on Sensor Network Protocols and Applications (Anchorage, AK, May 11, 2003).

[23] L. Zhou and Z. Haas, “Securing ad hoc networks”, IEEE Network Magazine, vol. 13, no. 6, November/December 1999.

[24] J. Hubaux, L. Buttyan, and S. Capkun, “The quest for security in mobile ad hoc networks”, in Proceedings of the ACM Symposium on Mobile Ad Hoc Networking and Computing (MobiHOC 2001), 2001.

[25] J. Kong, P. Zerfos, H. Luo, S. Lu, and L. Zhang, “Providing robust and ubiquitous security support for mobile ad-hoc networks”, in ICNP, 2001, pp. 251–260.

[26] M. G. Zapata, “Secure ad-hoc on-demand distance vector (SAODV) routing”, IETF MANET Mailing List, Message-ID: BC17B40.BBF52E09@nokia.com, Available at ftp://manet.itd.nrl.navy.mil/pub/manet/2001-10.mail, October 8, 2001.

[27] H. Luo, P. Zefros, J. Kong, S. Lu, and L. Zhang, “Self-securing ad hoc wireless networks”, in Seventh IEEE Symposium on Computers and Communications (ISCC ’02), 2002.

[28] J. Binkley and W. Trost, “Authenticated ad hoc routing at the link layer for mobile systems”, Wireless Networks, vol. 7, no. 2, pp. 139–145, 2001.

[29] B. Dahill, B. N. Levine, E. Royer, and C. Shields, “A secure routing protocol for ad-hoc networks”, Electrical Engineering and Computer Science, University of Michigan, Tech. Rep. UM-CS-2001-037, August 2001.

[30] J. Kong, H. Luo, K. Xu, D. L. Gu, M. Gerla, and S. Lu, “Adaptive security for multi-layer ad-hoc networks”, Special Issue of Wireless Communications and Mobile Computing, Wiley Interscience Press, 2002.

[31] C. Intanagonwiwat, R. Govindan, and D. Estrin, “Directed diffusion: A scalable and robust communication paradigm for sensor networks”, in Proceedings of the Sixth Annual International Conference on Mobile Computing and Networks (MobiCOM ’00), August 2000.

[32] Y. Yu, R. Govindan, and D. Estrin, “Geographical and energy aware routing: A recursive data dissemination protocol for wireless sensor networks”, University of California at Los Angeles Computer Science Department, Tech. Rep. UCLA/CSD-TR-01-0023, May 2001.

[33] B. Karp and H. T. Kung, “GPSR: greedy perimeter stateless routing for wireless networks”, in Mobile Computing and Networking, 2000, pp. 243–254.

[34] W. R. Heinzelman, A. Chandrakasan, and H. Balakrishnan, “Energy efficient communication protocol for wireless microsensor networks”, in 33rd Annual Hawaii International Conference on System Sciences, 2000, pp. 3005–3014.

[35] D. Braginsky and D. Estrin, “Rumour routing algorithm for sensor networks”, in First ACM International Workshop on Wireless Sensor Networks and Applications, 2002.

[36] S. Bandyopadhyay and E. Coyle, “An Energy-Efficient Hierarchical Clustering Algorithm for Wireless Sensor Networks”, in Proceedings of IEEE INFOCOM, April 2003.

[37] S. Basagni, Distributed Clustering Algorithm for Ad-hoc Networks, in International Symposium on Parallel Architectures, Algorithms, and Networks (I-SPAN), 1999.

[38] O. Younis, S. Fahmy, “Distributed clustering in ad-hoc sensor networks: A hybrid, energy-efficient approach”, in Proceedings of the IEEE Conference on Computer Communications (INFOCOM), Hong Kong 2004.

[39] T. Ito, H. Ohta, N. Matsuda and T. Yoneda, “A key pre-distribution scheme for secure sensor networks using probability density function of node deployment”, in Proceedings of the 3rd ACM workshop on security of ad hoc and sensor networks, 2005, pp. 69-75.

[40] Y. Mao, M. Wu, “Coordinated sensor deployment for improving secure communications and sensing coverage”, in Proceedings of the 3rd ACM workshop on security of ad hoc and sensor networks, 2005, pp. 117-128.

[41] W. Heinzelman, A. Chandrakasan, and H. Balakrishnan, “An Application-Specific Protocol Architecture for Wireless Microsensor Networks”, IEEE Transactions on Wireless Communications, Vol. 1, No. 4, October 2002, pp. 660-670.

[42] D. Niculescu and B. Nath, “Ad hoc positioning system (APS)”, in GLOBECOM. IEEE, November 2001, San Antonio.

[43] S. Capkun, M. Hamdi, J.P. Hubaux, “GPS-free positioning in mobile ad-hoc networks”, in Proceedings of Hawaii International Conference on System Sciences, January 2001.

[44] N. Bulusu, J. Heidemann, D. Estrin, “GPS-less low cost outdoor localization for very small devices”, Technical report 00-729, Computer science department, University of Southern California, April. 2000.

[45] B. Hofmann-Wellenhof, H. Lichtenegger, and J. Collins, “Global Positioning System: Theory and Practice”, Fourth Edition, Springer-Verlag, 1997.

[46] L.Zhou, J. Ni, V. Ravishankar, “Efficient key establishment for group-based wireless sensor deployments”, in Proceedings of the 4th ACM workshop on Wireless security, 2005, pp. 1-10.

[47] T. Dimitriou and I. Krontiris, “A localized, distributed protocol for secure information exchange in sensor networks”, in Proceedings of the 19th IEEE International Parallel and Distributed Processing Symposium (IPDPS'05) - Workshop 12 - Volume 13, April 2005.