TOWARDS A SECURE AND ENERGY EFFICIENT WIRELESS SENSOR NETWORK USING BLOCKCHAIN AND A NOVEL CLUSTERING APPROACH

by

NAZMUL ISLAM

Submitted in partial fulfilment of the requirements
for the degree of Master of Computer Science

at

Dalhousie University
Halifax, Nova Scotia
July 2018

To my wife Jahanara Pervin and our parents,

for supporting me in all odds with their guidance and inspiration.

Table of Contents

List of Tables

List of Figures

Abstract

Security has remained as one of the most crucial issues in Wireless Sensor Networks (WSNs) for many years. The emergence of new WSN based technologies, for e.g., smart homes and smart cities, have brought forward new requirements, such as service availability, immutability, and network transparency. Although the conventional designs and protocols of WSNs efficiently manage security, they seem to be limited in satisfying these requirements. Again in terms of energy efficiency, recent research has shown significant improvements by forming clusters prior to the selection of cluster heads. These improvements adopt different geometric fractals, such as the Sierpinski triangle, to divide the monitoring area into multiple clusters. However, performance of such approaches can be improved further by cognitive partitioning of the monitoring area instead of adopting random fractals.

This work proposes a Blockchain based protocol suite for WSNs to achieve service availability, immutability, and network transparency. It adopts a co-operative multiple Base Station system that minimizes the risk of network failure due to any attack on the Base Stations. Besides, a novel clustering approach that partitions the monitoring area in a cognitive way for balancing the energy consumption is also proposed. Its two-layered scrutinization process for the selection of cluster heads ensures minimum energy consumption from the network. Furthermore, it reduces the blind spot problem that escalates once the nodes start dying.

The proposed protocol suite in this work is simulated in terms of memory, communication, and computational overhead. The results show no significant falloffs in network lifetime or performance because of the adoption of Blockchain. The proposed clustering approach is also tested in terms of number of alive nodes per round, energy consumption of nodes and clusters, and distribution of alive nodes in the network. Results show a significant improvement in balancing the energy consumption among clusters and a reduction in the blind spot problem.

List of Abbreviations and Symbols Used

C_r	Units of energy consumed for running the receiver circuitry for one data unit
C_s	Units of energy consumed for running the sender circuitry for one data unit
$E_{K_{ii}}$	Encryption with the key K_{ii}
G	Vendermonde matrix
GK_i	The group key of i^{th} group
K_{ij}	Symmetric key between node N_i and N_j
N_i	Sensor node
P	A rectangular monitoring area
P_i	i^{th} equal partition of the monitoring area
S_i	Set of candidate nodes for CH from p_i
Sig_{BS_i}	Signature of BS_i
T	Transaction
TS	Time stamp
Z	Number of events in P within a given time frame
$\bar{D}$	Symmetric matrix
$\bar{D}^t$	Transpose of the matrix $\bar{D}$
$\bar{d}_{i,j}$	Distance between nodes N_i and N_j
$\bar{d}_{i,k_1,k_2,\ldots,k_k,j}$	Distance between nodes N_i and N_j via k_1, k_2, …, k_k
μ	Path loss exponent
c_{i_j}	j^{th} candidate nodes in p_i ($\in S_i$) for cluster head selection
g	Number of partitions, i.e., number of Cluster heads
m	Number of Base Stations
n	Number of sensor nodes
p_i	Adjusted area of P_i for energy balancing
$w(c_{i_j})$	Weight of node c_{i_j}

$w(p_i)$	Summation of all node's weight in p_i
ACO	Ant colony optimization
BC	Blockchain
BS	Base station
BSs	Base stations
CDMA	Code division multiple access
CH	Cluster head
CHs	Cluster heads
CSMA	Career sense multiple access
DoS	Denial of service
DSSS	Direct-sequence spread spectrum
FHSS	Frequency-hopping spread spectrum
FIS	Fuzzy inference system
FND	First node dead
GA	Genetic algorithm
GK	Group key
GS	Group size
HexxDD	Hexagonal cell-based data dissemination
HND	Half node dead
ID	Identifier
IoT	Internet of things

LEACH	Low-energy adaptive clustering hierarchy
LND	Last node dead
MAC	Message authentication code
P2P	Peer to peer
PoS	Proof of stake
PoW	Proof of work
PSO	Particle swarm optimization
SFLA	Shuffled frog leaping algorithm
TDMA	Time division multiple access
WSN	Wireless sensor network
WSNs	Wireless sensor networks

Acknowledgements

With deep respect and profound gratitude, I would like to thank my supervisor, Dr. Srinivas Sampalli, for his guidance throughout my thesis. This work would not have been completed without the precious support and encouragement from him. I am grateful because of the incredible attention and enthusiasm that I received from him during my research on the subjected area.

I would also like to thank Dr. Qiang Ye and Dr. Musfiq Rahman for giving valuable time to read my thesis and their feedback. Appreciation also goes to Saurabh Dey for his valuable ideas, assistance and feedback on a part of my thesis. Last but not the least, I offer my sincere thanks to everyone in my lab who supported me during my thesis works.

Chapter 1

Introduction

Wireless Sensor Networks (WSNs) are the significant part of the current research trends in smart homes [1, 2], smart industries [3–6], and smart cities [7]. Heterogeneous WSN with internet connectivity, termed as the Internet of Things (IoT) [8], is now empowering smart homes. It also facilitates industry automation by collecting the state of different machines on a single platform. Similarly, important organs of smart cities (e.g., smart parking [9]) would go non-functional without WSNs. It can be also deployed in a wide variety of other applications such as health care [10], environmental sensing [11], smart farming [12], and military defense [13]. In addition, WSNs play a significant role in emerging technologies, for e.g., big data [14] and cloud computing [15]. These recent applications involve commercial purposes and require service availability, data immutability, and network transparency along with energy efficiency in spite of severe resource constraints [16, 17] of wireless sensor nodes.

1.1 Motivation and challenges

Until now, a lot of research has been conducted on the security of WSNs [18] [19]. Most of these research works have focused on confidentiality, integrity, and authentication by developing lightweight cryptographic protocols. However, a majority of these works are limiting in service availability, immutability, and transparency of the network. Service availability refers to the continuation of operation even if some of the components in a WSN are compromised. In a multi-user WSN system (e.g., smart industries) service availability is a crucial issue as the Base Station (BS), from which the users get sensor data, is not free from vulnerability. Moreover, the sensor nodes solely rely on the BS for operational instructions, security materials, and data storage. In spite of such significances, few of the previous studies have discussed BS vulnerability and assured service availability by deploying multiple Base Stations (BSs). However, hardly any cooperation is found among the BSs in those works.

Consequently, associated data is lost and some parts of the network often go parentless while a BS is compromised. Immutability refers to the integrity of data while they are stored at the BSs [20]. Previous works have addressed the integrity of data on the path; however, they lack in providing a holistic security that also ensures data integrity at the BSs. Network transparency refers to the ability of a legitimate user to access the information as if it is stored in the local machine [21]. Users in a smart home application may want to know the status of the thermostat and control temperature while away from their home. Similarly, residents of a smart city may want to know the available parking spots nearby. Such applications need real-time data from the sensor networks, which is not addressed in most of the previous studies.

Typically, the communication among the nodes and the BSs can be established by a single or multi-hop path link. The limitation of single hop communication is that nodes far from the BSs lose energy rapidly due to the long communication range. On the other hand, nodes closer to the BSs die quickly in multi-hop communication since they forward all packets of the network to the BSs. This scenario is known as the hotspot problem and several approaches have been proposed to mitigate it by making the clusters unequal in terms of size. Among the unequal clustering approaches, fractal based ones ([22], [23]) have shown significant improvement in performance as they form clusters prior to the selection of CHs. Although these approaches successfully manage the hotspot problem, they suffer from the blind spot problem. This problem refers to the inability of capturing events due to the presence of dead nodes in the network. The lifetime of a network can be divided into two states: 1) the steady state- when all nodes are alive and 2) the declining state- when nodes start dying. As nodes are uniformly distributed, the network achieves high performance by capturing the desired number of events per unit of time in the steady state. In the declining state, the network cannot capture events uniformly as there persist dead nodes in the region. Hence performance of the network continuously degrades once the nodes start dying. For maintaining the performance, it is important to minimize the blind spot problem by shortening the declining state, which is lacking in a majority of existing works.

1.2 Contributions

This work proposes a protocol suite for WSNs that addresse service availability, immutability, and network transparency along with security to make the network more robust without sacrificing the efficiency. The protocol suite adopts cooperative BSs that ensures service availability while some of the BSs are compromised. Moreover, each BS in this work maintains a local Blockchain (BC) [24] to provide immutability of the stored data. Blockchain technology is currently receiving enormous interest due to its immutability and distributive nature and has been deployed widely in cryptocurrencies [25]. Although the level of distributiveness varies with different types of BC, immutability seems conspicuous in each of those types [26]. The network transparency is also ensured with the BC and the BSs by allowing user access to the status of the permitted nodes. Furthermore, a user can always verify the trustworthiness of a BS's data as all other BSs in the network share the same data with Blockchain technology.

This thesis also proposes a cognitive partition based unequal clustering approach to address the blind spot problem in WSNs. In addition to smaller closer clusters, the proposed approach ensures size-based balanced energy consumption. The selection process of cluster heads (CHs) in the proposed approach is divided into two layers, such as weight based selection of candidate CHs and cumulative distance based Cluster Head (CH) selection for each cluster from the candidates. The proposed approach guarantees CHs to have short distances among them and consumes least energy for packet forwarding. As a result, the lifetime of the network increases with a longer steady state than the declining state which reduces the blind spot problem. The contribution of this work is novel in the following aspects-

- Partitioning the network in a cognitive way to specify the size of the clusters for a balanced energy consumption.

- Adoption of two-layered scrutinization for the selection of CHs to guarantee minimal energy loss from the network.

- Shortening the duration of declining state to reduce the blind spot problem.

1.3 Thesis outline

The rest of the thesis is organized as follows: Chapter 2 reviews the recent works on related area. Chapter 3 presents the background theories of the proposed works. Chapter 4 presents the BC based WSN operational scheme in detail, i.e., its components, different operations of the proposed scheme, evolution of BC, and difference between Bitcoin BC and the BC used in the proposed work. Then, Chapter 5 details the proposed energy balancing cognitive partition based clustering approach. Simulation results and discussion along with feasibility and security analysis are given in Chapter 6. Finally, conclusions are drawn in chapter 7 with future research directions.

Chapter 2

Background

2.1 Wireless sensor networks (WSNs)

WSNs are the networks of spatially dispersed sensors aimed to sense the state of the surroundings. Typically, a WSN measures temperature, winds, humidity, sound, pollution levels, etc. of the environment and gather those data at a central location which is often termed as the sink or the Base Station (BS).

2.2 Network lifetime

It refers to the duration of the network operation. It is measured from the beginning of the network operation to the death of the last node.

2.3 Clustering in WSNs

Clustering in WSNs refers to dividing the network into multiple sub-networks each of which consists of member nodes and a CH. In WSNs, CHs work as the repeaters of typical computer networks. A member node collects data and sends to its CH that forward the same data to the BS. Clustering the WSNs can extend the lifetime by balancing the energy consumptions among the clusters.

2.4 Blockchain

Blockchain (BC), a distributed ledger system, re-emerged as an underlying technology of peer-to-peer electronic cash system [27] in 2008. Recently, BC is receiving more attention than before because of its potential applications beyond cryptocurrency, such as transaction of non-currency asset [28], smart contract [29], IoT [30], etc.

A BC consists of multiple blocks. Each block in a BC typically contains its ID, the ID of the previous block, and multiple transactions. The ID of the previous block in a

certain block creates a link between these two blocks. This relationship creates a chain of blocks as it continues to the first block. The ID of a block includes the hashed value of all transactions in the block, which is again included into the successive blocks. Hence the modification of any data in a block will affect all the successive blocks in the chain. Consequently, an attacker must modify all the successive blocks in the chain to tamper with a single transaction in a block, which is quite infeasible. Thus BC achieves the robust characteristic of immutability.

Currently, three types of BC are found in the literature, namely, public, private, and consortium BC [26]. All records in a public BC are visible to the public, whereas in private and consortium BC, only one party or a group maintains the BC, respectively. Again in a public BC, anyone can take part in the consensus process. On the other hand, it is only some selected entities in a consortium and only one entity in a private BC based system. Recently, different consensus algorithms are being used for different purposes. Some of the well-known consensus algorithms are: Proof of Work (PoW) for Bitcoin [27], Proof of Stake (PoS) for Peercoin [31], [32] for Ripple, etc. In a public BC, it becomes difficult to tamper with any data as a large number of nodes keep the BC stored. On the contrary, it is easier for an adversary to manipulate transactions in a consortium and a private BC. However, a public BC consumes a lot of time to propagate the transactions throughout the network because of involving a large number of nodes. Alternately, a private or consortium BC are more efficient in this perspective. Considering the pros and cons of different types, this work adopts consortium BC for WSNs. Comparison among the three BC types are summarized in Table 2.1.

2.5 Rahman and Sampalli's Proposal [33]

The protocol proposed by Rahman and Sampalli [33] establishes pairwise and group key among the sensor nodes and is lightweight enough to be appropriate for WSNs and IoT devices. It is used for the proposal in this work as an underlying cryptographic component for data security at the network layer. The proposal in [33] can be divided into *Key predistribution*, *Key agreement*, *Group key distribution* and *Key update*, which are briefly described below.

Key predistribution: The BS creates a $(\lambda+1)\times n$ matrix G over a finite field $GF(q)$

Table 2.1: Comparison among the different BC types

#	Properties	Public BC	Consortium BC	Private BC
1.	**Consensus determination**	All entities are allowed	Selected set of entities can participate	Only one entity determines consensus
2.	**Consensus process**	Permission less	Mutual	Permission required
3.	**Control**	Publicly distributed	Distributed among the members of a group	Centralised to one entity
4.	**Read permission**	Public	Only selected entities or public	Only one entity or set of entities or public
5.	**Robustness**	Nearly impossible to temper	Easier to temper than public BC	Easier to temper than public or consortium BC
6.	**Time consumption for transaction propagation**	High	Low	Low

and makes it publicly available. Here, n in the size of the network and $GF(q)$ is the large prime number to accommodate the key size. Linearly independent vendermonde matrix is used for constructing G by assuming distinct IDs of nodes as seeds. An example of G matrix for n nodes is shown in eq. 2.1. Now, the BS creates a $(\lambda+1)\times(\lambda+1)$ symmetric matrix $\bar{D}$ over $GF(q)$ and distributes A_i to each node N_i as its keying material. Here, A_i is the i^{th} column of A where $A = (\bar{D} \cdot G)^t$, and t denotes the transpose of a matrix.

$$G = \begin{bmatrix} 1 & 1 & 1 & \dots & 1 \\ 1 & 2 & 3 & \dots & n \\ 1^2 & 2^2 & 3^2 & \dots & n^2 \\ 1^3 & 2^3 & 3^3 & \dots & n^3 \\ & & \vdots & & \\ 1^\lambda & 2^\lambda & 3^\lambda & \dots & n^\lambda \end{bmatrix} \tag{2.1}$$

Key agreement: Two nodes willing to communicate with each other generate a secret pairwise key in this phase. For this purpose, they must know each other's IDs, i.e,

node N_i and N_j should know N_j's and N_i's ID respectively. They both then generate each other's corresponding columns in G, i.e., G_j and G_i respectively using these IDs as seed. Now, node N_i generates the pairwise key $K_{ij} = (G_j \cdot A_i)$. Similarly, node N_j also computes $K_{ji} = (G_i \cdot A_j)$. Here, K_{ji} and K_{ij} are symmetric as $K_{ij} = (G_j \cdot A_i) = G_i^t \cdot \bar{D} \cdot G_j = G_i^t \cdot \bar{D}^t \cdot G_j = G_i^t \cdot A_j^t = K_{ji}^t$. However, K_{ij} and K_{ji}^t are scalar elements and $K_{ji} = K_{ji}^t = K_{ij}$.

Group key: Group key allows a node to broadcast any message in the cluster. In [33], the BS takes the responsibility to establish the group key (GK) in each cluster. For g clusters in the network, the BS creates a set of group keys comprising g elements- $\{GK_1, GK_2, \ldots, GK_g\}$. Now, for each node N_i, the BS determines a cluster k, computes Y_i with a timestamp TS, and finally broadcasts M throughout the network. Calculations of Y_i and M are given in Eq. 2.2 and 2.3, respectively. Here, $E_{K_{ii}}$ refers to the encryption with key K_{ii} and MAC denotes the message authentication code. Once N_i receives the broadcast, it extracts Y_i, verifies the integrity from MAC, and gets GK_k.

$$Y_i = N_i||E_{K_{ii}}(GK_k)||TS||MAC_{K_{ii}}(N_i||E_{K_{ii}}(GK_k)||TS) \tag{2.2}$$

$$M = Y_1||Y_2||\ldots||Y_n \tag{2.3}$$

Key update: The BS periodically initiates the key update process for maintaining key refreshment. This process is also invoked if any node is revoked from the network. To update all keying materials of N_i ($N_i \notin R|R$: Set of revoked nodes), the BS first takes a new random $(\lambda{+}1) \times (\lambda{+}1)$ symmetric matrix $\bar{D}'$ over $GF(q)$ and computes $A_i' = (\bar{D}' \cdot G_i)^t$. Finally, the BS computes Y_i' as in eq. 2.4 where $H_i{=}\ A_i' + A_i$. Once Y's for all non-revoked nodes are prepared, the BS broadcasts M' (eq. 2.5). Each non-revoked node N_i extracts its part from M', verifies the content, and computes $A_i' = H_i - A_i$. To update the group keys, the BS selects a new set $\{GK_1', GK_2', \ldots, GK_g'\}$ and broadcasts like before.

$$Y_i' = N_i||E_{K_{ii}}(H_i)||TS||MAC_{K_{ii}}(N_i||E_{K_{ii}}(H_i)||TS) \tag{2.4}$$

$$M' = M'||Y_i; \forall N_i \notin R \tag{2.5}$$

The proposal in [33] deploys a single BS in the network. In single BS systems, the entire network gets compromised once the BS is compromised. Moreover, the

network has to trust the BS blindly and any tampering of data by the BS can remain undetected. Hence adopting muti-BS system with the BC provides two benefits- (1) network does not rely on one BS and can continue functioning even if some of the BSs are compromised and (2) it becomes very difficult for an adversary to tamper with any data in a BS as other BSs can detect the inconsistency in the BC.

2.6 Summary

In this chapter the theoritical background for the proposals in this thesis is presented briefly such as the BC and the proposal in [33]. BC is the base concept used in the proposed protocol suite, i.e., every transaction is managed by BC in this proposal. The proposal presented in [33] plays as the underlying cryptographic technique of this proposal.

Chapter 3

Related Work

3.1 Availability, immutability, and transparency in WSNs

The proposed protocol suite ensures service availability, immutability, and network transparency in WSNs by deploying multiple BSs and the BC technology. Recent works that have ensured related components of security with multiple BSs are summarized in this section.

Previously, several works have been done on multi-BS WSNs. According to [34], a single BS in large WSNs makes the power consumption inefficient; hence, the significance of deploying multiple BSs is undeniable. As nodes lose most of their energy while transmitting data to the BS [35], multiple BSs can reduce the energy consumption by reducing the traveling distance of the sensed data [36]. Tang et al. have proposed an approach in [37] that places sub-BSs at equal distances to create a virtual strip in the middle of the monitoring area. These sub-BSs work as rendezvous points for the main BS that collect and store data from the sensor nodes. Although this work seems to increase availability, it suffers from early energy depletion of nodes. Moreover, the placement of enhanced nodes along the virtual strip limits its applicability. Hexagonal cell-based Data Dissemination (HexDD), proposed in [38], deploys multiple BSs and provides network transparency. For this purpose, it constructs a hexagonal grid structure and also enforces BS mobility. However, some other works (e.g., [39] and [40]) have claimed their approaches outperform the proposal in [38] in terms of energy. Khan et al., using the similar virtual infrastructure as in [38], have tried to minimize the control data overhead and improve the service quality in [41] that scales according to the number of deployed nodes and data aggregation features. Bhattacharjee et al. have also adopted multiple BSs in [42] and focused on increasing network lifetime by solving the placement problem of BSs using the local search techniques. Reddy et al. in [43] and Dandekar et al in [35] have proposed multi-BS systems to shorten the hop distance. However, their focus is on reducing

the energy loss rather than the BSs' cooperation to provide availability. Also, the proposals in [44], [45], and [46] have deployed multiple BSs keeping network lifetime optimization and optimal data collection problem in mind.

3.2 Energy balancing unequal clustering

Previous works on unequal cluster formation in WSNs can be broadly categorized into probabilistic, deterministic and preset approaches. CHs are determined randomly in probabilistic approaches, whereas deterministic approaches adopt weight functions, fuzzy logic, heuristic techniques or a hybrid of these to determine the same. The preset approach simplifies its process by predetermining node locations, clusters, and CHs. The proposed approach presented in this work would be a new addition to the hybrid unequal clustering approaches as it includes both optimization and weight-based approach to select CHs. Recent works on deterministic unequal clustering approach are described below.

3.2.1 Weight based approaches

In these approaches, each node is assigned with a weight which is calculated based on different metrics such as node degree, residual energy, distance to the BS, etc. Typically, minimal weight is the criteria to select a node as a CH.

Several clustering approaches [47–56] have been proposed recently based on the weight-based technique. Among them, the approach proposed in [47] tries to balance the energy consumption to reduce the hotspot problem. It forwards data to the BS through relay nodes which are selected considering the residual energy. This approach allows regular nodes to join a cluster which has a CH with maximum residual energy and lesser distance to the BS. The approach proposed in [48] uniformly distributes the load throughout the network to reduce the hotspot problem. It divides the monitoring area into equal partitions and the nodes in a partition into unequal clusters. It also adopts a weight-based heuristic algorithm that takes residual energy, node degree, and distance to the BS as inputs to select the CHs. Similar to other approaches, clusters in the partition near to the BS become smaller in size with this approach. The proposed approach in [49] tries to balance the energy consumption by uniform distribution of the load. Here, the ratio of node's residual energy and its neighbor's average residual

energy is computed to select the CHs. In this approach, a cluster radius is determined by using the CH's residual energy and its distance to the BS. It introduces the concept of threshold distance for the CHs, where CHs having a distance to the BS lower then the threshold adopt single-hop data transfer. Otherwise, relay nodes are chosen based on the residual energy. The approach presented in [50] partitions the monitoring area into a number of hierarchical levels. It adopts a mathematical approach to construct unequal sized clusters; thus, improving the network lifetime. The proposal in [51] selects CHs through two steps, such as the random selection of tentative CHs and the selection of final CHs. Here, the tentative CHs are selected based on a probability model and the final CHs are selected based on their residual energy. In this approach, each sensor node preserves the minimum number of hop count to the BS which gives the optimal radius of a cluster. The approach presented in [52] selects its CHs based on the residual energy and distance to the BS. However, it only triggers the selection process once the residual energy of any current CH falls below a threshold level. It also employs relay nodes for CHs having a distance to the BS higher than a predefined value. The proposal in [53] determines cluster sizes based on the distance to the BS. It uses Dijkstra's algorithm [57] to find the shortest path route to the BS. The approach presented in [54] spatially distributes the clusters to balance the energy consumption in the network. For this purpose, it creates tracks around the BS where same sized clusters are formed in the same track. This approach considers the residual energy to select the candidate CHs. Final CHs are selected later from the candidates based on a distance metric rule. The proposed approach in [55] selects CHs based on the residual energy and coverage area, i.e., the more a node's sensing area covered by its neighbors, the higher its probability to be a CH. Finally, the approach in [56] considers the average energy of neighbor nodes beside of a node's residual energy to select it as a CH. Cluster formation in all of these approaches are similar to LEACH [58] and actuated after the selection of CHs.

3.2.2 Fuzzy logic based approaches

Fuzzy logic is also used in a number of protocol [59–63] for making decisions effectively, i.e, selecting the CHs and determining cluster sizes. For this purpose, it takes input parameters such as distance to the BS, centrality, distance from the neighbors,

node degree, residual energy, etc. and outputs CH selection probability and cluster size. The advantages of adopting fuzzy over classical approaches are: flexibility, low computational complexity, effective in terms of cost, memory, and design time.

The approach proposed in [59] uses random selection and residual energy to find the candidate CHs and final CHs respectively. This approach eliminates the hotspot problem by distributing loads with competition radius determination of the CHs. For this purpose, fuzzy logic is used which takes residual energy and distance to the BS as inputs. The output radius of a cluster is directly related to the CH's residual energy and distance to BS. The proposal in [60] also uses fuzzy logic for selecting the CHs and determining radii of the clusters. Here, the input parameters are the distance to the BS, node density, and residual energy, whereas the outputs are clusters' radii and the probable CHs. The final CHs are determined by a competition which requires exchange of messages. This approach uses Ant Colony Optimization (ACO) [64] to find the shortest path from a CH to the BS. The proposed approach in [61] uses Fuzzy Inference System (FIS) [65] to select the CHs in a distributed way which takes residual energy, link quality, and centrality of the node as inputs. This approach has made a significant improvement in WSN reliability by considering link quality while selecting the CHs. The fuzzy output is a value that indicates the probability of a node to become a CH. This approach also uses the scatter factor and the distance of a hypothetical hexagon to the BS for determining the number of CHs in that hexagon. The scatter factor is defined as the average distance of each node to its neighbor nodes in the hexagon. The higher the scatter factor, the more the CHs are required in that hexagon. The proposal in [62] uses probabilistic method to determine the tentative CHs and fuzzy logic to finalize the competition radii by considering node degree, residual energy, and distance to the BS. Node degree and residual energy are used again to determine the final CHs. In this approach, nodes can join a cluster based on CH's degree and distance to the BS. Finally, [63] presents an approach that takes the same input as FIS to determine both the CHs and cluster sizes. Clustering approaches associated with all these techniques are also similar to LEACH [58], i.e., clusters are formed after the selection of CHs.

3.2.3 Heuristic based approaches

Recently, a number of proposal have been made based on heuristic unequal clustering [66–72]. The proposed approach in [66] is a centralized unequal clustering approach that selects the CHs with Particle Swarm Optimization (PSO) algorithm [73] and then forms the smaller clusters near to the BS. It uses greedy algorithm for routing the data from a CH to the BS. For this purpose, relay nodes are selected based on the distance to the BS and the residual energy. The approach presented in [67] computes the number of CHs and their positions with Genetic Algorithm (GA) [74] in order to reduce the energy consumption from the network. Its operation is divided into rounds and each round consists of a setup phase and a steady state phase. The BS determines a number of CHs and their positions with GA in the first phase, whereas the route from the source node to the BS is determined in the second phase. This approach allows a node to send data directly to the BS if the node's distance to the BS is smaller than the distance to its CH. TDMA [75] and CDMA [76] schedules are used in this approach for intra-cluster and inter-cluster communication respectively. Similarly, the approach in [68] also divides its operation into several rounds each of which again consists of a setup phase and a steady state phase. In the setup phase, BS selects the CHs and forms clusters based on nodes' location, residual energy and the number of neighbors. The steady state phase forwards data to the BS through an optimal route. Similar to the approach in [67], CHs in this approach also uses TDMA schedule for intra-cluster communication. The proposal in [69] forms clusters of various sizes according to the residual energy and selects CHs with Shuffled Frog Leaping Algorithm (SFLA) [77]. Its operation is divided into the cluster establishment phase and the data transmission phase. Selection of the CHs in the cluster establishment stage is an optimization problem. In the data transmission phase, it adopts a greedy approach to find the route from source node to the BS. The operation of the approach presented in [70] can be divided into three phases, namely, the setup phase, neighbor finding phase, and the steady state phase. In the first and second phase, nodes are classified into different layers and messages are broadcast to find neighbors. This broadcast follows non-persistent Carrier Sense Multiple Access (CSMA) [78] protocol to accecss the medium. The third phase again can be divided into CHs selection, cluster formation, and data delivery. This approach uses fuzzy logic to select the CHs and ACO to

find the optimal route for data delivery. Here, input parameters of fuzzy logic are the number of neighboring nodes, residual energy, and the link quality. On the other hand, ACO uses distance to the BS, residual energy, delivery likelihood, and queue length to select the relay nodes. The approach presented in [71] proposes a unequal clustering and routing technique based on chemical reaction optimization [79]. It selects the CHs based on the optimization approach presented in [79] and assigns other nodes to the CHs based on a derived cost function. It also proposes a routing algorithm which is also based on the technique of [79]. The approach proposed in [72] combines an unequal clustering mechanism [80] to determine cluster sizes and a multi-objective immune algorithm [81] to produce routing tree. The cluster sizes are determined based on the residual energy and distance to the BS. Thus, these approaches apply different heuristic optimization methods to find the CHs and to determine the cluster size. In these approaches, clusters are formed after the selection of CHs.

3.2.4 Hybrid approaches

Among the recent works in hybrid unequal clustering approaches, the proposal in [22] focuses on equalizing the energy consumption from every cluster. For this purpose, it reverses the cluster formation steps by creating the clusters first then assigning the CHs to them. Hence, three phases of clustering the network in this approach are performed in sequence- cluster formation, CH selection and data transmission. In the cluster formation phase, a Sierpinski triangle (appendix A) is used to create smaller clusters near to the BS. While selecting the CHs, it considers node degree, residual energy and distance to the BS. On the other hand, the proposal in [82] adopts a voting scheme to construct unequal clusters and selects the CHs based on the residual energy, topology, and transmission power. However, its CH selection is a distributed approach unlike the approach presented in [22]. The recent works in deterministic unequal clustering approach are summarized in Table 3.1.

3.3 Limitations of the previous works

Although a few works have been done on service availability, immutability and network transparency by deploying multiple BSs, none of them provides absolute availability of the data. Once a BS is compromised, the associated data also become

Table 3.1: Summary of the deterministic unequal clustering approaches in WSNs.

Proposals	**Homogeneous node type**	**Distributed CH selection**	**CH selection process**
[47]	√	√	Weight
[48]	√	√	Weight
[49]	√	√	Weight
[50]	√	√	Weight
[51]	√	√	Weight
[52]	√	√	Weight
[53]	√	√	Weight
[54]	√	√	Weight
[55]	√	√	Weight
[56]	√	√	Weight
[59]	√	√	Fuzzy
[60]	√	√	Fuzzy
[61]	√	√	Fuzzy
[62]	√	√	Fuzzy
[63]	√	√	Fuzzy
[66]	√	×	Heuristic
[67]	√	×	Heuristic
[68]	√	×	Heuristic
[69]	√	×	Heuristic
[70]	√	×	Heuristic
[71]	×	√	Heuristic
[72]	√	√	Heuristic
[22]	√	√	Hybrid
[82]	√	√	Hybrid

unavailable in those works. Moreover, none of them has user-oriented design to support smart homes or smart cities. Furthermore, it becomes difficult to detect data forgery of any BS as no one is aware of the data contained by that BS.

Again the limitation of recent unequal clustering approaches is that their procedures of increasing the network lifetime prolong the declining state which introduces blind spot problem in the network. Declining state refers to the last stage of a network lifetime that begins when the nodes start dying. A long lasting declining state in a given lifetime can degrade the performance of any clustering approach. The approach proposed here tries to keep the declining state short by maintaining more equivalent residual energy in nodes after each round. For this purpose, it divides the monitoring area into several partitions before the selection of CHs which is similar

to the approach presented in [22]. However, in the proposed approach, clusters are formed by cognitive partitioning instead of adopting fractals. In addition, path length connecting potential CHs and the BS is counted for the selection of CHs.

3.4 Summary

This chapter summarizes the recent studies in the related area. Firstly, the works done on multi-BS WSNs are presented here. It is found that, though there are several works on multiple BS for ensuring service availability to some extent, a very few works have been done on immutability and network transparency. Secondly, recent works on unequal deterministic clustering approaches are also summarized. It is seen that among the four categories, the proposed approach relates to the hybrid unequal deterministic clustering approaches.

Chapter 4

The proposed protocol suite for WSNs

This chapter first outlines the major components of the protocol suite and then describes the operations of each protocol. Evolution of the BC with the proposed protocol suite is presented thereafter. Finally, the distinguishing characteristics of the BC used in this proposal are analyzed.

4.1 Components

4.1.1 Sensor node

The proposed scheme consists of n sensor nodes uniformly distributed throughout the monitoring area. Nodes get credentials from the BSs that are essential to form pairwise keys after the deployment. In addition, nodes also get group keys, which they use for intra-cluster communication. Each node senses data and sends to the BSs through CHs. Moreover, the BSs can query any sensor node's current status. Upon receiving such a query, sensor nodes sense the current data and pass to the BSs through CHs.

4.1.2 Cluster Heads (CHs)

CHs are selected by the BSs among the sensor nodes based on different factors, such as residual energy, distance to the nearest BS, etc. The objective of CHs is to convey the sensed data from the sensor nodes to the BSs.

4.1.3 Base Stations (BSs)

The roles of the BSs are similar to that of the conventional WSNs, i.e., initializing sensor nodes, selecting CHs, distributing group keys, and collecting data from the monitoring area. However, the proposed protocol suite allows cooperative BSs that manage all the data with the BC technology. Hence the entire network does not

get affected albeit some of the BSs are compromised. Again, it becomes hard for an adversary to modify any data as they are saved with the BC technology. The BSs are also responsible for controlling the user access to the sensor data.

4.1.4 Users (Us)

Users are the part of the network who are permitted to access the sensor data. For each user, the BSs maintain a record that indicates the accessible nodes. The proposal in this work includes two major protocols that involve users, such as (1) Access: getting the data record of accessible sensor node(s) and (2) Monitor: getting the status from accessible sensor node(s).

4.1.5 Transactions (Ts)

To accommodate the BC technology, the proposal in this work treats any message that comes to or goes from the BSs as a transaction. Hence all messages exchanged among the users, nodes, and BSs are treated as transactions. However, messages that do not involve any BS (e.g., CH to member node communications) are not treated as transactions. The structure of a transaction is shown in Table 4.1. The first field in the transaction T holds the previous transaction number committed by the same sensor node or user. With this field, all transactions of the same sensor node or user are linked together; hence, it becomes easier to retrieve those data. The second field in T indicates the sequence number of the transaction. This value increases with the new addition of transactions and does not depend on blocks. That is, if the value was v for the last transaction in the previous block, it becomes $v + 1$ for the first transaction of the next block. The third field contains the node or user ID for which the transaction is committed. In the proposed scheme all nodes and users are assigned with distinct IDs. The fourth field in T indicates the type of the transaction. This work considers five types of transactions and uses distinct values to indicate each of them. 'Genesis' is the very first transaction that is committed for initializing any entity in the network. As each node gets secret credentials in that phase, one genesis transaction for each node is created and saved in the block. 'Store' transaction is committed while a node sends data to the BSs. 'Access' and 'Monitor' transactions are committed by the users when they want to retrieve data record and know the status of a node respectively. If

Table 4.1: Structure of a Transaction

Previous Tx	Tx number	Device ID(s)	Tx type	SigReq	Data
			0=Genesis 1=Store 2=Access 3=Monitor 4=Update		

Tx: Transaction

the manager in a smart industry, for e.g., wants to retrieve the day long temperature record of a refrigerator, an 'Access' transaction is committed. On the other hand, if she wants to know the current temperature, a 'Monitor' transaction is committed. The 'Update' transaction is committed when the BSs revoke some of the nodes and refresh the secret credentials. The fifth field contains the signature of the user for committing 'Access' or 'Monitor' transactions that ensures one of the major security requirements, namely, 'non-repudiation'. Finally, the sixth field contains the data associated with the transaction. To understand the format of T, we can consider a user U_i wants to monitor the node n_i just after joining the network. Hence for U_i, the committed transaction would be $\langle l,\ l+1,\ U_i\ ,\ 3,$ Signature of $U_i\ ,$ status of $n_i\rangle$ while U_i's previous transaction was $\langle 0,\ l\ ,\ U_i\ ,\ 3,$ Signature of $u_i,\ 0\rangle$.

4.1.6 Policy

Policies are saved in the blocks to indicate the permissions of the sensor nodes and the users in a network. A new policy is created on any change in the network, for e.g., node's credentials update, node or user revocation, etc. BSs always refer to the latest policy in the chain to grant any request. Table 4.2 shows the structure of a policy where each record is checked upon receiving any request. In the beginning, BSs keep all node IDs in the policy to send them updates; however, discard revoked IDs in new policies later on.

4.1.7 Block and Blockchain (BC)

For this work, a block can be defined as a fixed sized collection of sequenced transactions and policies. In detail, blocks are considered to have a certain capacity and can

Table 4.2: Structure of the policy.

Requester	Request for	Device ID(s)	Action
	⋮		
U_h	Access	⟨ List of node IDs ⟩	Allow
	⋮		
$N_1 ... N_n$	Update	⟨ List of BS IDs ⟩	Allow

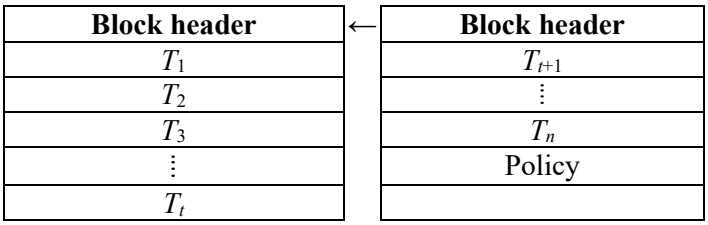


Figure 4.1: Structure of the BC used in the proposed protocol suite.

contain a fixed number of transactions and policies. Once a block is filled, transactions must be saved in the new block. Moreover, each block contains a block header that includes the hash of the previous block. Thus, a chain of blocks is created that ensures the immutability of the data. Fig. 4.1 shows the structure of the BC used in this work.

4.2 The proposed protocols

As the groundwork of BC based WSNs, the proposed suite includes six protocols: *initialization*, *group key (GK) establishment*, *node revocation*, *key update*, *store* and *monitor*. Description of each of these goes bellow.

4.2.1 Initialization

A base station BS_i generates the matrix A with $\bar{D}$ and G, as in [33], and loads the i^{th} column of A and G, i.e., A_i and G_i respectively to node N_i. BS_i then puts A_i and G_i into a transaction, saves to the current block, and securely transfer the transaction to other BSs. That is, to form a transaction, BS_i concatenates G_i and A_i and generates $T = \langle 0,\ i,\ N_i,\ 0,\ 0,\ G_i||A_i \rangle$. Here, the first value indicates N_i's initialization and no transaction is committed for N_i before, i is the current transaction number, i.e.,

$i-1$ transactions are committed so far in the BC, N_i is the node ID for which the transaction is committed. The fourth 0 indicates the Genesis type of the transaction. BS_i does not require any signature from the sensor nodes and puts 0 in the fifth position of the transaction. It uses the last field to save the exchanged data. BS_i also updates other BSs about T. Thus, the network can operate with other BSs if BS_i is compromised. To securely inform other BSs about T, BS_i first puts signature (Sig_{BS_i}) on T and then encrypts with other BSs' public keys (K^+s). Upon receiving, BSs decrypt the message with their corresponding private keys and then with BS_i's public key to extract T. Once each N_i is initialized, BS_i creates a pointer to the policy list in the current block. The policy initially contains a list of users and the related rules. Node IDs are also included in the policy that assist the BSs to decide which nodes are eligible to get the updates and which are revoked. Initially, all nodes are included in the eligible list. Finally, BS_i transfers the policy to other BSs in the same way as transactions. Here, BS_i could initialize all N_i first and then securely transfer the batch of Ts along with the policy to reduce the waiting time of N_i. Fig. 4.2 illustrates the initialization process.

4.2.2 Group key (GK) establishment

Credentials that BS_i provides during the initialization, allow nodes to generate pairwise keys without exchanging any message. However, using pairwise keys may become resource consuming in terms of time and energy. For instance, consider CHs request their members' status periodically. In a pairwise key system, a CH has to compute the key for each member of the cluster and broadcast a long message. Similarly, each member has to derive its own part and then decrypt to get the request. Thus, the process becomes time and energy consuming; hence, a group key for each cluster is significant.

The group key enables a node to securely and efficiently communicate with other members in the cluster. Fig. 4.3 illustrates the proposed group key establishment protocol. After the initialization and deployment, BS_i starts the cluster formation process which is described in the following chapter. BS_i sends GKs while forming the clusters. For this purpose, BS_i takes a random key and encrypts with K_{ii} which is computed as K_{ii}=$G_i \cdot A_i$ for N_i. BS_i then concatenates the encryptions generated

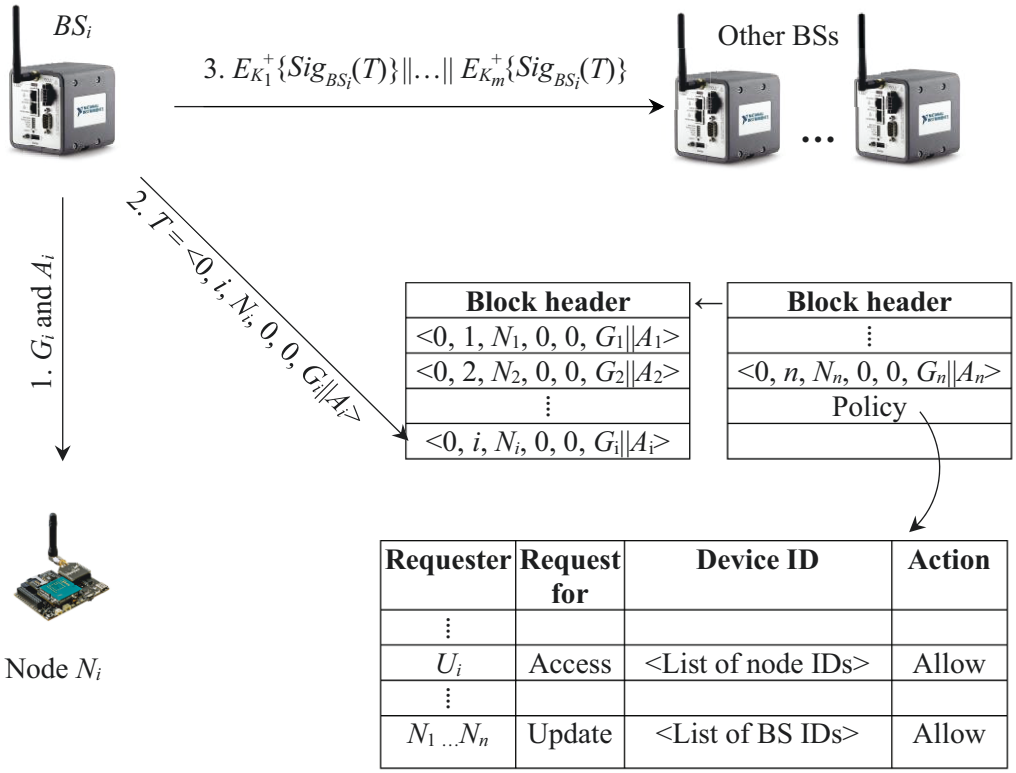


Figure 4.2: Initialization by BS_i.

for all members of the group and broadcast the message. Now, BS_i generates a transaction for each of the encryptions and saves into the current block. That is, BS_i generates the transactions $\langle 1,\ n+1,\ N_1,\ 0,\ 0,\ GK_j\rangle$, ...,$\langle h,\ n+h,\ N_h,\ 0,\ 0,\ GK_j\rangle$ for N_1 ...N_h, given that N_1 ...N_h are members of the same cluster. Here, the previous transaction number for N_1 is 1 which was committed during the initialization and its current transaction number is n+1 as the last transaction committed in the previous operation was n. Thus, BS_i distributes group keys to the members of each cluster in the network. Finally, BS_i updates the policy in the block that includes the 'Monitor' permission of CHs on their group members. This update is important as it indicates CHs' eligibility to have members current data while forwarding to the BSs. BS_i also updates other BSs about the transactions.

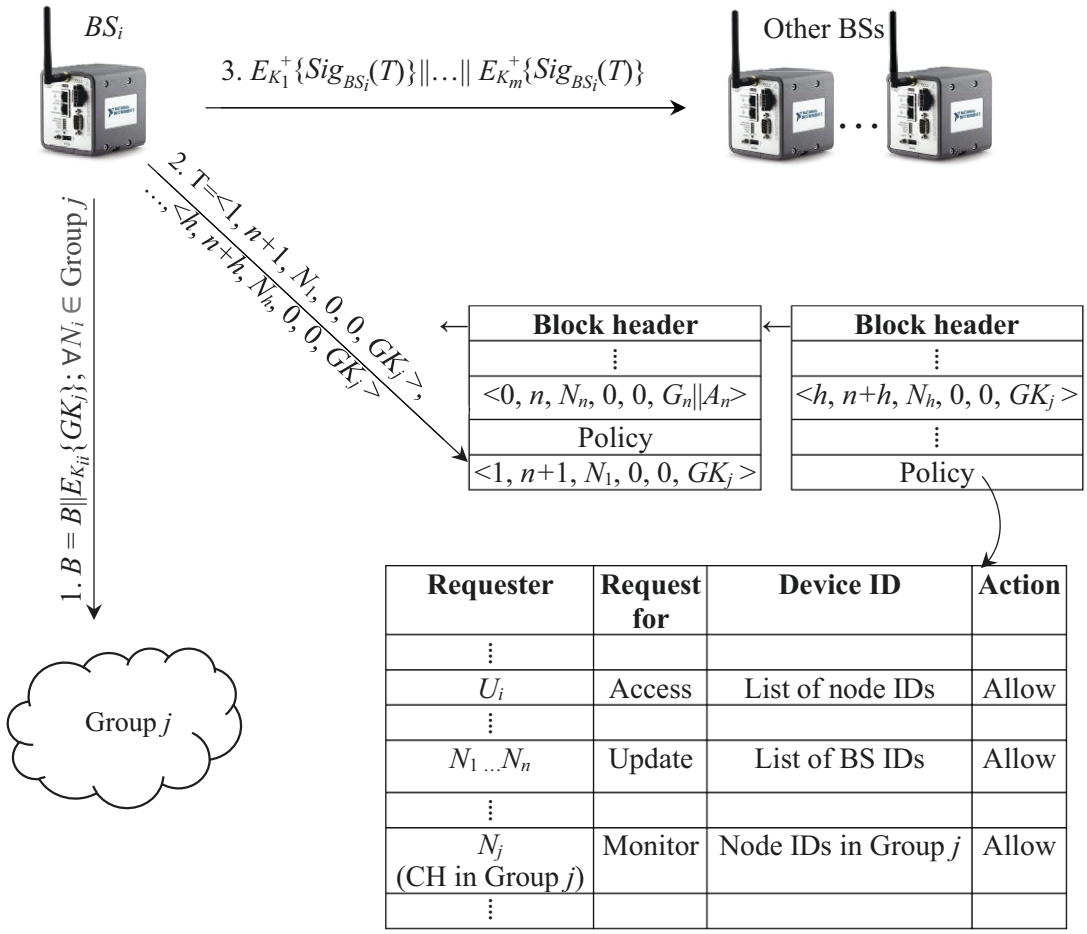


Figure 4.3: Group key establishment.

4.2.3 Node Revocation

This protocol is triggered when there is any compromised node in the network. It is assumed that there is an intrusion detection system present in the network that continuously monitors for intruders and detects any compromised node. An intruder may compromise N_i to leak its status or group key or to inject false data in the BC. Hence it becomes important to isolate N_i from the network as soon as it is detected as compromised. To revoke a node, BS_i first includes its ID into a set R that holds the IDs of all revoked nodes in the network. Then BS_i updates its policy that marks all node IDs except those in R as eligible to get updates and informs other BSs about the revocation. Finally, BS_i initiates the key update protocol where all nodes except the revoked ones get updated keys. Thus, the revoked nodes become separated from

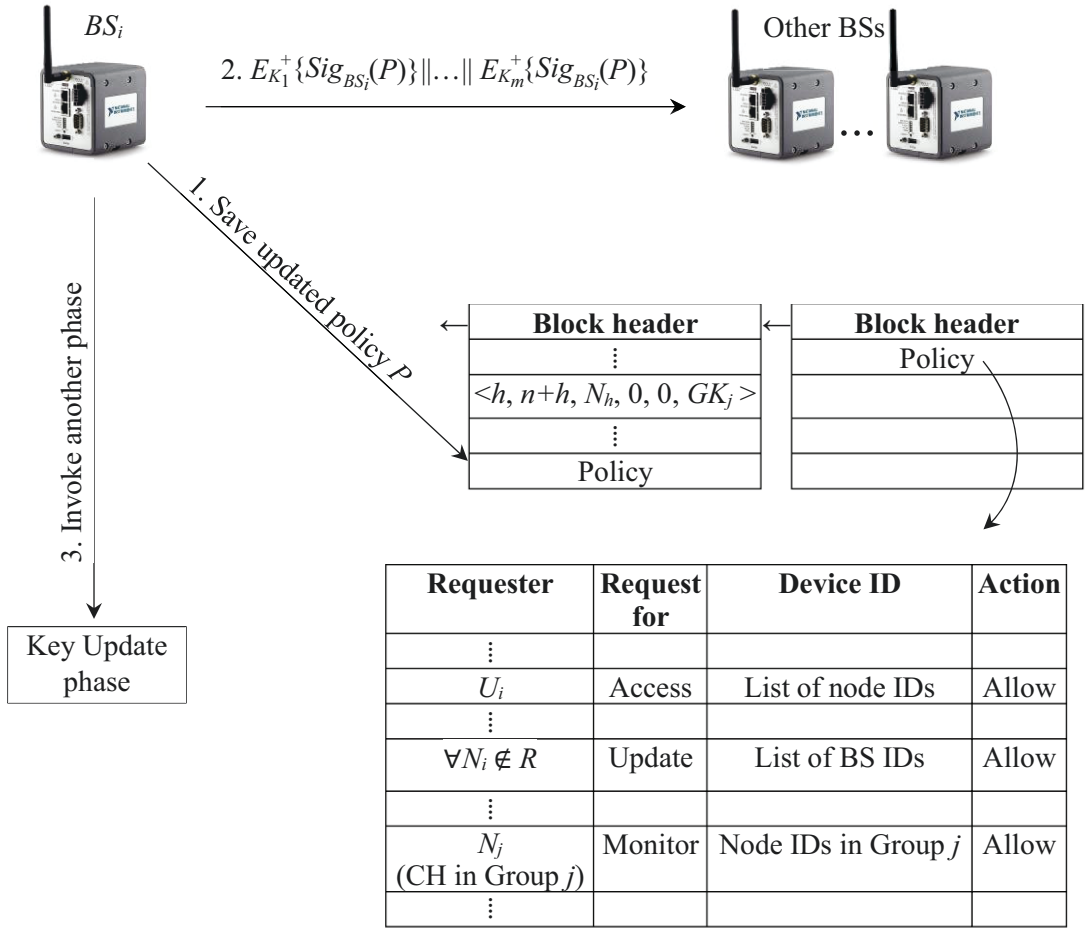


Figure 4.4: Node revocation by BS_i.

the network. Fig. 4.4 illustrates the revocation process.

4.2.4 Key Update

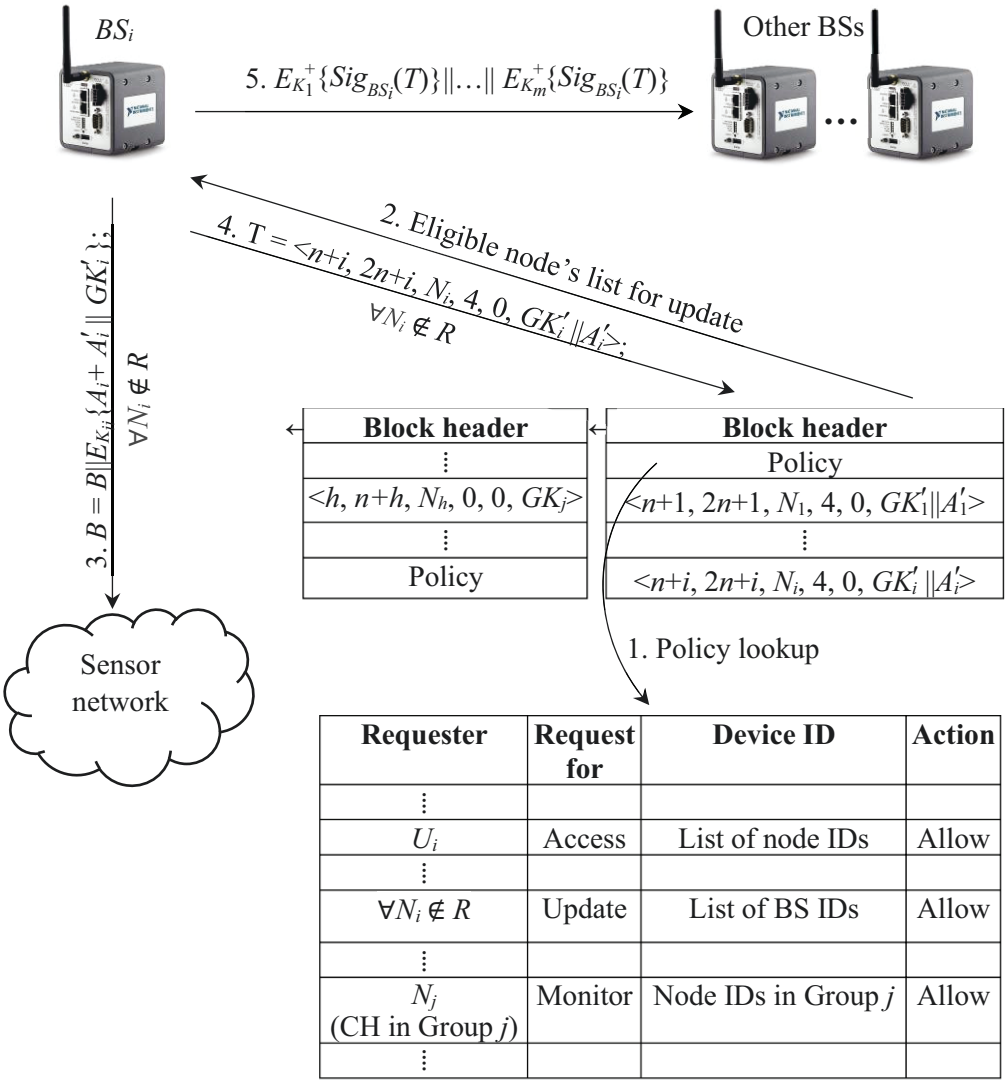
BS_i initiates the key update when there are compromised nodes in the network or new clusters are about to form. It can be initiated periodically also to ensure key freshness. It isolates the compromised nodes from the network. The process associated with this protocol satisfies the following security requirements- (1) *freshness*: ensures updated keys periodically for all nodes, (2) *confidentiality*: ensures that no external party can reveal exchanged messages, moreover, nodes in one group cannot reveal any message of another group, (3) *forward and backward secrecy*: no member leaving the network

or a group can reveal any associated message in the future, similarly, any node joining the network or a group cannot know the previously exchanged messages.

To update the keys, BS_i first looks into the latest policy and derives the non-revoked node IDs. Now, for each non-revoked node N_i, BS_i calculates $E_{K_{ii}}\{A_i + A'_i||GK'_i\}$, where A_i is the previous secret assigned to N_i and $A'i$ is the new secret for N_i. BS_i also concatenates the new group key GK'_i with the message before encryption. After preparing new secrets for all non-revoked nodes, BS_i concatenates and broadcasts them in a single message. Sensor nodes, upon receiving the message, retrieve their part, decrypt, and get the new secrets. BS_i creates a transaction for each non-revoked node along with their new secrets. In other words, BS_i creates the transaction $\langle n+1,\ 2n+1,\ N_1,\ 4,\ 0,\ GK'_1||A'_1\rangle$ for node N_1. Here, $n+1$ is the last transaction number for N_1 while receiving GK_1 and $2n+1$ is the current transaction number after the initialization and GK establishment of n nodes. BS_i uses the code-'4' to indicate that the transaction is committed for an 'Update' action. It also saves the new secret A'_1 assigned to N_1 into the transaction. Same as before, BSs does not require signatures from any sensor node upon sending the updates. Finally, BS_i informs other BSs about the transactions. Fig. 4.5 illustrates the process explained above.

4.2.5 Store

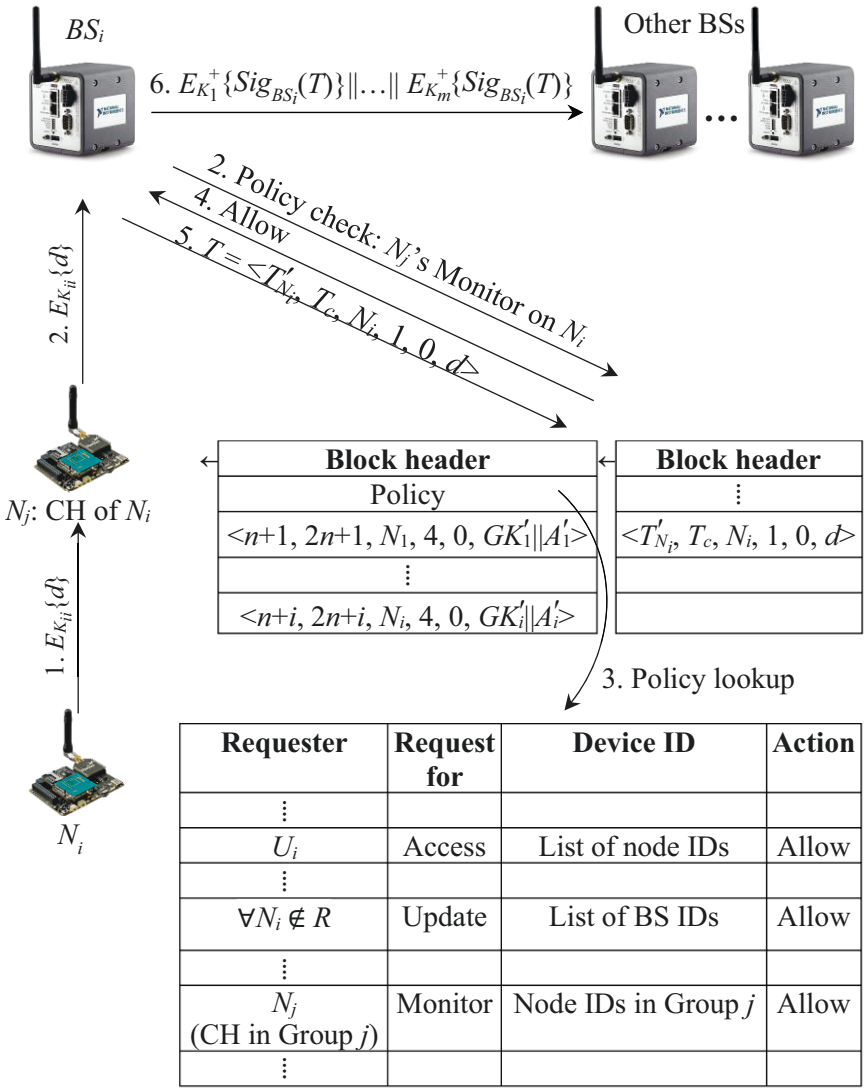
The status of N_i is stored in the BC with this protocol. It is initiated when N_i sends its sensed data to the BSs through its CH. Once BS_i receives the data, it checks for the 'Monitor' permission of the CH on N_i. This checking allows BS_i to identify an anomaly in the network, i.e., CH in one group is forwarding the message of another group, an intruder is acting as CH, etc. The permission 'Monitor' has two-sided usage in this work: (1) for the user requesting the status of a node and (2) for the node requesting to store data in the BC. Both the user and the CH in first and second case needs to have the 'Monitor' permission on the node to get its status. On a successful permission match in the policy, BS_i creates a transaction for the data, saves to the current block in the chain, and securely transfers the transaction to other BSs. Namely, N_i sends $E_{K_{ii}}\{d\}$ to its CH first to get the data d saved in the BC. Here, $E_{K_{ii}}\{d\}$ is the encryption of d by the key k_{ii}. Upon receiving $E_{K_{ii}}\{d\}$, CH forwards



Requester	Request for	Device ID	Action
⋮			
U_i	Access	List of node IDs	Allow
⋮			
$\forall N_i \notin R$	Update	List of BS IDs	Allow
⋮			
N_j (CH in Group j)	Monitor	Node IDs in Group j	Allow
⋮			

Figure 4.5: The key update process.

the same to the BSs. This process should address several vulnerabilities such as correctness of the forwarded message and reliability of the CH. However, these issues are not discussed in this work. After receiving the data and a successful policy check, BS_i decrypts the message to get d and saves the transaction $T = \langle T'_{N_i}, T_c, N_i, 1, 0, d\rangle$ for N_i. Here, T'_{N_i} and T_c are the previous transaction of N_i and the current transaction number respectively. Finally, BS_i sends $E_{K_1^+}\{Sig_{BS_i}(T)\} \|\ldots\| E_{K_m^+}\{Sig_{BS_i}(T)\}$ to other BSs and each BS saves the transaction T into the local BC. Fig.4.6 shows the overall storing process of data d from N_i to the BC.



Requester	Request for	Device ID	Action
⋮			
U_i	Access	List of node IDs	Allow
⋮			
$\forall N_i \notin R$	Update	List of BS IDs	Allow
⋮			
N_j (CH in Group j)	Monitor	Node IDs in Group j	Allow
⋮			

Figure 4.6: Data storing by a sensor node N_i.

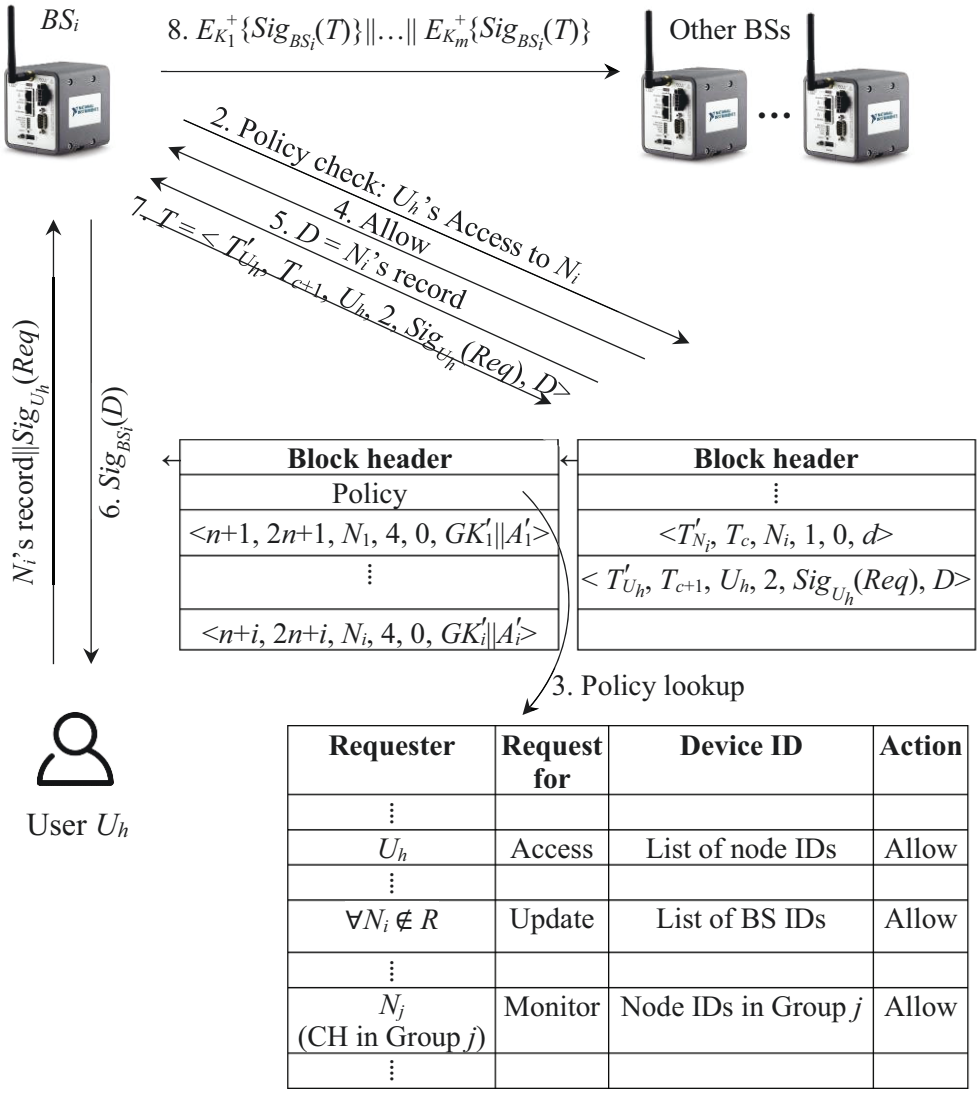
4.2.6 Data Access

'Data access' refers to acquiring the data records of a single node or a set of nodes from the BC. To access the record of a particular node or a set of nodes, a user first sends the request to BS_i along with her signature on it. Upon receiving the request, BS_i checks the eligibility of the user and retrieves the requested data series. Now, BS_i signs the record and sends to the user. Again, several security requirements, for e.g., integrity, authentication, verifiability, etc., should be satisfied while sending the

data. However, this work fulfills some of these requirements only to keep the focus on the basic works of a user-oriented BC based WSN. BS_i forms a transaction with the retrieved data and saves in the BC. Unlike the previous operations, BS_i now keeps the signature of the user in this transaction for non-repudiation. Storing the signatures in other protocols is avoided as nodes' credentials to form the signatures are already known to the BSs. Hence BS_i itself can form the signatures on behalf of the legitimate nodes while forming transaction. Thus, there is no requirement to maintain the nodes' signature in transactions committed for those protocols. Finally, BS_i securely informs other BSs about the transaction. That is, to access node N_i's record, user U_h forms a request (Req) and puts her signature on it as $Sig_{U_h}(Req)$. U_h then concatenates and sends "N_i's record$\| \; Sig_{U_h}(Req)$" to the BSs. BS_i checks the eligibility of U_h into the current policy for 'Access' operation. Then, BS_i retrieves the requested record D and sends along with BS_i's signature on it. Here, retrieving record is a fast process as the transactions of a node are linked together in the BC. BS_i then forms a transaction T that includes $Sig_{U_h}(Req)$ and D along with T_{U_h} and T_{c+1}. Here, T_{U_h} and T_{c+1} are the previous transaction of U_h and the sequence number of the current transaction respectively. BS_i uses the value 2 in transaction type field to indicate an access operation. Finally, BS_i sends $E_{K_1^+}\{Sig_{BS_i}(T)\} \; \| \; \ldots \| \; E_{K_m^+}\{Sig_{BS_i}(T)\}$ to all BSs to inform about T. Fig.4.7 illustrates the process described above.

4.2.7 Monitor

This protocol also involves users and is similar to the 'Access' operation. However, monitor operation refers to requesting a node or a set of nodes to sense the status and transmit to the user. Hence a user gets data directly from the sensor node and does not involve accessing any record from the BC. In this operation, BS_i checks the list of node IDs on which the user has monitor permission. If the user is requesting the status of a permitted node, BS_i sends a message to the node asking for its status. Upon receiving the message, node senses the status and securely sends to BS_i. In turn, BS_i puts a signature on the data and forwards to the user. Then BS_i forms a transaction with the request and the response data. Finally, it informs other BSs about the transaction through a secure process. Fig. 4.8 illustrates the process



Requester	Request for	Device ID	Action
⋮			
U_h	Access	List of node IDs	Allow
⋮			
$\forall N_i \notin R$	Update	List of BS IDs	Allow
⋮			
N_j (CH in Group j)	Monitor	Node IDs in Group j	Allow
⋮			

Figure 4.7: Data access by the user U_h.

explained above. Firstly, user U_i sends a request (Req) with her signature- "N_i's status$\|$ $Sig_{U_i}(Req)$" to BS_i. Then BS_i looks into the current policy table for U_i's eligibility. As U_i is requesting to monitor a permitted node, BS_i commands N_i to send its status. N_i senses the current data d', encrypts and sends it to BS_i. Here, N_i uses the key K_{ii} as before for encryption. Once received, BS_i decrypts to get d', prepares the signature $Sig_{BS_i}(d')$ and sends to U_i. After that BS_i creates the transaction T=$\langle T'_{U_i}, T_{c+2}, U_i, 3, Sig_{U_i}(Req), d'\rangle$, saves into the BC, and transfers T

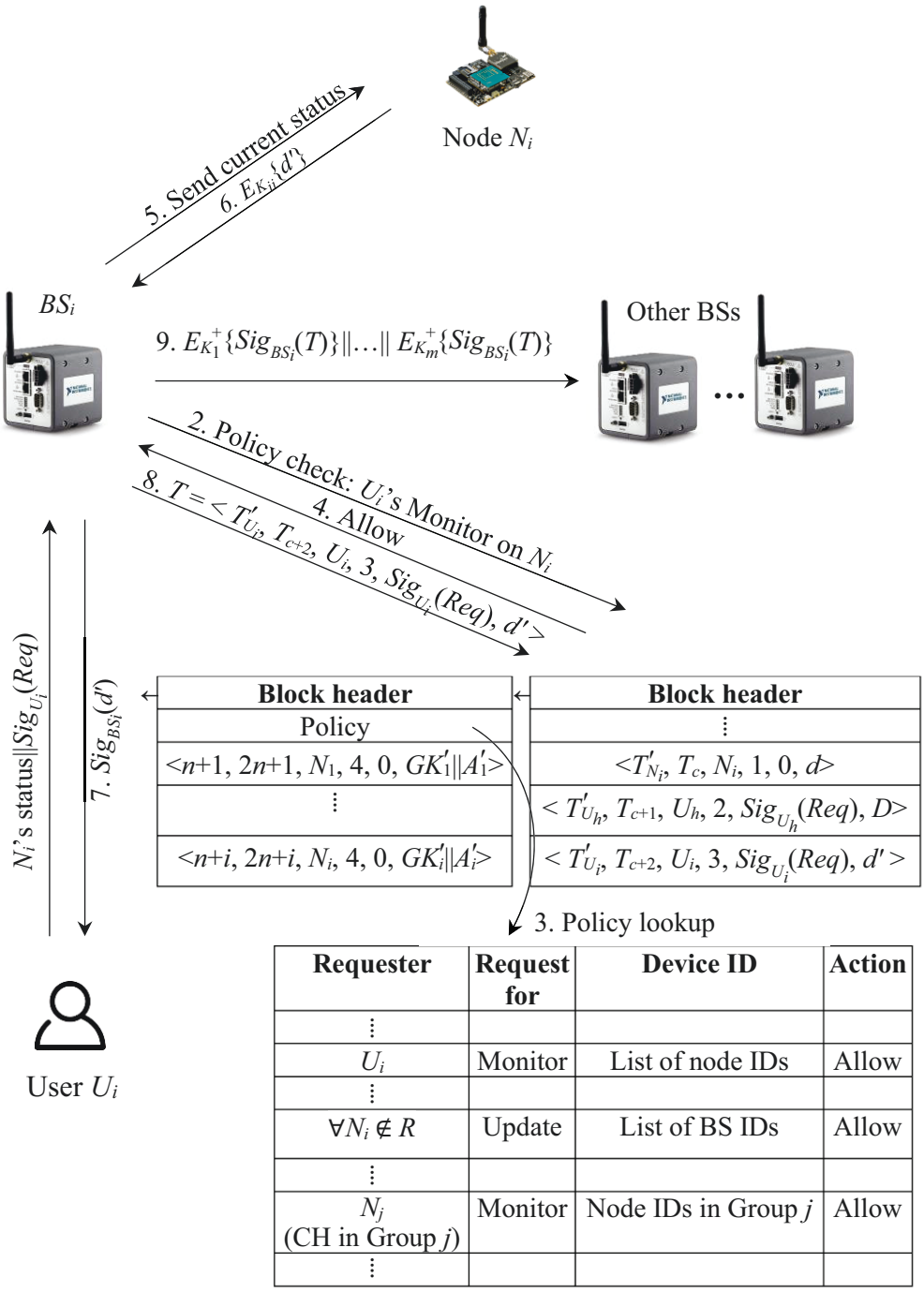


Figure 4.8: Monitor on node N_i by the user U_h.

to other BSs as $E_{K_1^+}\{Sig_{BS_i}(T)\} \parallel \ldots \parallel E_{K_m^+}\{Sig_{BS_i}(T)\}$. Here, T'_{U_i} is the previous transaction of U_i and T_{c+2} is the current transaction number.

4.3 Evolution of the BC

In the proposed scheme, BC evolves as the transactions are added to it. Fig. 4.9 demonstrates the evaluation of BC in the proposed scheme. It is assumed that BSs create g groups after initialization and deployment of n nodes. It is also assumed that each block can accommodate $n+1$ transactions and policy links. Further addition of transactions results in creation of now blocks in BC. This demonstration shows the content of BC after the following sequence of operations (a) initialization of nodes, (b) distribution of GKs among sensor nodes, (c) storing data to BC (d) data access by the user (e) monitor (f) key revocation and finally (g) key update.

During initialization (Fig. 4.9(a)), the BSs generate a transaction for each node to record the provided credentials. This consumes n cells in Block 1 and the final cell accommodates the policy link which dedicates u cells for u users and one cell to all nodes. The header of this block is kept 'null' as there is no previous block in BC. After deployment, BS forms groups and distributes group keys among the nodes. The number of group keys is same as the number of groups in the network. Nodes in the same group receives same group key and BS creates one transaction for each node. Assuming that there are g groups in the network, Fig. 4.9(b) shows the content of BC after the group key establishment. The block header contains the hash of the previous block and the transaction number starts from $n+1$ as Block 1 contained previous n transactions. Similar to Block 1, transactions for n nodes consumes up to n cells in Block 2 while group key establishment. Hence the last cell can only accommodate the policy which includes the previous rules in addition to the new rules for CHs and member nodes of each group. Assume that, node N_n has sent its data to BSs to get it stored in BC. As there is no cell available in Block 2, a new block Block 3 is created for the new transaction with the hash of previous block in its header (Fig. 4.9(c)). The sequence of this transaction is numbered as $2n+1$ as the last transaction of Block 2 was $2n$. Similarly, Fig. 4.9(d) shows the content of BC when user U_h wants to access the record of N_n. This transaction sits in Block 3 and is sequenced as $2n+2$. It keeps the data D which comprises only d in this scenario. Note that, to retrieve the record of N_n, BSs find the most recent transaction of N_n and gather all data from the linked transactions of type 1. Also assume that, U_h has sent another request to know the status of N_n. A new transaction with sequence number

$2n+3$ is created in Block 3 once the status d' is transmitted to U_h. This transaction includes $2n+2$ as its previous transaction and is shown in Fig. 4.9(e). Now assume that, BSs want to revoke N_n. For this purpose, BSs updates the policy and refreshes the credentials of rest of the nodes. Fig. 4.9(f) shows the content of BC while N_n revocation starts. The updated policy is linked in Block 3 and includes all rules from the policy as in Fig. 4.9(b) except that specifies eligible nodes' list to get updates. The new policy shows that only $N_1 \dots N_{n-1}$ are eligible to get the updates. After revocation, BSs must update other nodes, for which the content of BC is shown in Fig. 4.9(g). The first update transaction (type 4) is made for N_1 and accommodated in Block 3. Transactions for other $n-2$ nodes is accommodated in Block 4 which left 2 empty cell in the block after completing the operation.

4.4 Difference between the BC in Bitcoin and the BC employed in this proposal

A very successful application of BC is Bitcoin. Beside the Bitcoin network, different applications have implemented BC with different modifications to achieve their purposes. Before depicting the difference between the Bitcoin BC and the BC employed in this work, a concise description of Bitcoin's basic components is presented.

A Bitcoin Network consists of a set of components, for e.g., nodes, miners and so on, which follow the Bitcoin P2P protocol. A machine can play as a node by running an associated application. These nodes keep Bitcoin running by simply relaying the transactions throughout the network. Each node maintains a copy of the BC and can validate the transactions before forwarding them. Here, validation refers to checking the formation of the transactions against some rule and the balances are available to be spent. A machine can play a as miner by running a mining application. Upon receiving a transaction from the nodes, all miners compete among themselves to solve a PoW [27] algorithm. Only the first successful miner puts the transaction into the local BC and broadcasts to other nodes. Beside the nodes and miners, Bitcoin Wallet is also a major component. It is an application that enables users to view their Bitcoin holdings and send or receive Bitcoins. Table 4.3 refers to the comparison between the Bitcoin BC and the BC used in this work.

Table 4.3: Comparison between the Bitcoin BC and the BC employed in this proposal.

#	Parameters	BC in Bitcoin	BC in proposed protocol suite
1.	**BC Visibility**	Public	Only to BSs and users
2.	**Transaction chaining**	Input/output	Previous T of same node/user
3.	**Transaction mining**	All valid transactions are mined	All transactions
4.	**Mining requirement**	Proof of work	None
5.	**Double Spending**	Prohibited	Not applicable
6.	**Transaction verification**	Signature	Signature
7.	**Transaction dissemination**	Broadcast	Broadcast
8.	**Blocks stored by miner**	All blocks	All blocks
9.	**BC controller**	No one	BSs
10.	**Miner rewards**	Bitcoins	Nothing
11.	**Malicious miner**	Allowed to join	Not possible
12.	**Effects of 51%nodes**	Double spending	Network works as long as $\lceil \frac{m}{2} \rceil + 1$ BSs remain uncompromised
13.	**Encryption method**	Public/private keys	Symmetric keys

4.5 Summary

In this chapter, the major components of the proposed BC based protocol suite are described first, such as sensor nodes, CHs, BSs, BC, transaction, user, policy, etc. Then the protocol suite for WSNs is described which comprises initialization, GK establishment, revocation, key update, store, data access, and monitoring protocols. Evolution of the BC in the proposed scheme is presented thereafter. Finally, for better understanding, a comparison is shown between the Bitcoin BC and the BC used in this proposal.

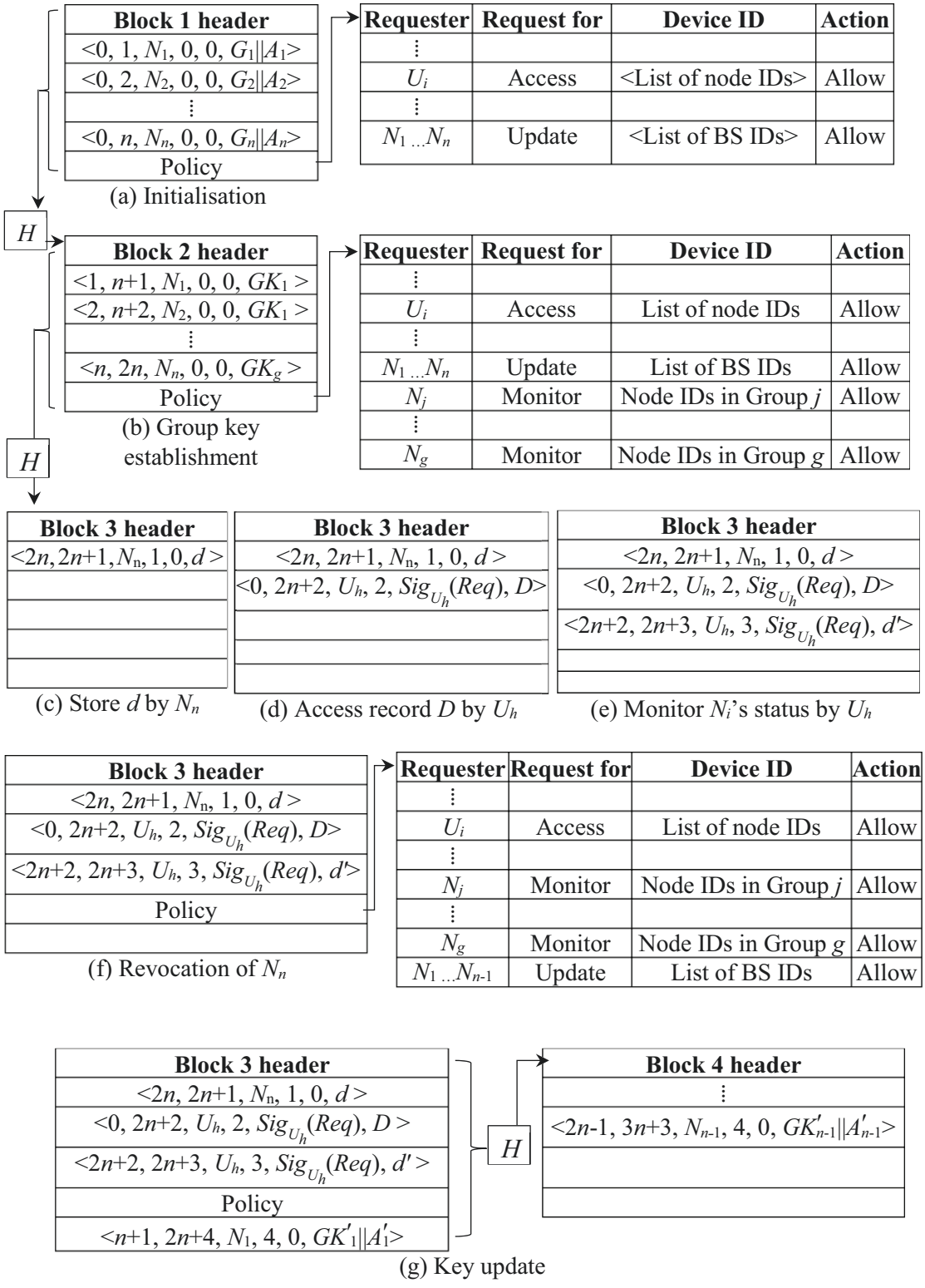


Figure 4.9: Evolution of the BC in the proposed scheme while- (a) initialization of nodes, (b) distribution of group keys among sensor nodes, (c) storing data to the BC (d) data access by the user (e) monitor (f) key revocation and (g) key update.

Chapter 5

The proposed energy balancing cognitive partitioning approach

This chapter describes a clustering approach for reducing the blind spot problem in WSNs. The proposed approach is divided into the cluster formation phase and the CHs selection phase in order. The CHs selection phase is further divided into the candidate selection and final CHs selection.

5.1 Network model

5.1.1 Theoretical representation

A WSN can be represented by the graph $\bar{G} = (V, \bar{E})$, where V is the set of all sensors in the network and $\bar{E} = \{(i, j) \subset V \mid \bar{d}_{i,j} \leq \bar{R}\}$ represents the wireless connection between nodes. Here, $\bar{d}_{i,j}$ is the distance between nodes N_i and N_j and $\bar{R}$ is the transmission range.

5.1.2 Graphical representation

In this work, sensor nodes are assumed to communicate with the BSs through CHs. A node senses data and forwards to its CH which relays the same data to the next CH. Thus, the data is received by the BSs. One CH is chosen for a cluster at a time from a set of candidate CHs. Fig. 5.1 illustrates the network model with BS_i, CHs, clusters, normal nodes and data flow from CHs to BS_i.

5.1.3 Assumptions

The assumptions made for describing the proposed approach are given bellow-

- Homogeneous sensor nodes with the same functionality and capacity are scattered uniformly within a rectangle area. The BSs are located at a distance from the monitoring area.

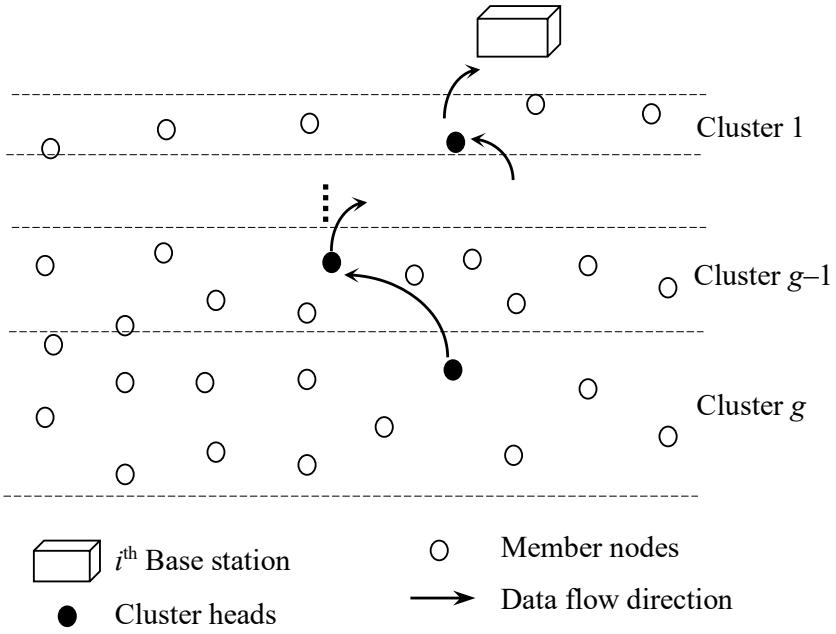


Figure 5.1: Architecture of a WSN with unequal clustering approach.

- Intra-cluster and inter-cluster communications are single hop and multi-hop data transfer respectively which are conducted by CHs, i.e., each CH has to send the traffic to the next CH towards BS_i.
- Only one CH is selected from each cluster in each round.
- As data aggregation is out of the scope of this work, it is assumed that each event is captured by the nearest sensor only and each event generates equal amount of data unit.

5.2 Details of the proposed approach

The proposed clustering approach can be divided broadly into 1) Energy balancing partition and 2) Repetitive operational rounds. After deployment, sensor nodes send their residual energy to BSs to facilitate the partition of the entire area P into g unequal sectors p_1, p_2, ..., p_g for balancing the energy during multi-hop data transmission. Here, p_1 is the closest sector to BSs and $p_1 < p_2 < \ldots < p_g$ in terms of size.

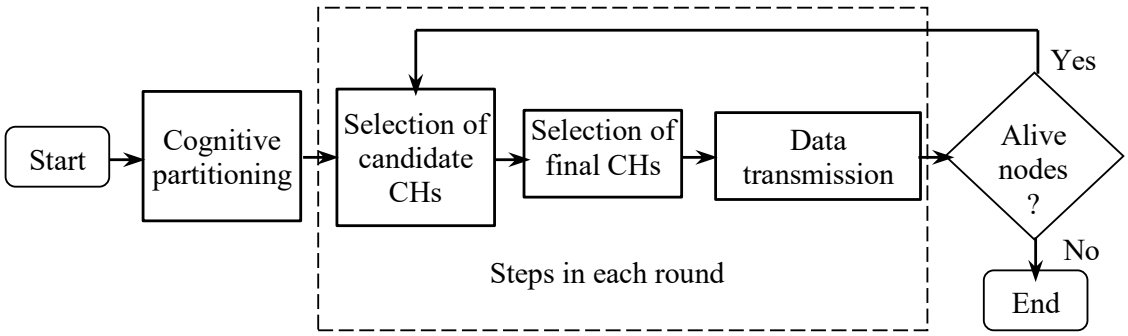


Figure 5.2: Components of the proposed clustering approach.

After the partition, operation of the entire network is divided into rounds. Each of these rounds again consists of three steps, namely, selection of candidate CHs, selection of final CHs from the candidates, and data transmission. Fig. 5.2 illustrates the steps associated with the proposed approach. Recent works have proved the effectiveness of cluster formation prior to the selection of CHs. Guiloufi et al [22] have used a Sierpinski triangle for this purpose which ensures smaller cluster size near to the BS. However, fractal based approaches fail to address any actual measurement of cluster size for balancing the energy consumption. The novelty of the proposed approach is that instead of forming clusters based on geometric fractals, it determines the actual size of p_i in a cognitive way for balancing the energy consumption.

5.2.1 Energy balancing cluster formation

As the first step towards the cluster formation, the BS divides the entire area equally into n partitions, namely P_1, P_2, ..., P_g where the separation line of any P_i and P_{i+1} is parallel to that edge of P which is closest to BSs. Now, assuming that each P_i is a single node and Z is the total number of events occurred in P within a given time frame, BS_i computes the energy loss at each P_i for forwarding the corresponding data to P_j using eq. 5.1. The probability of an event to occur in any partition P_i can be defined as P_i/P. Thus, the total number of events occurred on that area becomes $Z(P_i/P)$. Assuming one event generates one data unit, any partition P_i has to receive all the data from P_{i+1} and transfer to P_{i-1} after accumulating its own data. Hence P_i has to receive and transfer more data than P_{i+1}, i.e., energy loss in P_i is greater than

that in P_{i+1}. Therefore, the relation between energy losses in each partition becomes: $e_g < e_{n-1} < \ldots < e_1$. Eq. 5.2 shows the energy loss of P_g which, being the farthest partition, is not burdened with receiving any data from other partitions. BS_i now determines the percentage of area to adjust from P_i using eq. 5.3 and 5.4.

$$e_i = Z\left(\frac{\sum_{l=i-1}^{g} P_l}{P}\right) \cdot C_r + Z\left(\frac{\sum_{l=i}^{g} P_l}{P}\right) \cdot \bar{d}^{\mu}_{i,j} C_s \tag{5.1}$$

$$e_g = Z\left(\frac{P_g}{P}\right) \cdot \bar{d}^{\mu}_{g,g-1} C_s \tag{5.2}$$

$$\hat{d}_i = \left(\frac{1}{g} - \frac{e_i}{\sum e}\right) \times 100 \tag{5.3}$$

$$p_i = P_i + P_i \times \hat{d}_i \tag{5.4}$$

Here, $\hat{d}_i$ denotes the deviation of energy in P_i from the equidistributed energy in percentage. The proposed approach tries to minimize this deviation by adjusting the area of P_i by $\hat{d}_i\%$ in eq.5.4. The total energy loss of P_i within a given time frame can be factorized into three components such as the energy loss for receiving data from P_{i+1}, sending the same amount of data to P_{i-1} and sending the data of P_i's local events to P_{i-1}. As P_i has to receive the data from P_{i+1} and transfer to P_{i-1}, BS_i focuses on controlling P_i's local events to reduce its energy deviation. The more the number of local events in P_i, the more its energy loss. Again, the number of local events is proportional to the area because of the uniform distribution of nodes. Hence the adjusted area p_i is expected to have no or small deviation in energy from the equidistributed energy in percentage.

5.2.2 Selection of candidate CHs

BS_i selects CHs after the formation of g clusters in the network. For this, BS_i selects a set of candidate nodes S_i comprising all c_{i_j} in p_i such that $w(c_{i_j}) < \tau$, where $\tau = (Min_{w_i} + Min_{w_i} \times m)$. Here, Min_{w_i} is the minimum weight in p_i and m is a predefined value for determining τ. The weight function w is defined in eq. 5.5 for any node i.

$$w(i) = a_1 F_{i_1}^{\alpha} + a_2 F_{i_2}^{\beta} + \ldots + a_q F_{i_q}^{\gamma} \tag{5.5}$$

Here, a_1, a_2, ..., a_q are coefficients and F_{i_1}, F_{i_2}, ..., F_{i_q} are the associated factors with i, for e.g., residual energy, number of replaceable nodes [83], nodes degree, etc., and α, β, ..., γ are the orders of F_{i_1}, F_{i_2}, ..., F_{i_q}.

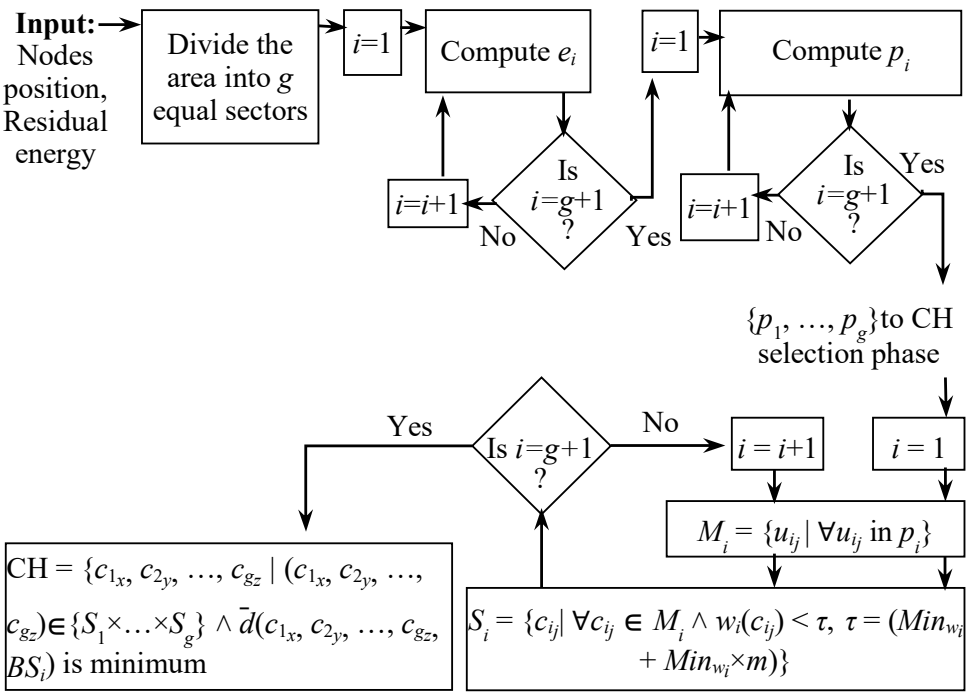


Figure 5.3: The cognitive partitioning and a round of CHs selection.

5.2.3 Selection of CHs

From $\{S_1, S_2, \ldots, S_g\}$, BS_i selects $\{c_{1_x}, c_{2_y}, \ldots, c_{g_z}\}$ as CHs for $p_1, p_2, \ldots, p_g$ such that $\bar{d}_{c_{1_x}, c_{2_y}, \ldots, c_{g_z}, BS_i}$ is minimum. Hence according to the rule of product, BS_i checks $\prod_{i=1}^{n} \mid S_i \mid$ values to find the CHs that yield the least distance. The steps associated in the proposed clustering approach is illustrated in Fig. 5.3.

5.3 Significance of the proposed CHs selection approach

A majority of the previous related works have considered individual distances to BS_i while selecting the CHs, whereas the proposed approach counts the total path to BS_i connecting all potential CHs. The significance of considering the total path can be understood with Fig. 5.4. If all other factors except the distance to BS_i were kept constant, the majority of existing clustering approaches would select $\{B_g, B_{g-1}, \ldots, B_1\}$ as CHs because of their short individual distances to BS_i. This may lead the network losing more power in a multi-hop communication due to long cumulative distance to BS_i through all CHs. The proposed approach eradicates the problem by

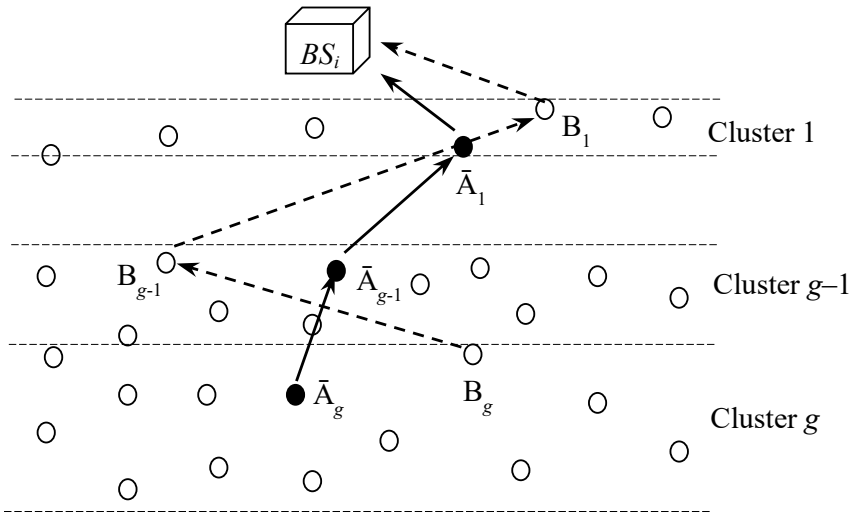


Figure 5.4: Significance of total path connecting all CHs instead of individual paths from CHs to BS_i.

selecting those candidate nodes as the CHs ($\{\bar{A}_g,\ \bar{A}_{g-1},\ \ldots,\ \bar{A}_1\}$ in Fig. 5.4) that yield the lowest cumulative distance to BS_i.

5.4 Summary

In this chapter a clustering approach is proposed that partitions the monitoring area in a cognitive way for energy balancing. In addition, the proposed approach adopts a two layered scrutinization for selecting cluster heads which ensures minimum energy consumption from the network. Consequently, it reduces the blind spot problem that escalates once the nodes start dying. To present the clustering scheme, theoretical and graphical representation of the network is shown first. Then, the proposed approach is presented which comprises multiple steps such as cognitive partitioning, candidate CH selection, and actual CH selection at the end.

Chapter 6

Analysis of the proposals

6.1 Security analysis

A protocol must satisfy three basic requirements to be robust in terms of security such as confidentiality, integrity, and authentication. The underlying mechanism [33] used in the proposal here successfully satisfies all of these requirements. Moreover, it can prohibit different attacks such as spoofed or replay attack, de-synchronization, Denial-of-Service (DoS), physical node capture, blackhole and wormhole attack, etc. However, the proposed work, along with these attacks, is rigid against other attacks that were irresistible with [33] only. Following is the collection of some of these attacks.

6.1.1 BS compromise

In [33], the network has only one BS that prepares and distributes all keying materials. Moreover, data gathered by the sensor nodes are also collected by the same BS. Hence, it is quite feasible for an adversary to compromise the entire network by simply taking over the BS. The proposal in this work includes multiple cooperative BSs to avoid such an attack. Here, other BSs can continue the operation if any of them is compromised. An adversary has to take control over more than half of the BSs to make the network compromised. Again, the BSs are not resource constrained and can apply robust security mechanisms. Hence, compromising the half of the BSs would be infeasible for the adversary in terms of time and computational resources.

6.1.2 Data tampering at BSs

This attack refers to altering any data after it is received and stored by the BS. The proposal in this work not only secures data on the way but also secures the stored data at the BSs with BC. As each data is linked to the previous one, tempering a data

requires all of the successive data to be tempered in all BSs, which is quite impossible without being undetected. Thus, BC introduces data immutability to this proposed work.

6.1.3 Malicious activities of the BSs

It refers to providing a user with illegitimate data, network with incorrect keying material, etc by the BSs. In single BS systems, such activities are difficult to detect, and the network has to rely on the BS blindly. The proposed work here deploys multiple BSs to eradicate such issues. Here, each BS maintains an independent replica of the BC that makes it difficult for a BS to provide a user with falsified data without being detected by other BSs. Similarly, providing the sensor nodes with incorrect keying material is also infeasible.

6.1.4 Illegitimate access

This proposal maintains policies along with the transactions that facilitate the access control in the network. Any change in the policy must be notified to other BSs. Thus, policies do not remain in any central BS and all BSs can deploy the same policy network-wide.

6.2 Performance analysis of the proposed protocol suite

Performance of the proposed protocol suite is measured in terms of nodes' memory overhead, communication overhead, and nodes' computational overhead.

Parameter setting: The simulation is executed for different sizes of q, such as 64, 128 and 256 bits. The underlying mechanism [33] of the proposed scheme is λ secure, i.e., an adversary needs to make at least λ+1 nodes compromised to take over the network. In other words, the network can continue its operation with λ compromised nodes at most. For this simulation, the security parameter λ is assumed to be 100 while the total number of nodes is 300 and the number of BSs is 5. Each node is assigned with a 16 bit ID. A keyed hash function is used as the Message Authentication Code (MAC) that generates a fixed 64 bit output. 'Skipjack' algorithm is used for the encryption and decryption as it is used by TinyOS [84] and produces a fixed-sized

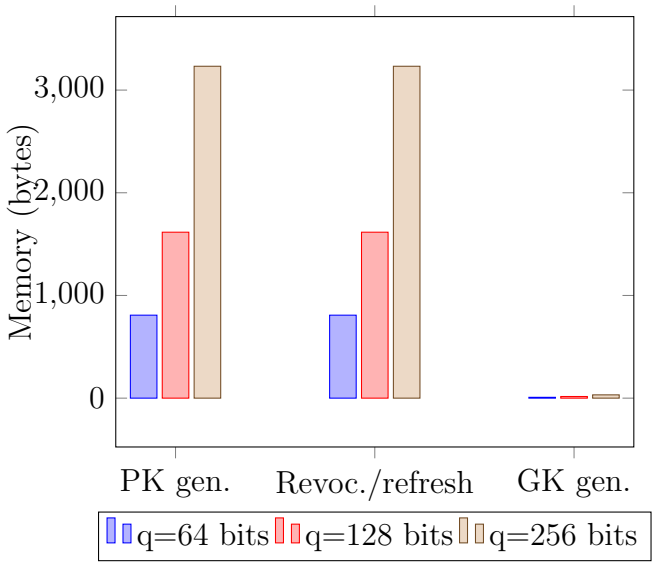


Figure 6.1: Memory overhead of a node in this proposal.

output.

6.2.1 Memory overhead

This work considers that sensor nodes in the network are resource constrained, whereas BSs and user devices are not. Hence, this experiment includes operations that require sensor nodes to store data. Thus, results from this experiment will help in feasibility analysis in the later part. Fig. 6.1 shows the simulation results in terms of memory overhead per node for the pairwise key generation (*PK gen.*), key revocation/refresh (*Revoc./refresh*) and group key generation (*GK gen.*) process. It shows that the size of q has an equivalent effect on memory overhead and the overhead remains the same for *PK gen.*, *Revoc./refresh*, and *GK gen.* Each of these operations, except *GK gen.*, consumed 808, 1616, and 3232 bytes of memory from a node while q is increased from 64 to 128 bits and then to 256 bits sequentially. For *GK gen.*, the memory requirement is the same as the size of q.

6.2.2 Communication overhead

Analysing the communication overhead is important as it is one of the factors that determine the performance of a protocol. Protocols having lower communication

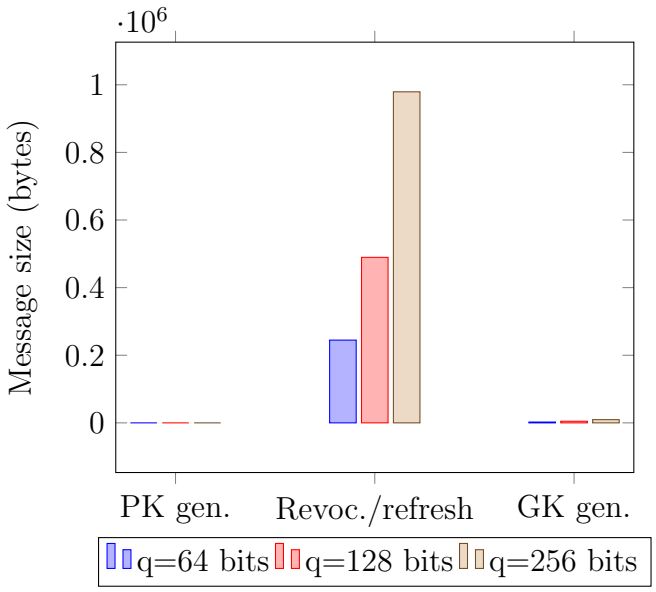


Figure 6.2: Communication overhead of this proposal.

overheads are considered to be more efficient. This experiment analyze the communication overheads of the crucial operations in the network, such as *PK gen.*, *GK gen.* and *Revoc./refresh*. Hence, data access or monitor operations are skipped from this experiment. Here, *PK gen.* and *GK gen.* operations are crucial because large communication overhead may incur high energy loss and long delay respectively to obtain secret credentials. Similarly, in *Revoc./refresh*, an adversary may get enough time to make the network compromised if key refreshment gets delayed due to a large communication overhead. On the other hand, communication overheads while accessing or monitoring the data affect only the Quality of Service (QoS) and do not introduce any vulnerability. Fig. 6.2 shows the communication overhead of the proposed scheme. There is no communication overhead for the nodes in *PK gen*, whereas in *Revoc./refresh*, the communication overheads are 244800, 489600, and 979200 bytes in terms of message size when q = 64, 128, and 256 bits respectively. In *GK gen.*, the overheads are 2400, 4800, and 9600 bytes in terms of message size for the same q values.

6.2.3 Computation overhead

Similar to the communication and memory overhead, computation overhead is also measured for the operations that require significant computation from the sensor

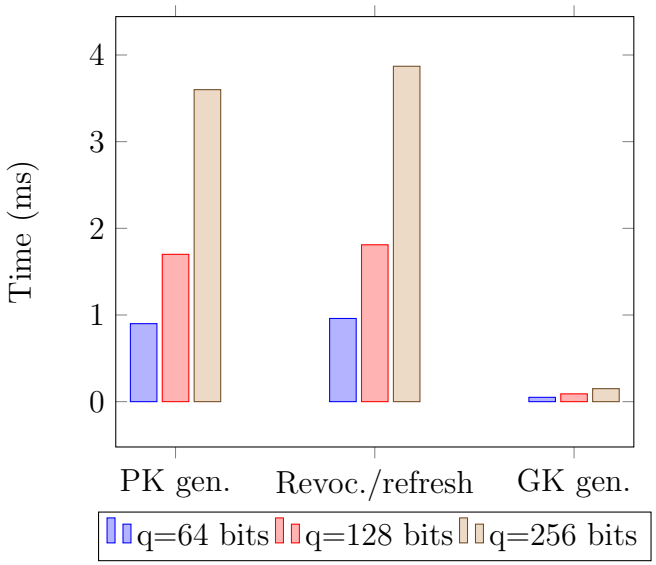


Figure 6.3: Computation overhead of a node in this proposal.

nodes. In *PK gen.*, a node performs λ degree polynomial operation, i.e., addition and multiplication of λ number of q bit data to get the pairwise key. Again, in *Revoc./refresh*, a node has to perform other related operations beside of the λ degree polynomial. Similarly, in *GK gen.*, a node performs all the operations other than the λ degree polynomial. Hence, the computational overheads of these operations are significant for analysing the feasibility of the proposed protocol suite in WSN motes. Fig. 6.3 shows the result of this experiment where each measurement is taken from the node's perspective. It shows that a node takes 0.9, 1.7 and 3.6 milliseconds (ms) for q = 64, 128, and 256 bits respectively while generating a pairwise key. After receiving the broadcast from the BSs, it requires 0.96, 1.81, and 3.78 ms to verify the message and update the keying material. Finally, a node expends 0.05, 0.09, and 0.15 ms to get its group key after receiving the broadcast from the BSs.

6.3 Performance analysis of the proposed clustering approach

6.3.1 Parameters and energy consumption model

The proposed clustering approach has been simulated with MATLAB. To evaluate the proposed approach, energy consumption model (Eq. 6.1) of any node N_i is the same as the approach given in [22].

Table 6.1: Simulation parameters.

Parameters	**Values**
Total number of nodes (n)	300
Size of the monitoring area (P)	200 m × 200 m
Weight threshold (m)	0.02
Energy dissipation to run the receiver circuitry ($\bar{e}_{elec}$)	50×10^{-9}
Energy consumed by transmitter power amplifier ($\bar{e}_{amp}$)	10×10^{-12}
Length (bits) of data to send (k)	4000
Initial energy ($\bar{e}_{init}$)	1 J
BS_i coordinate	(100, -50)
Number of clusters (g)	4
Path loss exponent (μ)	2
Round interval	Iteration of one loop

$$\bar{e}_c(N_i) = \bar{e}_{RX}(k) + \bar{e}_{TX}(k, \bar{d}_{i,j}) \tag{6.1}$$

$$\bar{e}_{RX}(k) = \bar{e}_{elec} \times k$$
$$\bar{e}_{TX}(k, \bar{d}_{i,j}) = \bar{e}_{amp} \times k \times \bar{d}_{i,j}^{\mu}$$

$$w(i) = \frac{\bar{e}_c(N_i)}{\bar{e}_{init}(N_i)} \tag{6.2}$$

Here, $\bar{e}_{elec}$ is the energy dissipation to run the receiver circuitry for k bits of data and $\bar{e}_{amp}$ is the energy consumption by the transmitter power amplifier to send the same bits of data over a distance $\bar{d}_{i,j}$. Here, $\bar{d}_{i,j}$ is the distance between nodes N_i and N_j. The weight function w considers $\bar{e}_c(N_i)$ and $\bar{e}_{init}(N_i)$, i.e., the consumed energy and the initial energy of N_i, which is given in Eq. 6.2. The list of simulation parameters and the cluster properties generated with the proposed approach are shown in Table 6.1 and Table 6.2, respectively.

6.3.2 Network lifetime

Fig. 6.4 shows the number of alive nodes per round with the proposed approach and with the approach presented in [22]. From the figure it is seen that, nodes start

Table 6.2: Size of the clusters.

Clusters	**Length (m)**	**Width (m)**	**Intermediate clusters to BS_i in order**	**Number of member nodes**
p_1	200	41.07	-	62
p_2		46.78	p_1	71
p_3		52.05	p_2,p_1	79
p_4		59.65	p_3,p_2,p_1	88

Table 6.3: Comparison between the proposed approach and the approach presented in [22] in terms of FND, HND, and LND.

Clustering approach	**FND**	**HND**	**LND**
The proposed approach	2971	3450	3550
The approach presented in [22]	1400	2800	3600
Efficiency of the proposed approach: Average efficiency gain: 71%	52%	21%	−2%

dying at about Round 1400 in [22]. At Round 2000, the network loses about 8% of its sensor nodes. Again, 50% of the sensor nodes fall into dead state at Round 2800 which raises to 90% at Round 3500. The proposed approach, on the other hand, shows a more steeper curve in terms of alive nodes per round. However, nodes in the proposed approach survive more rounds than the proposal in [22]. The proposed approach keeps the nodes alive upto the Round 2971. Unlike the approach of [22], it loses 8% of the nodes at Round 3080 and 50% at Round 3450. Soon after the death of 50% of the nodes the network survive only for few more rounds which ends with the last node death at Round 3550. Table 6.3 represents a comparison between the proposed approach and the approach presented in [22] in terms of First Node Dead (FND), Half Node Dead (HND), and Last Node Dead (LND). The proposed approach is more efficient in terms of FND and HND. Although it falls behind the approach presenetd in [22] for LND, it successfully minimizes blind spot problem by minimizing the duration between FND and LND.

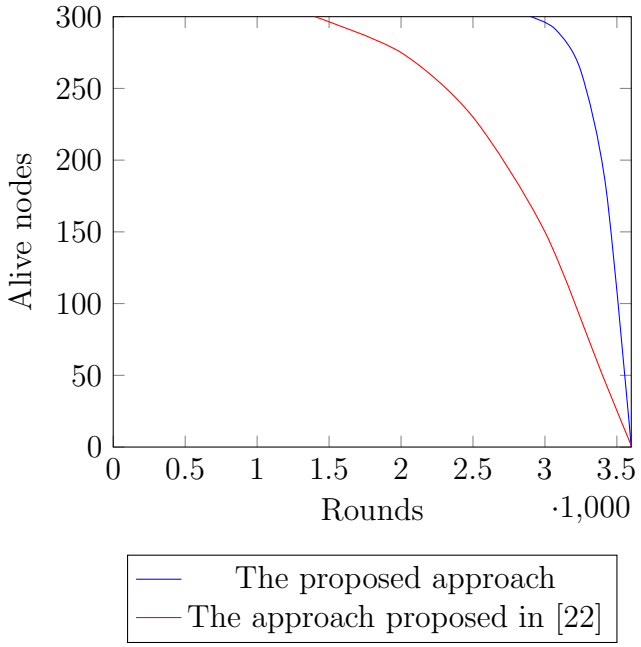
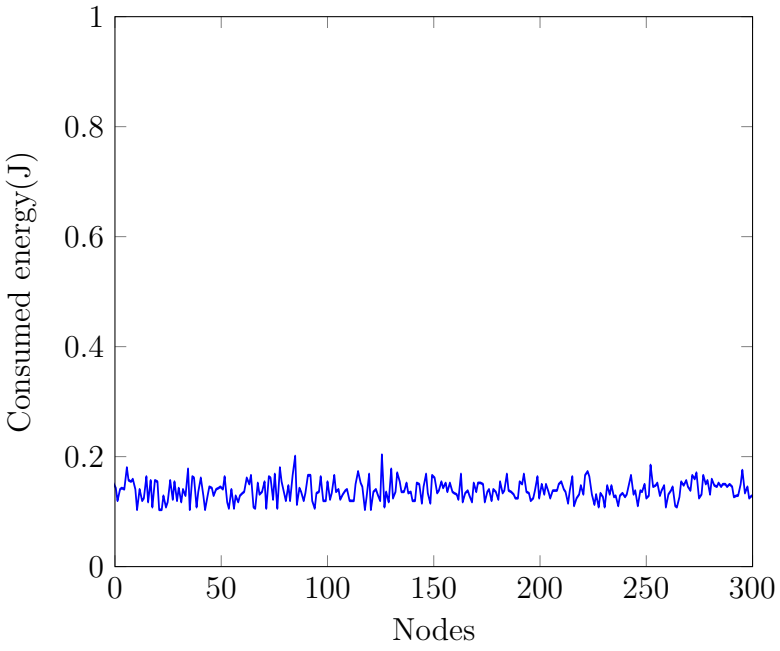


Figure 6.4: Proposed approach Vs. the approach presented in [22] in terms of number of alive nodes per round.

6.3.3 Balanced Energy consumption

Fig. 6.3 represents the consumed energy of each node till Rounds 500, 1700, and 2970 with the proposed approach. From the figure it is seen that the values are equivalent for all nodes in any particular round and they fluctuate ± 0.05J from the average value of that round. At Round 1700, the values remain within the range of 0.5J and 0.6J, i.e., about 55% of the initial energy is consumed until this round. At Round 2970, the values fluctuate between 0.89J to 1J; hence, from this round nodes starts falling into the dead state. Table 6.4 summarizes the energy consumption of the approach proposed in [22] till Rounds 250, 700, and 1500. This approach consumes 55% of the nodes' initial energy at Round 700. The energy consumption varies ± 0.07J from the average value in early rounds and raises up to ± 0.13J in later rounds. The result justifies that the proposed approach is about 50% more efficient in terms of balancing the residual energy after the completion of a round. Fig. 6.4 shows average energy consumption in each cluster at Rounds 500, 1700, and 2970 with the proposed approach. From the figure it is seen that the average energy consumption in any cluster varies between 3×10^{-4}J and 3.5×10^{-4}J. Hence, this figure also indicates that the proposed approach maintains an equivalent residual energy in all clusters



(a) Till Round 500.

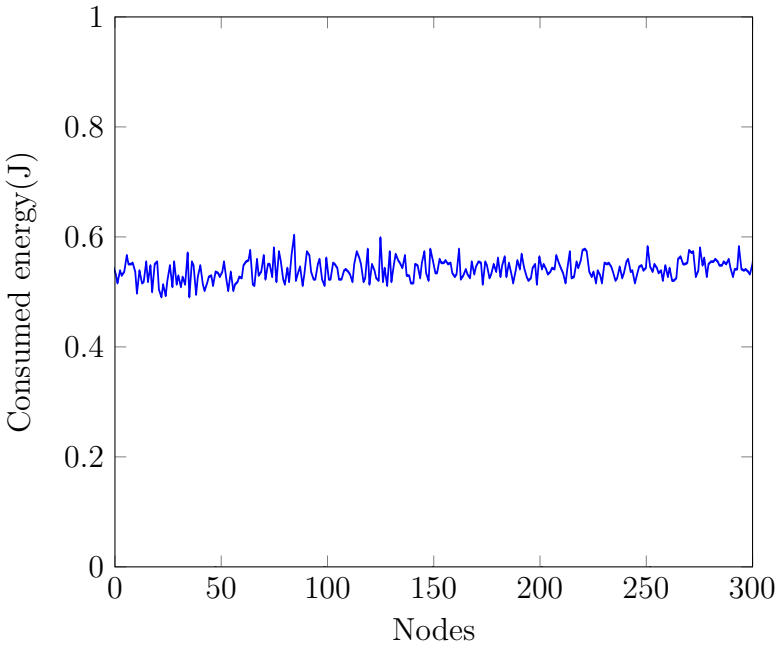
Table 6.4: Nodes energy consumption in [22] .

Rounds	**Min. energy consumption (J)**	**Max. energy consumption (J)**	**Average (J)**
250	0.01	0.15	0.06
700	0.35	0.52	0.43
1550	0.74	1.00	0.87

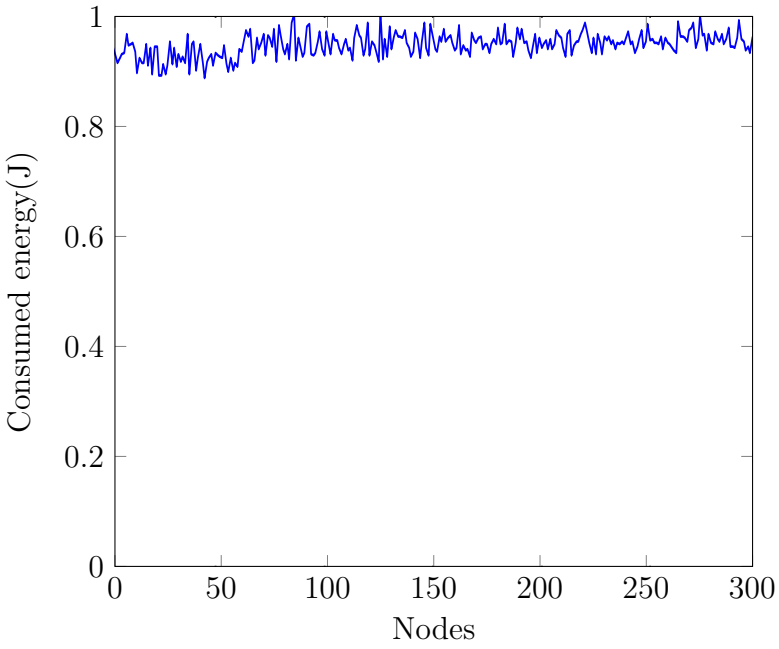
throughout the lifetime of the network.

6.3.4 Distribution of dead and alive nodes

Distribution of dead and alive nodes is also an important aspect to consider while evaluating the performance of a clustering approach. Beside of reducing the duration of the declining state, the clustering approach should ensure uniform occurrence of node's death throughout the network during this state. This maintains the consistency between the alive nodes and the probability of capturing an event during the declining state. Fig. 6.3 shows dead and alive nodes distribution during the declining state when 10%, 50%, and 90% nodes are dead in the network. From the figure it is inferred that the proposed approach maintains a uniform distribution of alive nodes during the declining state.



(b) Till Round 1700.



(c) Till Round 2970.

Figure 6.3: Energy consumption till different Rounds of the proposed approach.

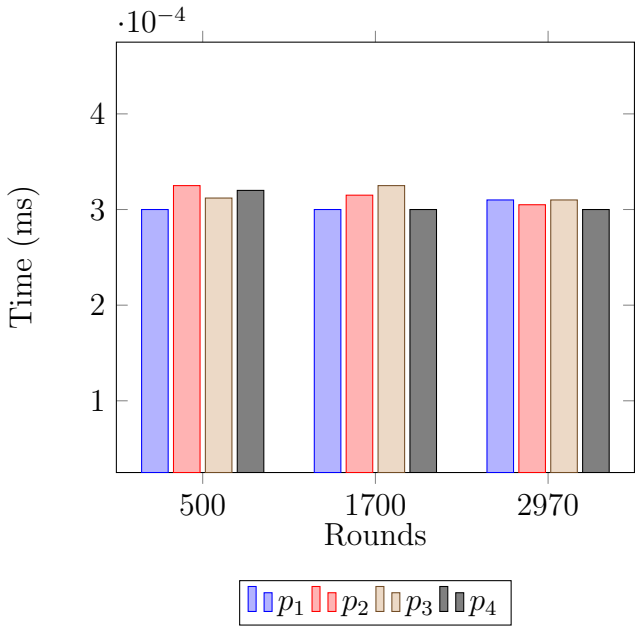
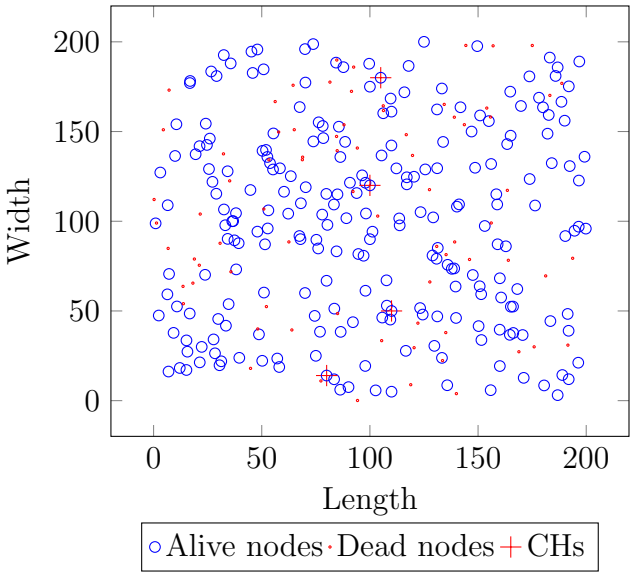


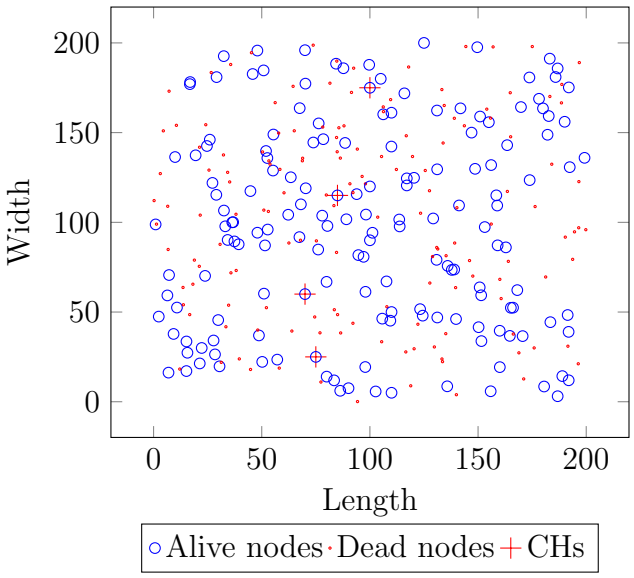
Figure 6.4: Average energy consumption in each cluster at Rounds 500, 1700 and 2970.

6.3.5 Effect of the number of clusters

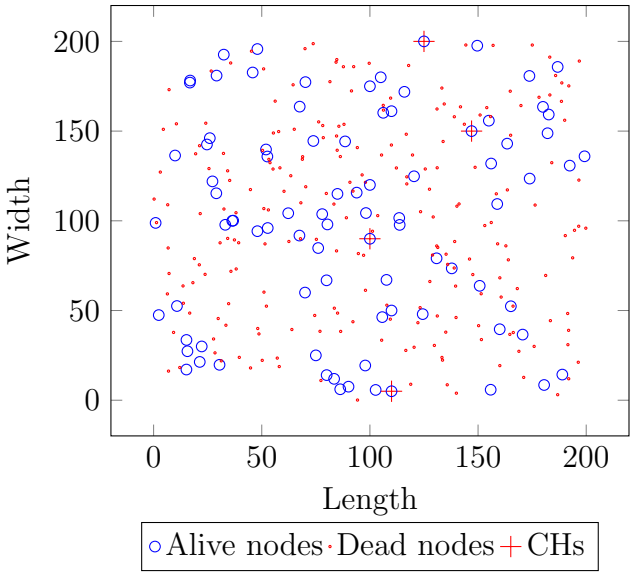
The number of clusters affects the performance of the proposed clustering approach as shown in. Fig. 6.4. The figure shows that the increased number of clusters produces more steeper curves, i.e., the duration of the declining state decreases. From the figure it is seen that the duration of declining state is about 500 rounds with 4 clusters, whereas it is only about 400 rounds for 8 clusters. Finally, the round gap between FND and LND decreases to less than 300 with 16 clusters in the network. Although a higher number of cluster decreases the blind spot problem by reducing the declining stare, it incurs a long delay while the selection of CHs by the BSs. With the proposed approach, BSs select candidate CHs from each cluster and search for a combination of them that yields the lowest path cost. Hence, BSs have to check more combinations for an increased number of clusters. Although the number of candidate CHs is determined by the weight function, experiment shows that the number of clusters exponentially affects the time to complete a given number of rounds. Fig. 6.5 shows the consumed time to complete 3500 rounds with 4, 8, 12, and 16 clusters in the network. From the figure it is shown that it requires about 100s to complete 3500 rounds with 4 cluster. This requirement increases with the number of the clusters



(a) While 10% of the nodes are dead.



(b) While 50% of the nodes are dead.



(c) While 90% of the nodes are dead.

Figure 6.3: The distribution of dead and alive nodes in the network with the proposed approach.

and reaches to 14456s for 16 clusters.

6.3.6 Effect of the mobility of nodes

The mobility of the nodes introduces new challenges while designing a protocol as its topology changes time to time. The clustering approach presented in this thesis assumes that the nodes are uniformly distributed throughout the monitoring area and forms equal clusters in the first step. Then the area of each cluster is adjusted to equalize the energy consumption. Here, the energy consumption of a cluster depends on the number of its member nodes. If the nodes were mobile few of the nodes might move to the other clusters right after the cluster formation and cause an imbalance in the consumption of energy. However, if the nodes go mobile after the cluster formation and do not enter into other clusters then the consumed energy of the clusters remain equivalent in each round. Hence this proposal support the mobility of a node only after the cluster formation process and inside of its cluster.

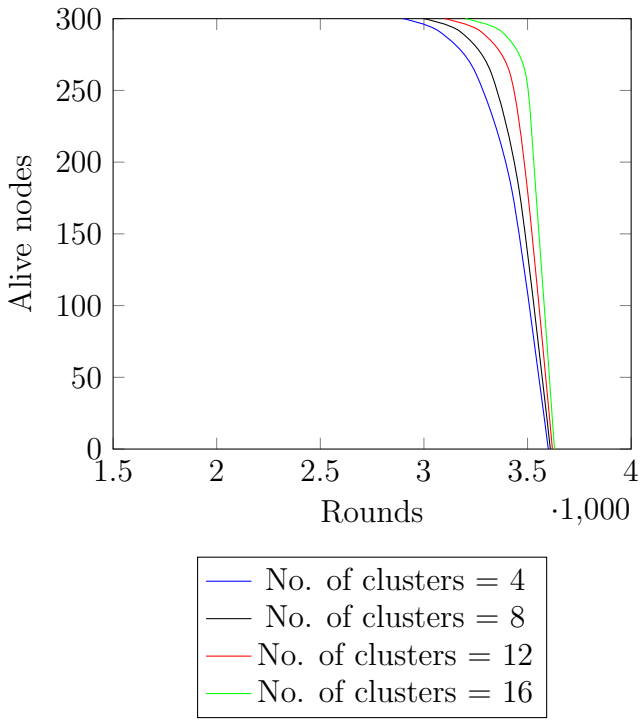


Figure 6.4: Effect of the number of clusters.

6.4 Feasibility analysis

6.4.1 Specifications of TelosB mote

In this section, the overheads of the proposed protocols are analyzed for a real sensor node − the TelosB mote. TelosB is a research-oriented mote developed by UC Berkeley. It is equipped with an MSP430 micro-controller that incorporates a 16 bit RISC CPU of 8 MHz, 48K bytes flash memory (ROM), and 10K bytes of RAM. The RF transceiver on TelosB is IEEE 802.15.4/ZigBee compliant and can have upto 250 kbps data rate.

6.4.2 TelosB motes with the proposed protocol suite

From the simulation it is seen that, *PK gen.* and *Revoc./refresh* requires same memory that is greater than the requirement in *GK gen.* Whereas in terms of communication and computational overhead, *Revoc./refresh* requires more resources. Hence, the selected mote should accommodate the operations of *Revoc./refresh* phase to prove the feasibility of this proposal as the BSs and users are not resource constrained.

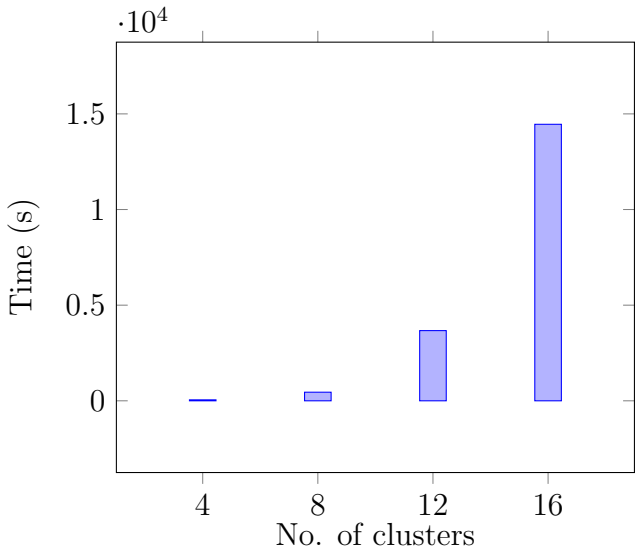


Figure 6.5: Consumed time to complete 3500 rounds with different number of clusters.

Here, a node has to store $(\lambda{+}1){\cdot}q$ and $(\lambda{+}1){\cdot}ID$ bits of keying material and a q bit group key, i.e., $(\lambda{+}1){\cdot}(q{+}ID){+}q$ bits in total. Fig. 6.6 shows the total storage a node requires to hold these credentials for different values of λ and q while $ID =$ 16 bits. It can be seen from the figure that the mote can accommodate almost all combination of λ and q. Hence, the proposed protocol suite is certainly applicable if λ and q are chosen wisely. Selection of values for λ and q are described in discussion section below for further clarification.

6.5 Discussion

6.5.1 The proposed protocol suite

The memory overhead of a node, as shown in fig. 6.1, is 808 bytes while the value of q is 64 bits. Here, it is assumed that the network is λ secure; hence, each node must accommodate $\lambda{+}1$ number of preloaded secrets. As the value of λ is assumed to be 100, a node in the simulation saves 101 secrets each of which are of 64 bits, i.e., 808 bytes in total. Similarly, when the key size is increased to 128 and 256 bits, memory consumption also increases to 1616 and 3232 bytes respectively. On the other hand, a node maintains only one key to communicate with its group. Hence, the memory requirement is same as the key size in *GK gen.*

In this proposal, there is no communication overhead for *PK gen.* as nodes do

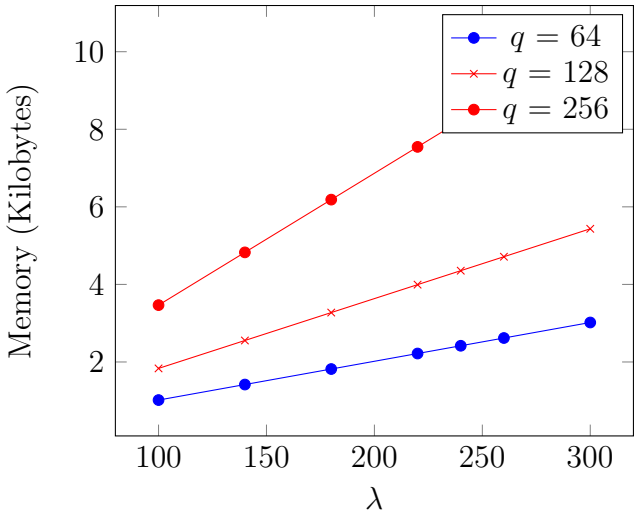


Figure 6.6: Mote's memory consumption on different values of λ and q.

not need to exchange any data to generate secret shared keys. Instead, a node can calculate the key knowing only the ID of another node. In *Revoc./refresh*, a BS broadcasts one message of size around 244800 bytes to the network to revoke one node. Note that, this broadcast takes the form $B = B \parallel E_{K_{ii}}\{A_i + A'_i \parallel GK_i\}$ and sends λ+1 number of new secrets along with a GK to each node. Here, for the revoked nodes $A'_i = A_i$ and GK_i = 'null'; hence, they do not receive any update. As the value of λ is considered to be 100, the size of the broadcast becomes (101+1)×300×8 bytes for q = 64 bits. Similarly, the size of the broadcast increases with the increase in the size of q. In *GK gen.*, a BS broadcasts only GKs for each node; hence, the size of the message becomes 300×8 bytes. In this proposal, GK update is also suggested along with A' whenever the BSs attempts to revoke a node. Hence, *Revoc./refresh* includes both A' and GK'. In [33], GK' distribution is shown separately and not included in the node revocation experiment. However, it did not deny the importance of updating all GKs while revocating a node to ensure robustness. Hence, communication overhead of this proposal becomes same as [33] if GK' distribution is not considered. The proposed protocol suite also involves the broadcasts of transactions among the BSs. In this experiment these broadcasts are made after serving the nodes and users to avoid any delay. Hence, communication overhead related to these broadcasts are not considered.

In the experiment on computational overhead, a node has to compute a λ degree

polynomial to generate a pairwise key in *PK gen.* which costs 0.9 ms for $q = 64$ bits. Whereas in *Revoc./refresh*, a node takes 0.96 ms for the same size of q. This time is measured from the receiving of BSs' broadcast to the calculation of new keys. Within this time, a node performs MAC verification, decryption, subtraction, and the calculation of λ degree polynomial. Hence it consumes more time than *PK gen.* that involves a λ degree polynomial operation. *Revoc./refresh* provides a node with both group and pairwise keying materials in a broadcast. Whereas in *GK gen.*, nodes receive only the group keys. Hence, a node has to perform all the operations of *Revoc./refresh* but the λ degree polynomial in *GK gen.*. Thus, it costs 0.05 ms to complete the operation. For $q = 128$ and 256 bits, computational overheads increase accordingly.

Table 6.5 shows a summary of the overheads from this experiment. It is seen that the proposal is maintaining the performance as same as [33] in spite of applying the BC. It applies the BC in the BS level that allows WSN to have the benefits of Blockchain and at the same time frees nodes from involving into resource consuming operations. Moreover, the TelosB mote can accommodate almost all combination of λ and q (Fig. 6.6) as the most memory consuming operation, the evaluation of λ degree polynomial, hardy requires more than 10K bytes of memory. Other operations such as encryption, decryption, and hash maintain the size of the output same as the input. Besides, the operating systems for sensor nodes in WSNs (e.g., TinyOS 1.x [84]) are mostly fewer than 500 bytes. The higher values of either λ or q increase the security of the network. However, for a desired security level, it is better to lower the value of q and raise the value of λ. With a higher q value, the processor requires more clock pulses for an operation and consumes more energy from the sensor nodes; thus, shortens the network lifetime. Hence raising the value of λ can increase the security without affecting the network lifetime in this case.

6.5.2 The energy balancing cognitive partitioning approach

In terms of alive nodes per round (Fig. 6.4), the proposed approach shows a steeper curve than the approach proposed in [22]. However, nodes with the proposed approach survive more rounds than the approach in [22]. This is because the proposed approach guarantees the shortest path connecting each CH to BS_i. Thus, nodes lose less energy

Table 6.5: Summary of the overheads.

Phases	**Node's memory**	**Communication**	**Computation**
PK gen.	$(\lambda+1)\cdot q$	0	Evaluation of λ-degree polynomial
Revoc./refresh	$(\lambda+1)\cdot q$	1 broadcast by the BSs, size: $N\cdot(\lambda+1)\cdot q$	1H, 1 Decr., Evaluation of λ-degree polynomial
GK gen.	q	1 broadcast by the BSs, size: $GS \cdot q$	$1H$, 1 Decr.

GS: Group size, H: Hash, Decr: Decryption

while serving as CHs. Again, the cognitive partitioning of the monitoring area enables each cluster to maintain a member size that ensures equivalent energy consumptions in them. Also, from the Table 6.3 it is seen that the average efficiency of the proposed approach is more than the proposal in [22] in this regard.

From Fig. 6.3 and Table 6.4, it is seen that the proposed approach is almost 50% more efficient in terms of stabilizing nodes residual energy. Furthermore, Fig. 6.4 shows small differences in average energy consumption in each cluster at different rounds which indicates a balanced energy consumption among the clusters also. Hence, nodes are left with small and equivalent energy while they enter into the declining state. With the small remaining energy network hardly runs for few rounds, i.e., the duration of the declining state becomes small which, in turn, reduces the blind spot problem in the network

6.6 Summary

In this chapter, the security analysis is presented for the BC based protocol suite first. Then performance analysis is made in terms of memory, communication, and computation overhead. For the proposed clustering approach, the performance analysis is made in terms of network lifetime, energy consumption, and alive node distribution. Then, the feasibility analysis shows that the BC based approach can be deployed with the real sensor motes available today. Finally, the discussion part explains the simulation results.

Chapter 7

Conclusion

7.1 Work summary

The work of this thesis can be divided broadly into two parts−

Firstly, a new protocol suite is presented for WSNs with Blockchain technology. This proposal tries to address the following issues: how to achieve service availability while some of the BSs are compromised and how to make data immutable at the BSs. Unlike single BS systems, this work employs multiple cooperative BSs to provide service availability albeit some of the BSs are compromised. Each BS in this work holds an independent BC to provide immutability of the data. Multiple instances of the same BC allows BSs to verify the trustworthiness of any BS duiring its operations. This work also achieves network transparency by allowing user access to the network status. The proposed approach of applying the BC into WSNs does not consume additional resources from the sensor nodes as shown in the previous chapter. Finally, it is also shown that the proposal is feasible to deploy with currently available sensor motes.

Secondly, a new unequal clustering approach is proposed that minimizes the blind spot problem while prolonging the network lifetime. This approach promises equivalent and least energy consumption from the clusters in each round. For this purpose, the clusters are formed by dividing the monitoring area into multiple partitions in a cognitive way. Such a partitioning approach ensures the consumption of energy to be equivalent in each cluster. Furthermore, the proposed approach adopts two layered scrutinization while selecting the CHs. This ensures the least energy consumption from the network and prolongs the steady state. In addition, nodes are left with small energy before entering into the declining state as they maintain equivalent energy consumption throughout the steady state. As a consequence, the declining state becomes short which reduces the blind spot problem in WSNs.

7.2 Future research directions

This proposal lays the groundwork for future research in this area. With the proposed protocol suite, a network can continue its operation albeit some of the BSs are compromised. However, an adversary can retrieve all the previous data from the compromised BSs. Deploying encryption mechanisms that require multi-party decryption would be a good future research issue in this regard. Moreover, the BC in a BS grows with time as the sensor nodes start sending data. Hence an efficient memory management for the BSs needs to be addressed. Connecting the BSs with Cloud would also be a good future research issue. Moreover, an appropriate consensus algorithm would allow any BS to join the network and add more robustness to these protocols.

Furthermore, the future work for extending the proposed approach would be the cognitive partitioning of the network to support scalability. Considering different node matrices for the selection of candidate CHs, finding an efficient optimization algorithm for the selection of final CHs, and cognitive grid partition of WSNs would also be some good future research issues.

Bibliography

[1] J. M. Batalla, A. Vasilakos, and M. Gajewski, "Secure smart homes: opportunities and challenges," *ACM Computing Surveys (CSUR)*, vol. 50, no. 5, pp. 75–82, 2017.

[2] A. Alaiad and L. Zhou, "Patients' adoption of wsn-based smart home healthcare systems: An integrated model of facilitators and barriers," *IEEE Transactions on Professional Communication*, vol. 60, no. 1, pp. 4–23, 2017.

[3] P. S. Sandra, C. M. Sandeep, V. Nair, M. V. Vindhuja, S. S. Nair, and M. P. Raja, "Wsn based industrial parameter monitoring using smartwatch," in *2017 International Conference on Circuit ,Power and Computing Technologies (ICCPCT)*, April 2017, pp. 1–6.

[4] A. Kumar, K. Ovsthus, and L. M. Kristensen, "An industrial perspective on wireless sensor networksa survey of requirements, protocols, and challenges," *IEEE Communications Surveys & Tutorials*, vol. 16, no. 3, pp. 1391–1412, 2014.

[5] Q. Chi, H. Yan, C. Zhang, Z. Pang, and L. Da Xu, "A reconfigurable smart sensor interface for industrial wsn in iot environment," *IEEE transactions on industrial informatics*, vol. 10, no. 2, pp. 1417–1425, 2014.

[6] K. Das, P. Zand, and P. Havinga, "Industrial wireless monitoring with energy-harvesting devices," *IEEE internet computing*, vol. 21, no. 1, pp. 12–20, 2017.

[7] A. Abid, A. Kachouri, and A. Mahfoudhi, "Data analysis and outlier detection in smart city," in *International Conference on Smart, Monitored and Controlled Cities (SM2C)*. IEEE, 2017, pp. 1–4.

[8] R. Fantacci, T. Pecorella, R. Viti, and C. Carlini, "A network architecture solution for efficient iot wsn backhauling: challenges and opportunities," *IEEE Wireless Communications*, vol. 21, no. 4, pp. 113–119, 2014.

[9] A. Bagula, L. Castelli, and M. Zennaro, "On the design of smart parking networks in the smart cities: An optimal sensor placement model," *Sensors*, vol. 15, no. 7, pp. 443–467, 2015.

[10] Y. Zhang, L. Sun, H. Song, and X. Cao, "Ubiquitous wsn for healthcare: Recent advances and future prospects," *IEEE Internet of Things Journal*, vol. 1, no. 4, pp. 311–318, 2014.

[11] P. Corke, T. Wark, R. Jurdak, W. Hu, P. Valencia, and D. Moore, "Environmental wireless sensor networks," vol. 98, no. 11. IEEE, 2010, pp. 1903–1917.

[12] F. Viani, F. Robol, M. Bertolli, A. Polo, A. Massa, H. Ahmadi, and R. Boualleague, "A wireless monitoring system for phytosanitary treatment in smart farming applications," in *2016 IEEE International Symposium on Antennas and Propagation (APSURSI)*, June 2016, pp. 2001–2002.

[13] K. A. Kumar, A. V. Krishna, and K. S. Chatrapati, "Interference minimization protocol in heterogeneous wireless sensor networks for military applications," in *Proceedings of the 1st International Conference on Information and Communication Technology for Intelligent Systems*, vol. 2. Springer, 2016, pp. 479–487.

[14] S. Rani, S. H. Ahmed, R. Talwar, and J. Malhotra, "Can sensors collect big data? an energy-efficient big data gathering algorithm for a wsn," *IEEE Transactions on Industrial Informatics*, vol. 13, no. 4, pp. 1961–1968, 2017.

[15] S. Arjun, D, A. Bala, V. Dwarakanath, S. Sampada, K., R. Prahlada, and H. Pasupuleti, "Integrating cloud-wsn to analyze weather data and notify saas user alerts during weather disasters," in *2015 IEEE International Advance Computing Conference (IACC)*, June 2015, pp. 899–904.

[16] P. Rawat, K. D. Singh, H. Chaouchi, and J. M. Bonnin, "Wireless sensor networks: a survey on recent developments and potential synergies," *Journal of supercomputing*, vol. 68, no. 1, pp. 1–48, 2014.

[17] G. Papadopoulos, "Challenges in the design and implementation of wireless sensor networks: A holistic approach-development and planning tools, middleware, power efficiency, interoperability," in *2015 4th Mediterranean Conference on Embedded Computing (MECO)*, June 2015, pp. 1–3.

[18] I. Tomić and J. A. McCann, "A survey of potential security issues in existing wireless sensor network protocols," *IEEE Internet of Things Journal*, vol. 4, no. 6, pp. 1910–1923, 2017.

[19] A. Merlo, M. Migliardi, and L. Caviglione, "A survey on energy-aware security mechanisms," *Pervasive and Mobile Computing*, vol. 24, pp. 77–90, 2015.

[20] P. Helland, "Immutability changes everything," *Commun. ACM*, vol. 59, no. 1, pp. 64–70, Dec. 2015. [Online]. Available: http://doi.acm.org/10.1145/2844112

[21] C. Pappas, K. Argyraki, S. Bechtold, and A. Perrig, "Transparency instead of neutrality," in *Proceedings of the 14th ACM Workshop on Hot Topics in Networks*. ACM, 2015, p. 22.

[22] A. B. F. Guiloufi, N. Nasri, and A. Kachouri, "An energy-efficient unequal clustering algorithm using sierpinski trianglefor wsns," *Wireless Personal Communications*, vol. 88, no. 3, pp. 449–465, 2016.

[23] N. Sabor, M. Abo-Zahhad, S. Sasaki, and S. M. Ahmed, "An unequal multi-hop balanced immune clustering protocol for wireless sensor networks," *Applied Soft Computing*, vol. 43, pp. 372–389, 2016.

[24] S. Underwood, "Blockchain beyond bitcoin," *Communications of the ACM*, vol. 59, no. 11, pp. 15–17, 2016.

[25] U. Mukhopadhyay, A. Skjellum, O. Hambolu, J. Oakley, L. Yu, and R. Brooks, "A brief survey of cryptocurrency systems," in *2016 14th Annual Conference on Privacy, Security and Trust (PST)*, Dec 2016, pp. 745–752.

[26] Z. Zheng, S. Xie, H. Dai, X. Chen, and H. Wang, "An overview of blockchain technology: Architecture, consensus, and future trends," in *2017 IEEE International Congress on Big Data (BigData Congress)*, June 2017, pp. 557–564.

[27] S. Nakamoto, "Bitcoin: A peer-to-peer electronic cash system," 2008.

[28] N. N. Pokrovskaia, "Tax, financial and social regulatory mechanisms within the knowledge-driven economy. blockchain algorithms and fog computing for the efficient regulation," in *2017 IEEE International Conference on Soft Computing and Measurements (SCM)*, May 2017, pp. 709–712.

[29] Y. H. Chen, S. H. Chen, and I. C. Lin, "Blockchain based smart contract for bidding system," in *2018 IEEE International Conference on Applied System Invention (ICASI)*, April 2018, pp. 208–211.

[30] A. Dorri, S. S. Kanhere, R. Jurdak, and P. Gauravaram, "Blockchain for iot security and privacy: The case study of a smart home," in *2017 IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops)*, March 2017, pp. 618–623.

[31] T. Duong, A. Chepurnoy, L. Fan, and H.-S. Zhou, "Twinscoin: A cryptocurrency via proof-of-work and proof-of-stake," pp. 1–13, 2018. [Online]. Available: http://doi.acm.org/10.1145/3205230.3205233

[32] L. S. Sankar, M. Sindhu, and M. Sethumadhavan, "Survey of consensus protocols on blockchain applications," in *2017 4th International Conference on Advanced Computing and Communication Systems (ICACCS)*, Jan 2017, pp. 1–5.

[33] M. Rahman and S. Sampalli, "An efficient pairwise and group key management protocol for wireless sensor network," *Wireless Personal Communications*, vol. 84, no. 3, pp. 2035–2053, 2015.

[34] V. Shah-Mansouri, A.-H. Mohsenian-Rad, and V. W. Wong, "Lexicographically optimal routing for wireless sensor networks with multiple sinks," *IEEE Transactions on Vehicular Technology*, vol. 58, no. 3, pp. 1490–1500, 2009.

[35] D. R. Dandekar and P. Deshmukh, "Energy balancing multiple sink optimal deployment in multi-hop wireless sensor networks," in *3rd International Advance Computing Conference (IACC)*. IEEE, 2013, pp. 408–412.

[36] T. K. Jain, D. S. Saini, and S. V. Bhooshan, "Increasing lifetime of a wireless sensor network using multiple sinks," in *2014 11th International Conference on Information Technology: New Generations*, April 2014, pp. 616–619.

[37] B. Tang, J. Wang, X. Geng, Y. Zheng, and J.-U. Kim, "A novel data retrieving mechanism in wireless sensor networks with path-limited mobile sink," *International Journal of Grid & Distributed Computing*, vol. 5, pp. 133–140, 2012.

[38] A. T. Erman, A. Dilo, and P. Havinga, "A virtual infrastructure based on honeycomb tessellation for data dissemination in multi-sink mobile wireless sensor networks," *EURASIP Journal on Wireless Communications and Networking*, vol. 2012, no. 1, pp. 17–27, 2012.

[39] H. Grichi, O. Mosbahi, M. Khalgui, and Z. Li, "New power-oriented methodology for dynamic resizing and mobility of reconfigurable wireless sensor networks," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 48, no. 7, pp. 1120–1130, July 2018.

[40] C. Tunca, S. Isik, M. Y. Donmez, and C. Ersoy, "Ring routing: An energy-efficient routing protocol for wireless sensor networks with a mobile sink," *IEEE Transactions on Mobile Computing*, vol. 14, no. 9, pp. 1947–1960, 2015.

[41] A. W. Khan, J. I. Bangash, A. Ahmed, and A. H. Abdullah, "Qdvgdd: Query-driven virtual grid based data dissemination for wireless sensor networks using single mobile sink," *Wireless Networks*, pp. 1–13, 2017.

[42] S. Bhattacharjee and K. Agarwal, "Energy efficient multiple sink placement in wireless sensor networks," in *4th International Conference on Networking, Systems and Security (NSysS)*. IEEE, 2017, pp. 1–7.

[43] N. G. Reddy, N. Chitare, and S. Sampalli, "Deployment of multiple base-stations in clustering protocols of wireless sensor networks (wsns)," in *2013 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, Aug 2013, pp. 1003–1006.

[44] S. Mahmud and H. Wu, "Lifetime aware deployment of k base stations in wsns," pp. 89–98, 2012. [Online]. Available: http://doi.acm.org/10.1145/2387238.2387256

[45] S. Yasotha, V. Gopalakrishnan, and M. Mohankumar, "Multi-sink optimal repositioning for energy and power optimization in wireless sensor networks," *Wireless Personal Communications*, vol. 87, no. 2, pp. 335–348, 2016.

[46] R. Deng, S. He, and J. Chen, "An online algorithm for data collection by multiple sinks in wireless-sensor networks," *IEEE Transactions on Control of Network Systems*, vol. 5, no. 1, pp. 93–104, 2018.

[47] B. Gong, L. Li, S. Wang, and X. Zhou, "Multihop routing protocol with unequal clustering for wireless sensor networks," in *International Colloquium on Computing, Communication, Control, and Management (ISECS)*, vol. 2. IEEE, 2008, pp. 552–556.

[48] Y. Wang, T. Yang, and D. Zhang, "An energy efficient and balance hierarchical unequal clustering algorithm for large scale sensor network," *Journal of Information Technology*, vol. 8, no. 1, pp. 28–38, 2009.

[49] J. Yu, Y. Qi, G. Wang, Q. Guo, and X. Gu, "An energy-aware distributed unequal clustering protocol for wireless sensor networks," *International Journal of Distributed Sensor Networks*, vol. 7, no. 1, pp. 145–202, 2011.

[50] W. K. Lai, C. S. Fan, and L. Y. Lin, "Arranging cluster sizes and transmission ranges for wireless sensor networks," *Information Sciences*, vol. 183, no. 1, pp. 117–131, 2012.

[51] T. Liu, Q. Li, and P. Liang, "An energy-balancing clustering approach for gradient-based routing in wireless sensor networks," *Computer Communications*, vol. 35, no. 17, pp. 2150–2161, 2012.

[52] M. Mohamed-Lamine, "New clustering scheme for wireless sensor networks," in *2013 8th International Workshop on Systems, Signal Processing and their Applications (WoSSPA)*, May 2013, pp. 487–491.

[53] U. Hari, B. Ramachandran, and C. Johnson, "An unequally clustered multihop routing protocol for wireless sensor networks," in *2013 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, Aug 2013, pp. 1007–1011.

[54] M. M. Afsar and M. Younis, "An energy- and proximity-based unequal clustering algorithm for wireless sensor networks," in *39th Annual IEEE Conference on Local Computer Networks*, Sept 2014, pp. 262–269.

[55] N. Mazumdar and H. Om, "Coverage-aware unequal clustering algorithm for wireless sensor networks," *Procedia Computer Science*, vol. 57, pp. 660–669, 2015.

[56] V. Gupta and R. Pandey, "An improved energy aware distributed unequal clustering protocol for heterogeneous wireless sensor networks," *Engineering Science and Technology, an International Journal*, vol. 19, no. 2, pp. 1050–1058, 2016.

[57] Y. Deng, Y. Chen, Y. Zhang, and S. Mahadevan, "Fuzzy dijkstra algorithm for shortest path problem under uncertain environment," *Applied Soft Computing*, vol. 12, no. 3, pp. 1231–1237, 2012.

[58] Y. Chen, C. Shen, K. Zhang, H. Wang, and Q. Gao, "Leach algorithm based on energy consumption equilibrium," in *2018 International Conference on Intelligent Transportation, Big Data Smart City (ICITBS)*, Jan 2018, pp. 677–680.

[59] H. Bagci and A. Yazici, "An energy aware fuzzy unequal clustering algorithm for wireless sensor networks," in *International conference on Fuzzy systems (FUZZ)*. IEEE, 2010, pp. 1–8.

[60] S. Mao, C. Zhao, Z. Zhou, and Y. Ye, "An improved fuzzy unequal clustering algorithm for wireless sensor network," *Mobile Networks and Applications*, vol. 18, no. 2, pp. 206–214, 2013.

[61] S. Gajjar, A. Talati, M. Sarkar, and K. Dasgupta, "Fucp: Fuzzy based unequal clustering protocol for wireless sensor networks," in *2015 39th National Systems Conference (NSC)*, Dec 2015, pp. 1–6.

[62] R. Logambigai and A. Kannan, "Fuzzy logic based unequal clustering for wireless sensor networks," *Wireless Networks*, vol. 22, no. 3, pp. 945–957, 2016.

[63] B. Baranidharan and B. Santhi, "Ducf: Distributed load balancing unequal clustering in wireless sensor networks using fuzzy approach," *Applied Soft Computing*, vol. 40, pp. 495–506, 2016.

[64] S. S. Iyengar, H. C. Wu, N. Balakrishnan, and S. Y. Chang, "Biologically inspired cooperative routing for wireless mobile sensor networks," *IEEE Systems Journal*, vol. 1, no. 1, pp. 29–37, Sept 2007.

[65] J.-S. R. Jang, C.-T. Sun, and E. Mizutani, "Neuro-fuzzy and soft computing; a computational approach to learning and machine intelligence," *IEEE Transactions on Automatic Control*, vol. 42, no. 10, pp. 1482–1484, 1997.

[66] C.-J. Jiang, W.-R. Shi, X.-l. TANG *et al.*, "Energy-balanced unequal clustering protocol for wireless sensor networks," *The Journal of China Universities of Posts and Telecommunications*, vol. 17, no. 4, pp. 94–99, 2010.

[67] M. Abo-Zahhad, S. M. Ahmed, N. Sabor, and S. Sasaki, "A new energy-efficient adaptive clustering protocol based on genetic algorithm for improving the lifetime and the stable period of wireless sensor networks," *International Journal of Energy, Information and Communications*, vol. 5, no. 3, pp. 47–72, 2014.

[68] S. Salehian and S. K. Subraminiam, "Unequal clustering by improved particle swarm optimization in wireless sensor network," *Procedia Computer Science*, vol. 62, pp. 403–409, 2015.

[69] F. Xunli and D. Feiefi, "Shuffled frog leaping algorithm based unequal clustering strategy for wireless sensor networks," *Applied Mathematics & Information Sciences*, vol. 9, no. 3, pp. 1415–1426, 2015.

[70] S. Gajjar, M. Sarkar, and K. Dasgupta, "Famacro: fuzzy and ant colony optimization based mac/routing cross-layer protocol for wireless sensor networks," *Procedia Computer Science*, vol. 46, pp. 1014–1021, 2015.

[71] P. S. Rao and H. Banka, "Novel chemical reaction optimization based unequal clustering and routing algorithms for wireless sensor networks," *Wireless Networks*, vol. 23, no. 3, pp. 759–778, 2017.

[72] N. Sabor, M. Abo-Zahhad, S. Sasaki, and S. M. Ahmed, "An unequal multi-hop balanced immune clustering protocol for wireless sensor networks," *Applied Soft Computing*, vol. 43, pp. 372–389, 2016.

[73] Y. Shi *et al.*, "Particle swarm optimization: developments, applications and resources," in *evolutionary computation, 2001. Proceedings of the 2001 Congress on*, vol. 1. IEEE, 2001, pp. 81–86.

[74] H. Mühlenbein, M. Schomisch, and J. Born, "The parallel genetic algorithm as function optimizer," *Parallel computing*, vol. 17, no. 6-7, pp. 619–632, 1991.

[75] D. D. Falconer, F. Adachi, and B. Gudmundson, "Time division multiple access methods for wireless personal communications," *IEEE Communications Magazine*, vol. 33, no. 1, pp. 50–57, 1995.

[76] S. L. Miller, "An adaptive direct-sequence code-division multiple-access receiver for multiuser interference rejection," *IEEE Transactions on communications*, vol. 43, no. 234, pp. 1746–1755, 1995.

[77] B. Amiri, M. Fathian, and A. Maroosi, "Application of shuffled frog-leaping algorithm on clustering," *The International Journal of Advanced Manufacturing Technology*, vol. 45, no. 1-2, pp. 199–209, 2009.

[78] L. Jiang and J. Walrand, "A distributed csma algorithm for throughput and utility maximization in wireless networks," *IEEE/ACM Transactions on Networking (ToN)*, vol. 18, no. 3, pp. 960–972, 2010.

[79] A. Y. S. Lam and V. O. K. Li, "Chemical-reaction-inspired metaheuristic for optimization," *IEEE Transactions on Evolutionary Computation*, vol. 14, no. 3, pp. 381–399, June 2010.

[80] Z. Yang, J. Liu, and X. Chen, "An optimal mechanism of leach protocol for wireless sensor networks," in *2009 ISECS International Colloquium on Computing, Communication, Control, and Management*, vol. 4, Aug 2009, pp. 254–257.

[81] X. Lu, Y. Ding, and K. Hao, "Immune clonal selection algorithm for target coverage of wireless sensor networks," *International Journal of Modelling, Identification and Control*, vol. 12, no. 1-2, pp. 119–124, 2011.

[82] H. Xia, R.-h. Zhang, J. Yu, and Z.-k. Pan, "Energy-efficient routing algorithm based on unequal clustering and connected graph in wireless sensor networks," *International Journal of Wireless Information Networks*, vol. 23, no. 2, pp. 141–150, 2016.

[83] M. Ojo, D. Adami, and S. Giordano, "A sdn-iot architecture with nfv implementation," in *Globecom Workshops (GC Wkshps)*. IEEE, 2016, pp. 1–6.

[84] M. Amjad, M. Sharif, M. K. Afzal, and S. W. Kim, "Tinyos-new trends, comparative views, and supported sensing applications: A review," *IEEE Sensors Journal*, vol. 16, no. 9, pp. 2865–2889, 2016.

Appendix A

Sierpinski's triangle in WSN clustering

The Sierpinski's triangle, also known as Sierpinski's gasket, is a fractal triangle that is constructed from an initial equilateral triangle. Firstly, the triangle is divided into four smaller triangles by connecting the mid-points of its sides. Now, the middle triangle is removed and for each of the remaining triangles previous step is repeated. The process is illustrated in fig. A.2. The same approach can be adopted in clustering the WSNs where BS is located at the center and the monitoring area is square shaped such that its two diagonal creates four equilateral triangles. Then for each of the triangles the above mentioned approach is applied for cluster formation. The process is illustrated in fig. A.1.

Figure A.1: Sierpenski's triangle formation.

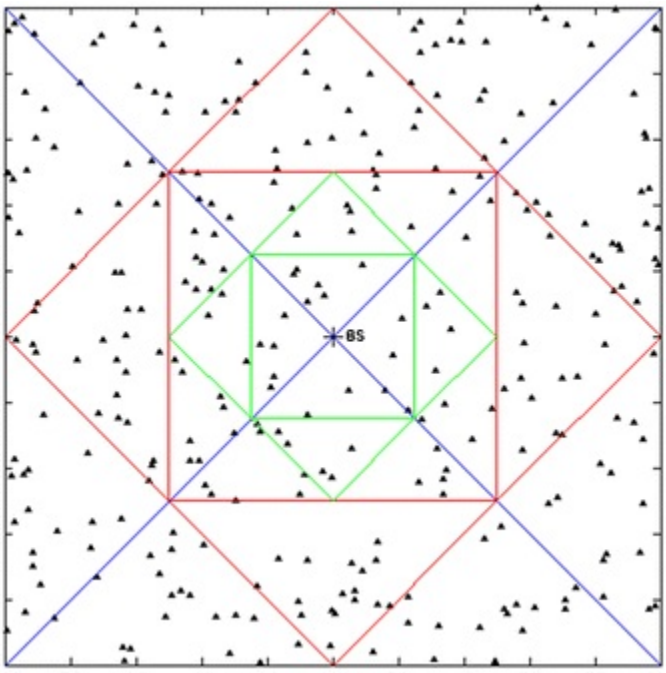


Figure A.2: Cluster formation with Sierpenski's triangle.