This chapter presents a detail literature survey of routing protocols, data aggregation and dissemination schemes and fault tolerant schemes in WSNs. It also further surveyed security requirements, possible attacks and elaborate the work done by other researchers.

Rest of the chapter has been organized in nine sections. Initial three sections deal with energy efficient routing protocol, data aggregation/dissemination schemes and fault tolerant schemes for WSNs. Next four sections discuss the work done with respect to security in WSNs followed by mobility management and finally the chapter is summarized.

3.1 Energy Efficient Routing Protocols in WSNs

According to [33, 39] the battery energy of a SN is depleted by: (i) computational processing and, (ii) transmission and reception of data. The network layer controls both of these factors.

Routing protocols have a large scope of research work when implemented in WSNs, because the functioning of these protocols depends upon the type of network structure designed for the application or the network operations carried out using these protocols for a specific application model.

3.1.1 Flat Routing Protocols in WSNs

In flat networks, each SN typically plays the same role and SNs collaborate together to perform the sensing task. Due to the large number of such SNs, it is not feasible to assign a global identifier to each SN. This consideration has led to data centric routing, where the BS sends queries to certain regions and waits for data from the sensors located in the selected regions. Since data is being requested through queries, attribute-based naming is necessary to specify the properties of data. Early works on data centric routing, viz., Sensor Protocols for Information via Negotiation (SPIN) [36] and Directed Diffusion [37] has minimizes the energy issue through data negotiation and elimination of redundant data. Rumor routing [38] is a variation of directed diffusion and
is mainly intended for applications where the geographic routing is not feasible.

3.1.2 Hierarchical Routing Protocols in WSNs
Hierarchical or cluster-based routing, are well known techniques with special advantages related to scalability and efficient communication. The concept of hierarchical routing is utilized to perform energy-efficient routing in WSNs. In a hierarchical architecture, higher energy SNs can be used to process and send the information while low energy SNs can be used to perform the sensing in the proximity of the target. Some of routing protocols in this group are: Low-Energy Adaptive Clustering Hierarchy (LEACH) [5], Power-Efficient Gathering in Sensor Information System (PEGASIS) [6], Threshold-Sensitive Energy Efficient TEEN [40] and Adaptive Threshold-Sensitive Energy Efficient (APTEEN) [41].

In LEACH cluster is formed on the basis of received signal strength and local CH is selected to route the data to the BS. The CH is elected randomly and rotated among the SNs in the cluster. The cluster head communicates directly with the BS and neighbouring CH in the network.

Distributed hierarchical approach in LEACH’s makes it scalable. Data aggregation is done by the CHs and then forwarded to the BS which means that compromise or failure of a CH causes a significant problem. Selection of CH in LEACH is a challenging task. Another drawback of LEACH is that it uses single-hop routing where each node can transmit directly to the cluster leader and the BS and can therefore not be applied to networks deployed in large regions.

In [42], authors introduce CH Election mechanism using Fuzzy (CHEF) logic. They apply the fuzzy if-then rule to the CH election mechanism. By using fuzzy logic, the computational overhead is reduced and the network lifetime is extended. In addition, the operation of this mechanism is localized. The BS does not collect and elect CHs. The SNs only elect CHs among themselves using the fuzzy logic. The routing protocol [43] based on dynamic clustering protocol which grants a large lifetime for the network. The key idea of this protocol is to reduce transmission in intra-clusters when the objective is
to collect the maximum or minimum data values in a region (like temperature, humidity, etc.).

A number of different strategies for multi-hop routing, including minimum distance and minimum-hop routing have been presented in [44]. In [45], authors have proposed efficient integer linear program formulations for assigning SNs to clusters in a two-tired network where the higher powered relay SNs are used as CHs. The relay SNs can use a single hop model or multi-hop model to send the data to the BS. The objectives in both cases are to maximize the lifetime of the network.

The authors in [46] take a unique look at the CH election problem, specifically concentrating on applications where the maintenance of full network coverage is the main requirement. The approach for cluster-based network organization is based on a set of coverage-aware cost metrics that favor SNs deployed in densely populated network areas as better candidates for CH, active SNs and routers. Compared with traditional energy based selection methods, using coverage-aware selection of CH, active SNs and routers in clustered WSN increases the time during which full coverage of the monitored area can be maintained depending on the application scenario.

In [47], authors analyzing the advantages and disadvantages of conventional hierarchical communication protocols using MATLAB, they have developed Distance-Based Segmentation (DBS), a cluster-based protocol that significantly decreases the energy imbalance in the network by integrating the distance of the SNs from the BS into clustering policies. Furthermore, a Media Access Control (MAC) protocol that eliminates redundant delays in the cluster formation period of conventional protocols is utilized as media access scheme.

The SNs closest to the BS tend to deplete their energy faster than other SNs [48, 49], which is known as an energy hole around the BS. No more data can be delivered to the BS after energy hole appears. Therefore, a lot of energy is wastes and the network lifetime ends prematurely.

In [50], authors have proposed a non-uniform SN distribution strategy to achieve the sub-balanced energy depletion. In this strategy if the number of SNs in coronas increases then the network can achieve sub-balanced energy depletion.
In [51], authors have presented an important corona model to maximize the network lifetime by using adjustable transmission range. In this model the maximum transmission range of sensors is divided into several levels. The sensors SNs belong to the same corona have the same range of transmission, whereas different coronas have different transmission ranges. The authors concluded that transmission ranges to all coronas is the effective approach to extend the network lifetime.

In [52], authors have presented a short survey on the main techniques used for energy conservation in WSNs. The main focus is primarily on duty cycle scheme which represent the most suitable technique for energy saving. In [53], authors reviewed the existing definitions of network lifetime as propose in the literature. They discussed about the merits and demerits of the existing definitions and summarized additional requirements. They have also introduced a number of new performance metrics that have found to be useful in the context of WSN applications.

The authors in [54] have presented a systematic survey and comprehensive taxonomy of the energy saving schemes. They have also introduced mobility as a new energy saving paradigm with the purpose of maximizing the network lifetime. In [55], authors have proposed a new genetic algorithm (GA), for scheduling the data gathering of relay SNs, which significantly extends the network lifetime of a relay SN network. For smaller networks, GA based approach is always finds the optimal solution. This algorithm can easily handle large networks as compared to traditional routing schemes.

The authors have presented a novel CH election problem in [44], specifically designed for applications where the maintenance of full network coverage is the main requirement. This approach is based on a set of coverage-aware cost metrics that favor SNs deployed in densely populated network areas as better candidates for CH SNs, active SNs and routers. Compared with traditional energy-based selection methods, the coverage-aware selection of CH SNs increases the network lifetime depending on the application scenario.

In [57], authors have proposed and evaluated an Unequal Cluster based Routing (UCR) protocol for mitigating the hot spot problem in WSNs. It is designed for long lived, source-driven WSN applications, such as periodical environmental information reporting.
In [58], authors have studied a generic strategy of radioactivity minimization wherein each SN maintains the radio switched on just in the expected packet arrival intervals and guarantees low communication latency. They define a probabilistic model that allows the evaluation of the packet loss probability that results from the reduced radioactivity. This model can be used to optimally choose the radioactivity intervals that achieve a certain probability of successful packet delivery for a specific radioactivity strategy. They also define a cost model that estimates the energy consumption of the proposed strategies, under specific settings.

WSNs attracted lots of researchers because of its potential wide applications and special challenges. For past few years, WSNs mainly focused on technologies based on the homogeneous WSN in which all SNs have same system resource but recently heterogeneous WSN is becoming more and more popular and the results of researches [59, 60] show that heterogeneous SNs can prolong network lifetime and improve network reliability without significantly increasing the cost.

A heterogeneous SN is more expensive and is capable to provide data filtering, fusion and transport. It may posses one or more type of heterogeneous SNs, e.g., enhanced energy capacity or communication capability. They may be line powered, or their batteries may be replaced easily. Compared with the normal SNs, they may be configured with more powerful microprocessor and more memory. They may communicate with the BS via high bandwidth, long-distance network, such as Ethernet. The presence of heterogeneous SNs in a WSN can increase network reliability and lifetime. The main basic and important deployment problem is to decide how many and where heterogeneous SNs should be deployed in the network.

The heterogeneous resources are basically divided into three categories: computational heterogeneity, link heterogeneity and energy heterogeneity [61].

Computational heterogeneity signifies that the heterogeneous SN has a more powerful microprocessor or microcontroller and more storage memory than the normal SN. The SNs with more powerful computational resources can provide complex data processing and longer-term storage.
Link heterogeneity means that the heterogeneous SN has high-bandwidth and long haul network transceiver (Ethernet or 802.11) network, than the normal SN. Link heterogeneity can provide more reliable data transmission. Therefore, the reliability of the data transmission will increase by link heterogeneity.

Energy heterogeneity means that the heterogeneous SN is line powered, or its battery is replaceable. Among above three categories of resource heterogeneity, the energy heterogeneity is most important because both computational heterogeneity and link heterogeneity will consume more battery energy resource. If there is no energy heterogeneity, computational heterogeneity and link heterogeneity will bring negative impact to the WSN.

In an application for habitat monitoring [62], authors proposed a tiered system architecture in which data collected at numerous, inexpensive SNs is filtered by local processing on its way through larger, more capable and more expensive SNs. The necessity of heterogeneity and the mechanisms of packet forwarding and processing are demonstrated in [63, 64]. However, how to use heterogeneous SNs effectively has not been studied comprehensively.

PEGASIS [6] discusses how to extend the lifetime of WSNs. It is a near optimal chain-based protocol, not a clustering scheme. In PEGASIS, each SN communicates only with an adjacent neighbor and takes turns transmitting to the BS. A number of issues are discussed, viz., reducing the amount of energy spent per processing round, minimizing overall distance between non-leader SNs of the system and minimizing the number of data transmission to the BS. However, in PEGASIS, SNs die in random locations since the CHs have been chosen without any concern on an overall lifetime of each SN.

To prolong the network lifetime by distributing energy consumption, HEED (Hybrid Energy-Efficient Distributed Clustering) [7] is a standalone distributed clustering approach in which each SN considers two factors: remaining energy and communication cost before making a decision to join one cluster or the other. In HEED, once selected a CH is maintained for a fixed number of iterations. This is in contrast to some other approaches where the CHs are newly selected in every step. This is to reduce the unnecessary high setup cost associated with the CH selection process.
In [65], authors have studied the impact of heterogeneity of SNs, in terms of their energy and have proposed a heterogeneous – aware protocol to prolong the time interval before the death of the first SN, which is crucial for many applications where the feedback from the WSN must be reliable. In [66], authors have proposed a new distributed energy efficient clustering scheme for heterogeneous WSNs, which are called DEEC. In DEEC, the CHs are elected by a probability based on the ratio between residual energy of each SN and the average energy of the network. The SNs with high initial and residual energy will have more chances to become CHs than the SNs with low energy.

In [67], authors address the deployment problem of heterogeneous WSNs and is supported by an algorithm to decide how many and where heterogeneous SNs should be deployed in the WSN. The core algorithm is based on the locations of all SNs, can optimize placement of heterogeneous SNs in an arbitrary WSN to increase the network lifetime and reliability. In [68], authors propose a distributed election clustering protocol to prolong the stable region of heterogeneous WSNs, which is based on remaining energy and communication cost to elect suitable CH SNs. Compared with classical clustering protocol, it can maintain load balancing of networks and extremely prolong the stable region and the network lifetime.

In [69], authors have presented an Energy-Efficient Protocol with Static Clustering (EEPSC) which partitions the network into static clusters and utilizes CHs to distribute the energy load among high power SNs for extending the network lifetime. But the authors have not investigated the effect of heterogeneity in the network system.

Energy-Efficient Hierarchical Clustering Algorithm (EEHCA) scheme [70] improves the performance of LEACH and HEED (Hybrid Energy-Efficient Distributed clustering), in terms of network lifetime. EEHCA adopts a new method for CH election, which can avoid the frequent election of CH. In order to improve the performance of the WSN new concept of backup CHs is introduced. Therefore, when SNs finished the communication within their own clusters and the CHs have finished the data aggregation, the head clusters will transmit aggregated data to the BS.

A novel energy efficient centralized clustering algorithm for WSNs is presented in [71]. This algorithm generates a set of possible clustering
alternatives, which helps in finding the optimal clustering and present a performance evaluation of the proposed scheme by using two performance metrics Max-Min and Max-Sum. It is found that Max-Sum improves the system lifetime performance over Low Energy Adaptive Clustering Hierarchy-C (LEACH-C).

In [72], authors presented the Multi-hop Routing Algorithm for Inter CH Communication (MRACHC). The algorithm was multilayer multi-hop routing algorithm, which worked on the principle of divide and conquers and was performing well in terms of load balance and energy efficiency then LEACH.

3.1.3 Location Based Routing Protocols
In location-based routing, all the SNs are addressed by using their locations. Depending upon the strength of the incoming signals, it is possible to calculate the nearest neighboring SN’s distance. Due to obstacles in the network often the signal strength becomes weaker and SNs are unable to find the nearest neighbor SNs. There are many location-based schemes of which Geographic and Energy aware Routing (GEAR) [73] and Geographic Adaptive Fidelity (GAF) [74] are two important schemes.

3.2 Data Aggregation and Dissemination Schemes in WSNs
The main goal of data aggregation algorithms is to gather and aggregate data in an energy efficient manner so that network lifetime can be enhanced. Data gathering is defined as the systematic collection of sensed data from multiple sensors to be eventually transmitted to the BS for processing. Since SNs are energy constrained, it is inefficient for all the sensors to transmit the data directly to the BS. Data generated from neighboring sensors is often redundant and highly correlated. In addition, the amount of data generated in large WSNs is usually enormous for the BS to process. Hence, we need methods for combining data into high quality information at the sensors or intermediate SNs which can reduce the number of packets transmitted to the BS resulting in conservation of energy and bandwidth. This can be accomplished by data aggregation. It may be defined as the process of aggregating the data from multiple SNs to eliminate redundant transmission and provide fused
information to the BS and it usually involves the data fusion at intermediate SNs and transmits the aggregated data to the BS.

Data aggregation attempts to collect the most critical data from the sensors and make it available to the BS in an energy efficient manner with minimum data latency. Data latency is important in many applications such as environment monitoring where the freshness of data is also an important factor. It is critical to develop energy efficient data aggregation algorithms so that network lifetime can be enhanced. There are several factors which determine the energy efficiency of a WSN such as network architecture, the data aggregation mechanism and the routing protocol. The architecture of the WSN plays a vital role in the performance of different data aggregation protocols.

3.2.1 In-Network Aggregation

In a typical WSN scenario, different SN collect data from the environment and then send it to some central SN or BS which analyzes and process the data and then send it to the application. But in many cases, data produced by different SNs can be jointly processed while being forwarded to the BS. In-network aggregation deals with this distributed processing of data within the network.

Data aggregation techniques explore how the data is to be routed in the network and the processing method that are applied on the packets received by a SN. They have a great impact on the energy consumption of SNs and thus on network efficiency by reducing number of transmission or length of packet. In [75], authors define in-network aggregation process “In-network aggregation is the global process of gathering and routing information through a multi-hop network, processing data at intermediate SNs with the objective of reducing resource consumption (in particular energy), thereby increasing network lifetime”.

There are two approaches for in-network aggregation: with size reduction and without size reduction. In-network aggregation with size reduction refers to the process of combining and compressing the data packets received by a SN from its neighbors in order to reduce the packet length to be transmitted or forwarded towards BS. As an example, consider the situation when a SN receives two packets which have a spatial correlated data. In this case it is
worthless to send both packets. Instead of that one should apply any function like Average (AVG), Maximum (MAX) and Minimum (MIN) to send a single packet. This approach considerably reduces the amount of bits transmitted in the network and thus saving a lot of energy but on the other hand, it also reduces the precision of data value received. In-network aggregation without size reduction refers to the process of merging data packets received from different neighbors into a single data packet but without processing the value of data. As an example, two packets may contain different physical quantities (i.e., temperature and humidity) and they can be merged into a single packet by keeping both values intact but keeping a single header. This approach preserves the value of data and transmit more bits in the network but still reduce the overhead by keeping single header.

These two approaches depend on many factors like the type of application, data arrival rate and network characteristics. There is also a trade-off between energy consumption and precision of data for the two approaches.

Most of the work available in literature on in-network aggregation mainly deals with problem of forwarding packets from source to BS, to facilitate aggregation therein. Actually the main idea behind were to enhance existing routing protocols such that they can efficiently aggregate data. Most of the data aggregation techniques fall under three categories - tree-based approaches, multi-path approaches and cluster-based approaches. There are also some hybrid approaches that combine any of these three techniques. These approaches are described in the coming sections with giving details of some of the main techniques by different authors.

3.2.2 Tree Based Approach
The simplest way to aggregate data is to organize the SNs in a hierarchical manner and then select some SNs as the aggregation point or aggregators. The tree-based approach perform aggregation by constructing an aggregation tree, which could be a minimum spanning tree, rooted at BS and source SNs are considered as leaves. Each SN has a parent SN to forward its data. Flow of data starts from leaf SNs up to the BS and therein the aggregation done by parent SNs. The way this approach operates has some drawbacks. In case of packet loss at any level of tree, the data will be lost not only for a single level
but for the whole related sub-tree as well. In spite of high cost for maintaining tree structure in dynamic networks and scarce robustness of the system, this approach is suitable for designing optimal energy efficient aggregation technique.

A data-centric protocol is based on aggregation tress, known as Tiny Aggregation (TAG) approach [76]. TAG works in two phases: distribution phase and collection phase. In distribution phase, TAG organizes SNs into a routing tree rooted at BS. The tree formation starts with broadcasting a message from BS specify level or distance from root. When a SN receive this message it sets its own level to be the level of message plus one and elect parent as SN from which it receives the message. After that, SN rebroadcast this message with its own level. This process continues till all the SNs elect their parents. After tree formation, BS sends queries along structure to all SNs in the network. TAG uses database Structured Query Language (SQL) for selection and aggregation functions. In collection phase, data is forwarded and aggregated from leaf SNs to root. A parent SN has to wait for data from its entire child SN before it can send its aggregate up the tree. Apart from the simple aggregation function provided by SQL (e.g., COUNT, MIN, MAX, SUM and AVG), TAG also partitions aggregates according to the duplicate sensitivity, exemplary and summary and monotonic properties. Though TAG periodically refresh tree structure of network but most of the tree-based schemes are inefficient for dynamic network.

In [77], authors proposed a reactive data-centric protocol for applications where BS ask some specific information by flooding, known as directed diffusion paradigm as shown in Figure 3.1. The main idea behind directed diffusion paradigm is to combine data coming from different sources and en-route them by eliminating redundancy, minimizing the number of data transmission, thus maximizing network lifetime. Directed diffusion consists of several elements: interests, data messages, gradients and reinforcements.

The BS requests data by broadcasting an interest message which contains a description of a sensing task. This interest message propagates through the network hop-by-hop and each SN also broadcast interest message to its neighbor. As interest message propagates throughout the network, gradients are setup by every SN within the network. The gradient direction is set toward
the neighboring SN from which the interest is received. This process continues
till gradients are setup from source SN to BS. Loops are not checked at this
stage but removed at later stage. After these paths of information flow are
formed, then best path are reinforced to prevent further flooding according to a
local rule. Data aggregation took place on the way of different paths from
different sources to BS. The BS periodically refreshes and resends the interest
message and it starts to receive data from sources to provide reliability. The
problem with directed diffusion is that it may not be applied to applications
(e.g. environmental monitoring) that require continuous data delivery to BS.
This is because query driven on demand data model may not help in this
regard.

Also matching data to queries might require some extra overhead at the
SNs. Mobility of BS SNs can also degrade the performance as path from
sources to BS cannot be updated until next interest message is flooded
throughout the network. To cope up with above issue if introduce frequent
flooding then also too much overhead of bandwidth and battery power will be
introduced. Furthermore, exploratory data follow all possible paths in the
network following gradients which lead to unnecessary communications
overhead.

In [78], authors proposed a new low-control-overhead data dissemination
scheme, which they called as Pseudo-Distance Data Dissemination (PDDD),
for efficiently disseminating data from all SNs to mobile BS. In the
development of PDDD the following assumptions are made: (1) all source
SNs maintain routes to mobile BS, (2) no periodical messaging for topological
changes due to mobile BS, (3) all links are bi-directional and no control
messages will lost, (4) mobile BS SNs will be equipped with unlimited battery
power and (5) network partitioning is not considered. Data dissemination
process is influenced by Directed Diffusion [37]. Though mobile BS
periodically broadcast interest message, SNs do not send exploratory data and
do not wait reinforcement message because each SN already has routes to the
BS. After getting interest message, adjacent SNs set a parent-child relationship
using pseudo-distance of each SN and finally a Partial Ordered Graph (POG)
has been build. Optimal data dissemination is achieved in terms of path length
by forwarding packets to a parent SN until topology is unchanged. Then each
SN is assigned a level for a corresponding BS with pseudo-distance. In order to overcome the shortcoming of POG, author used Totally Ordered Graph (TOG) in place of POG. Due to mobility of BS all SNs have to maintain routes and for any change in topology SNs have to again change route accordingly which led to energy waste.

In [79], authors proposed an energy-aware spanning (ESPAN) tree algorithm for data aggregation. E-Span is a distributed protocol in which source SN which has highest residual energy is chosen as root. Other source SNs choose their parent based on residual energy and distance to the root. The protocol uses configuration message to exchange information of SN, i.e., residual energy and distance to the root. Each SN performs single-hop broadcast operation to send packets. Single-hop broadcast refers to the operation of sending a packet to all single-hop neighbors [79].

### 3.2.3 Multi-Path Approach

One of the main drawbacks of tree-based approach is the scarce robustness of the system. To overcome this drawback, many approaches are available in the literature. The theme behind these approaches is that instead of sending partially aggregated data to a single parent SN in aggregation tree, a SN sends data over multiple paths. In this scheme each SN sends data to its possibly multiple neighbors by exploiting the wireless medium characteristic. Thus, data will flow from sources to BS along multiple paths and aggregation can be performed by each intermediate SN. Clearly, schemes using this approach will make the system robust but with some extra overhead. One of the aggregation structures that fit well with this approach is ring topology, where network is divided into concentric circles with defining levels according to the hop distance from BS.

In [80], authors have presented a data aggregation technique using multi-path approach, known as synopsis diffusion. Synopsis diffusion works in two phases: distribution of queries and data retrieval phase. During distribution of queries phase, a SN sends a query in the network. The network SNs then form a set of rings around the querying SN. The SN which is i hop away from querying SN is considered as ring R_i. In the second phase, aggregation starts
from outermost ring and propagate level by level towards the BS. Here, a source SN may have multiple paths towards BS.

In [81], authors describe a new strategy for data gathering in WSN which considers both issues: energy efficiency and robustness. Authors first say that single path to connect each SN to the BS is simple and energy-saving approach but expose a high risk of disconnection due to SN/link failures. But multi-path approach would require more SNs to participate with consequent waste of energy. Authors present a clever use of multi-path only when there is loss of packet which is implemented by smart caching of data at SNs. Authors also argue that in many practical situation data may be gathered only from a particular region, so they use a different approach that relies on a spanning tree and provides alternative paths only when a malfunctioning is detected. Algorithm adopts a tree-based approach for forwarding packets through the network. In the ideal situation when no failures occur, this is certainly the best choice, as the minimum numbers of SNs are engaged in the transmission phase. In the presence of link or SN failures, the algorithm discovers an alternative path which ensures the delivery of as many packets as possible within the time constraints. The problem with this approach is that it may cause the arising of hot spots and SNs along preferred paths will consume their energy resources quickly, possibly causing disconnection in the network.

3.2.4 Cluster Based Approach

We have discussed about hierarchical organization of the network in tree-based approach. Another important technique to organize the network in hierarchical manner is cluster-based approach. In cluster-based approach, whole network is divided into several clusters. Each cluster has a CH which is selected among cluster members. CHs do the role of aggregator which aggregate data received from cluster members locally and then transmit the result to BS. The advantages and disadvantages of the cluster-based approaches are similar to tree-based approaches.

A Maximum Lifetime Data Aggregation (MLDA) algorithm which finds data gathering schedule provided location of sensors and BS, data packet size and energy of each sensor has presented in [82]. A data gathering schedule specifies how data packet are collected from sensors and transmitted to the BS.
for each round. A schedule can be thought of as a collection of aggregation trees.

In [83], authors present a Two-Phase Clustering (TPC) scheme. Phase I of this scheme creates clusters with a CH and each SN within that cluster form a direct link with CH. Phase I of this scheme is similar to various scheme used for clustering. The CH rotation is localized and is done based on the remaining energy level of the SNs which minimizes time variance of sensors and this leads to energy saving from unnecessary CH rotation. In phase II, each SN within the cluster searches for a neighbor closer than CH which is called data relay point and setup up a data relay link. Now the SNs within a cluster either use direct link or data relay link to send their data to CH which is an energy efficient scheme. The data relay point aggregates data at forwarding time to another data relay point or CH. In case of high network density, TPC phase II setups unnecessary data relay link between neighbors as closely deployed sensors will sense same data and this lead to a waste of energy.

An energy efficient and secure pattern based data aggregation protocol is designed for clustered environment which is presented in [84]. In conventional method data is aggregated at CH and it eliminates redundancy by checking the content of data. This protocol says that instead of sending raw data to CH, the cluster members send corresponding pattern codes to CH for data aggregation. If multiple SNs send the same pattern code then only one of them is finally selected for sending actual data to CH. For pattern matching, authors present a pattern comparison algorithm.

In [85], authors discuss two parts: in the first part, they propose an analytical model to formulate the data aggregation delay at a SN that operates under IEEE 802.15.4 and uses beacon enabled slotted CSMA-CA. Since most of the sensors that are manufactured comply with IEEE 802.15.4 and CSMA-CA is adopted for its simplicity compared with contention free protocols. In the second part, they propose a heuristic algorithm for constructing an energy efficient data aggregation tree under the constraint that the data gathered from the furthest SN will reach the root of the tree within the specified delay bound.

In [86], authors propose LEECF (Low-Energy Event Centric Fusion), an event-centric-based protocol which utilizes the centric SN to aggregate the event data among the triggered sensors in a short delay to decrease the
complexity. They also consider a fast data fusion algorithm by designing a confidence matrix for target identity and prove that it attains the same precision as Dempster-Shafer (DS) evidence formula in a much less time.

The proposed scheme begins with a setup phase of level at which the direction and distance of source to BS are organized. When an event occurs, it enters into the search phase of aggregator, when the center SN of source SNs is found, followed by the construction phase of fusion Steiner tree, then the event data converges to the aggregator (tree root) along the fusion tree at which the data fusion phase begins to run and the fusion result is relayed to the BS.

In [87], the authors develop a data fusion algorithm that combines the cluster-based design of WSNs using fuzzy logic methods. The performance of a network is increased by eliminating redundancy and power consumption, ensuring fault-tolerance between sensors and managing effectively the available communication bandwidth between network components.

### 3.3 Fault Tolerant Schemes in WSNs

SNs in WSNs are prone to failure due to energy depletion, hardware failure, communication link errors, malicious attack and so on. Unlike the cellular and ad hoc networks where energy has no limits in BS or batteries can be replaced as needed, SNs in WSNs have very limited energy and their batteries cannot usually be recharged or replaced due to hostile or hazardous environments. So, one important characteristic of WSNs is the stringent power management of the SNs. Two components of a SN, viz., sensing unit and wireless transceiver, usually directly interact with the environment which is subject to variety of physical, chemical and biological factors. It results in low reliability of performance of the SNs. Even if condition of the hardware is good, the communication between the SNs are affected by many factors, such as signal strength, antenna angle, obstacles, weather conditions, interference.

The services provided by a WSN to a large extent depend on fault tolerance, because it cannot be assumed that all sources of error can be eliminated, even through careful engineering. By considering service availability, we understand the probability with which a request will lead to a
valid and useful response. The availability of WSN is defined as:

\[ P(A) = \frac{MTTF}{MTTF + MTTR} \]

where MTTF stands for Mean Time to Failure and MTTR stands for Mean Time to Repair [88].

By analyzing this equation we can conclude that systems that constantly fail and require long repair time will result in systems with very low availability. However, systems that have a high MTTF and can be quickly repaired are considered highly available systems.

Attacks on the availability of SNs are analyzed in [71]. These attacks directly influence the availability of services. Fault tolerant systems can overcome faults and system failures, therefore increasing the MTTF and system availability. Fault tolerance techniques especially crafted for WSNs have not been extensively studied so far. The most important one is replication, which is well suited for WSNs due to their inherent SN redundancy. Fault Tolerance has been discussed in detail in the literature on distributed systems [90]. Although WSNs have special characteristics that distinguish them from traditional distributed systems, many of the existing techniques still apply. An introduction to the fundamental mechanisms for implementing replicated services in distributed systems is discussed in [91].

To comprehend fault tolerance mechanisms, it is important to point out the difference between faults, errors and failures. Various definitions of these terms have been used [87, 89]. This article refers to the definition given in [89]:

- A fault is any kind of defect that leads to an error.
- An error corresponds to an incorrect (undefined) system state. Such a state may lead to a failure.
- A failure is the (observable) manifestation of an error, which occurs when the system deviates from its specification and cannot deliver its intended functionality.

3.3.1 Fault Tolerance At Different Levels

There are five levels of fault tolerance, viz., physical layer, hardware layer, system software layer, middleware layer and application layer [74]. Based on
the study, we classify fault tolerance in WSNs into four levels from the system point of view. More specifically, fault tolerance in a WSN system may exist at hardware layer, software layer, network communication layer and application layer.

3.3.1.1 Hardware Layer
Faults at hardware layer can be caused by malfunction of any hardware component of a SN, such as memory, battery, microprocessor, sensing unit and network interface (wireless radio). There are three main reasons that cause hardware failure of SNs. The first is that WSNs are usually for commercial use and SNs are cost sensitive. Therefore, design of a SN will not always use the highest quality components. The second is that strict energy constraints which restrict long and reliable performance of SNs. In [93-95] due to stress from the environment and inadequate enclosures, the SNs were exposed to direct contact with water causing short circuits. The report of a large-scale deployment in a potatoes field [96] indicated that the antennas from the SNs were quite fragile and would become loose when inserting the SN into the packaging. When the battery of a SN reaches a certain stage, sensor readings may become incorrect. This has been observed in a field many outlier readings were generated in the network caused by imminent battery failure. For example, sensor readings may become incorrect when the battery of a SN reaches a certain level [97].

The third is that WSNs are often deployed in harsh and hazardous environments which affect normal operation of SNs. The wireless radios of SNs are severely affected by these environment factors.

3.3.1.2 Software Layer
Software of a SN consists of two components: system software, such as operating system and middleware, such as communication, routing and aggregation. An important component of system software is to support distributed and simultaneous execution of localized algorithms. Software bugs are a common source of errors in WSNs. One promising method is through software diversity where each program is implemented in several different versions. Since it is difficult to provide fault tolerance in an economic way at
hardware level of a SN, numerous fault-tolerant approaches are expected at the middleware level. The majority of current applications in WSNs are simple. To adapt the real-life applications, there is a need to develop much more complex middleware for WSNs.

In [98], the researchers reported that a software bug caused the longest continuous network outage taking the system offline for three days until the SNs could be reprogrammed manually.

### 3.3.1.3 Network Communication Layer

Faults at network communication layer are the faults on wireless communication links. Assuming that there is no error on hardware, link faults in WSNs are usually related to surrounding environments. In addition, link faults can also be caused by radio interference of SNs. For example, SN ‘a’ cannot successfully receive a message from SN ‘b’ if SN ‘a’ is within interference range of other SNs that are transmitting messages at the same time. The standard way to enhance the performance of wireless communication is to use aggressive error correction schemes and retransmission. These two methods may cause further delay of operation. It should be pointed out that there is always a trade-off between fault tolerance and efficiency.

In WSNs, communication links between SNs are highly volatile and are not always yield the same delivery rate of messages in field trials as in lab trials. For instance, in [99] a delivery rate of only 76% of the messages was observed; in [83,100] the instability of the links between SNs leads to constant changes in the routing paths.

In [101], authors present an experimental study on fault tolerant routing for wireless sensor grid networks. A network may be separated into partitions owing to the broken radio links resulted from SN failures. Since the leveling algorithm works effectively for un-partitioned network, a combined technique with extended transmission range is investigated to resolve the network-partitioning problem. In [102], authors propose FTPASC (Fault Tolerant Power Aware protocol with Static Clustering), a static clustering based protocol, which engages high power SNs for power consuming tasks and as a result extends the network lifetime. In classical static clustering, when the CHs
energy was depleted all the SNs of that cluster lose their communication with the BS and become essentially dead. In each cluster, FTPASC chooses the SN with maximum energy as the CH. Thus, not only there is always one CH for each cluster, but also the overhead of dynamic clustering is removed.

3.4 Security Requirements and Possible Attacks In WSNs

Security goals in WSNs depend on the need to know what we are going to protect. We determine four security goals in WSNs: confidentiality, integrity, authentication and availability (CIAA).

**Confidentiality:** This is the ability to conceal a message from a passive attacker, where the message communicated on WSNs remains confidential.

**Integrity:** This is the ability to confirm that the message has not been tampered with, altered, or changed while it was on the network.

**Authentication:** If the messages are from the SN it claims to be from, we need to determine the reliability of the message’s origin.

**Availability:** If a SN has the ability to use the resources, the network is available for the messages to move on.

3.4.1 Security Classes

In [104] author has identified four classes of security in computing systems. We integrate these four threat classes in WSNs. In computing systems the major assets are hardware, software and data. In WSNs, our goal is to protect the network itself, the SNs and communication among the SNs. There are four classes of threats which exploit the vulnerability of our security goals.

- ** Interruption:** In interruption, a communication link in WSNs becomes lost or unavailable. Examples of this sort of threat are SN capture, message corruption, addition of malicious code and so on.

- **Interception:** In interception a WSN has been compromised by an adversary where the attacker gains unauthorized access to a SN or to data in it. Example of this type of attack is capture of SN.

- **Modification:** In modification an adversary not only accesses the data but tampers with it, for example, modifying the data packets being transmitted, causing a denial of service attack, flooding the network
with bogus data and so on.

- **Fabrication**: In fabrication, an adversary adds false data, making the whole network unreliable.

### 3.4.2 Security Threats in WSNs

There are number of security threats in computing. In [105] authors have presented possible security threats in WSNs and are given below:

**Passive Information Gathering**: An adversary with powerful resources collecting information from WSNs if information is not encrypted.

**SN Subversion**: Capture of a SN may reveal its information including disclosure of cryptographic keys, hence compromising the whole WSN.

**False SN**: Addition of a malicious SN by an adversary to inject the malicious data, false SN would be computationally robust to lure other SNs to send data to it.

**SN Malfunction**: A malfunctioning SN will generate inaccurate data that would jeopardize the integrity of a WSN, especially when that SN is a data aggregating SN for example, a cluster leader.

**SN Outage**: Means SNs stops functioning. WSNs protocols should be robust enough to mitigate the effects of SN outages by providing an alternate route.

**Message Corruption**: When contents of a message are modified by an attacker, it compromises the message integrity.

**Traffic Analysis**: Even if the message transfer is encrypted in WSNs, its still leaves the high probability of analysis of communication patterns and sensor activities revealing enough information to enable an adversary to cause more malicious harm to WSNs.

### 3.4.3 Security Attacks in WSNs

In [106] authors have presented detailed attacks in WSNs which are described as follows:

- **Routing Loops**: In WSNs, routing loops attack the information exchanged between SNs. False error messages are generated when an attacker alters and replays the routing information. Routing loops
attract or repel the network traffic and increase SN-to-SN latency.

- **Selective Forwarding:** Selective forwarding is a way to influence the network traffic by believing that all the participating SNs in network are reliable to forward the message. In selective forwarding attack, malicious SNs simply drop certain messages instead of forwarding every message. Once a malicious SN cherry picks on the messages, it reduces the latency and deceives the neighboring SNs that they are on a shorter route. Effectiveness of this attack depends on two factors. First is the location of the malicious SN. The closer it is to the BS, the more traffic it will attract. Second is the percentage of messages it drops. When a selective forwarder drops more messages and forwards less, it retains its energy level, thus remaining powerful to trick the neighboring SNs.

- **Sinkhole Attacks:** In sinkhole attacks, an adversary attracts the traffic to a compromised SN. The simplest way of creating a sinkhole is to place a malicious SN where it can attract most of the traffic, possibly closer to the BS or malicious SN itself, which deceptively acts as a BS. One reason for sinkhole attacks is to make selective forwarding possible to attract the traffic toward a compromised SN. The nature of WSNs where all the traffic flows toward one BS makes this type of attack more susceptible.

- **Sybil Attacks:** This is a type of attack where a SN creates multiple illegitimate identities in WSNs either by fabricating or stealing the identities of legitimate SNs. Sybil attacks can be used against routing algorithms and topology maintenance; it reduces the effectiveness of fault-tolerant schemes such as distributed storage and disparity. Another malicious factor is geographic routing where a Sybil SN can appear at more than one place simultaneously.

- **Wormholes:** In wormhole attacks, an adversary positioned closer to the BS can completely disrupt the traffic by tunneling messages over a low-latency link. Here an adversary convinces the SNs which are multi-hop away that they are closer to the BS. This creates a sinkhole because an adversary on the other side of the sinkhole provides a
better route to the BS.

- **Hello Flood Attacks:** This involves broadcasting a message with stronger transmission power and pretending that the HELLO message is coming from the BS. Message receiving SNs assume that the HELLO message sending SN is the closest one and they try to send all their messages through this SN. In this type of attack, all SNs will be responding to HELLO floods and wasting the energies. The real BS will also be broadcasting the similar messages but will have only a few SNs responding to it.

- **DoS Attacks:** Denial of service attacks occur at a physical level, causing radio jamming, interfering with the network protocol, battery exhaustion and so on.

### 3.5 Security Approach in WSNs

On the basis of study we have observed that in WSNs security should be implemented at least at following four layers.

#### 3.5.1 Application Layer

Data are collected and managed at application layer therefore it is important to ensure the reliability of data. In [107]author has presented a resilient aggregation scheme that is applicable to a cluster-based network where a cluster leader acts as an aggregator in WSNs. However, this technique is applicable if the aggregating SN is in the range with all the source SNs and there is no intervening aggregator between the aggregator and source SNs. In the hierarchical clustering approach, a communication channel between the aggregator and BS has potentially limited bandwidth because the cluster leader as an aggregator itself is a SN [107, 108]. To prove the validity of the aggregation, cluster leaders use the cryptographic techniques to ensure the data reliability. We will discuss the cryptography in key management section.

#### 3.5.2 Network Layer

The network layer is responsible for routing of messages from SN to SN, SN to cluster leader, cluster leaders to cluster leaders, cluster leaders to the BS
Routing protocols in WSNs are of two types: (1) ID-based protocols, in which packets are routed to the destination based on the IDs specified in the packets and (2) data-centric protocols [109] in which packets contain attributes that specify the type of data being provided. In [108] authors have described routing attacks presented in [106], in WSNs as below:

- Packets are dropped completely, or selectively.
- The network is flooded with global broadcasts.
- Some SNs in the network are misguided into believing that SNs are either multiple hops away or do not exist at all in the neighbors.
- A significant proportion of the traffic is tunneled from one place in the network to another distant place of the network depriving other parts of the network that under normal circumstances would have received the traffic themselves.
- Sometimes traffic is lured to a particular SN or a small group of SNs, depriving other parts of the network that normally would have received the traffic themselves.

Security of routing protocols depends on the location of SNs and the encryption techniques.

### 3.5.3 Data Link Layer

The data link layer does the error detection and correction, as well as encoding of data. The link layer is vulnerable to jamming and Denial of Service attacks. TinySec [110] has introduced link layer encryption, which depends on a key management scheme. However, an attacker having better energy efficiency can still wage an attack. Protocols like LMAC [111] have better anti-jamming properties, which are viable countermeasures at this layer.

### 3.5.4 Physical Layer

The physical layer emphasizes the transmission media between sending and receiving SNs; the data rate, signal strength and frequency types are also
addressed in this layer. Ideally, the FHSS frequency hopping spread spectrum is used in WSNs.

3.6 Key Management Schemes in WSNs

Key management is the set of techniques and procedures which support the establishment and maintenance of keying materials between authorized parties.

A Master key based, pre distribution scheme is based on a single master key that is pre deployed in all the SNs. Each SN uses part of the memory to store the master keys. A pair of SNs exchanges random nonce values. The pair uses a master key to establish the session keys. Compromise of master key makes the whole WSN insecure. Dutertre et al. [27] have presented a lightweight key management system where they have proposed more than one master key on an assumption that WSNs are deployed in various phases.

A probabilistic key pre-distribution scheme is presented by Eschenauer and Gligor [103] where each SN receives a random subset of keys from a large key pool before deployment. To agree on a key for communication, two SNs find one common key within their subsets and use that key as their shared key. In this scheme if an adversary captures single SN, it can communicate with all those SNs which share a common key with the captured SN.

Chan et al. [28] extended the idea of Eschenauer and Gligor [103] and developed three key pre-distribution schemes; q-composite, multi-path reinforcement and random- pair-wise keys schemes to overcome the weakness of basic random key pre distribution scheme. In this scheme the key pool size S is reduced and multiple keys are used to establish communications instead of just one. This scheme strengthens the network resilience against SN capture when the number of SNs captured is small. However, if the large number of SNs have been captured, this scheme tends to reveal a larger fraction of network to the adversary.

A BS is used as a trusted third party to distribute the keys to provide link keys to SNs such as Kerberos [32]. In this keying mechanism a BS becomes the single point of vulnerability if compromised. Due to extensive communication with the BS, SNs closer to the BS will lose their battery early. This approach assumes some level of reliable communication mode between
the SN and the BS before any key scheme is deployed.

Du et al. [29] pair-wise key pre-distribution is an effort to improve the resilience of the network by lowering the initial payoff of smaller scale network attacks and pushing the adversary towards a larger scaled attack to compromise the network. In later work Du et al. [30] present a key scheme based on deployment knowledge. This key management scheme takes advantage of deployment knowledge where the sensor’s position is known prior to deployment. Because of the randomness of deployment, it is not feasible to know the exact location of neighbors, but it is realistic to know the set of likely neighbors.

Du et al. [31] make an extension on the key management scheme developed by Eschenauer and Gligor [103]. This scheme describes the deployment knowledge where sensors are prearranged in a sequence before deployment. Once dropped from a plane or helicopter, sensors have better probability to recognize the prearranged sequence of neighbor sensors. This prior deployment knowledge is useful for key pre-distribution. When neighbor sensors are known, key pre-distribution becomes trivial; for each SN x we just need to generate a pair-wise key between x and each of its neighboring SNs and we save these keys in x’s memory. This guarantees that each SN can establish a secure channel with each of its neighbors after deployment.

3.7 Secure Routing Schemes in WSNs
Secure routing in WSNs is challenging due to the unique characteristics of WSNs as compared to wired and wireless ad hoc networks. Traditional IP-based routing is not a viable solution due to a relatively large number of SNs because the overhead of IP maintenance is very high. The following are the issues that need to be kept in mind while designing a secure routing protocol in WSNs.

- SNs are self-organizing due to the ad hoc deployment and SNs are left unattended after the deployment.
- In WSNs, most of the time flow of data would be from SNs to cluster leader and BS.
- Careful route management due to SNs limitations.
• Frequent changes in network topology due to the dynamic nature of WSNs.
• WSNs are application-specific and data-centric.
• Secure location of SNs because Global Posting Systems (GPS) are not suitable for WSNs.

There are very few routing protocols proposed to address the secure routing issues in WSNs.

SPINS is a suite of security building blocks proposed by Perig et al. [36], which is a collection of security protocols SNEP (Secure Network Encryption Protocol) and μTESLA, a micro version of TESLA (Time Efficient Streamed Loss-tolerant Authentication). SNEP provides data confidentiality and two-way data authentication with minimal overheads and μTESLA provides authenticated streaming broadcast. SPINS leaves some questions like security of compromised SNs, DoS issues and network traffic analysis issues unaddressed. Furthermore, this protocol assumes a static network topology ignoring the ad hoc and mobile nature of SNs.

INSENS [34, 35] tolerates intrusion by bypassing the malicious SNs instead of detecting the intrusion. Even if a malicious SN exists in the network, INSENS mitigates the impact of that intrusion. INSENS does not rely on conventional anomaly-based intrusion detection techniques; instead it uses an intrusion-tolerant mechanism that reduces the harm caused by presence of a small number of undetected intruders in the network by incorporating redundancy in routing.

3.8 Mobility Management Schemes in WSNs
Mobility of the SNs has been exploited for improving, sensing and communication coverage [12-14]. The idea presented in [12] is to have the SNs move into positions that minimize the energy cost of reporting streams of data to the sink, which is statically placed. The protocols proposed by Wang et al. [13] aim at moving mobile SNs from densely deployed areas to areas with coverage holes, where for some reasons a limited number of SNs have been deployed. The three protocols are proven by simulations to be effective in
terms of coverage, deployment time and moving distance. Minimization of the
energy consumption of moving SNs has been subsequently addressed by the
authors in [14] by letting the SN “move logically”—that is, only after they can
decide whether their movement maximizes the coverage or not. The idea of
using mobile SNs has been explored within the robotic community, where
mobile robots are equipped with SNs. Typical examples are the works by
Howard et al. [15, 16]. In [15] an algorithm for the deployment of the
members of a robotic team into an unknown environment is given. The aim of
this algorithm is to maximize the coverage area, while maintaining line-of-
sight contact among the robots. In [16] the same authors present another
scheme to distribute the mobile SNs throughout a given area. The fields are
constructed in such a way that each SN is repelled by obstacles and other SNs,
thereby forcing the SN to spread throughout the area.

Distributed algorithms for the mobility of SNs have been investigated in
[17]. In [17] mobility algorithms are proposed that move the SNs to positions
that reduce the transmission power needed to send the data to the (static) sink.
The positions for the moving SNs are determined via distributed simulated
annealing, as opposed to a greedy strategy that could lead to a suboptimal
placement. By using distributed simulated annealing, a SN based only on
information on its current neighbors accepts a “bad move” with a positive
probability. This move could be locally not optimal, but could benefit the
network globally in the longer run.

Works that consider mobile SNs and robots are mostly concerned with
sensor deployment time and with sensing coverage. The costs associated with
sensor movements as well as the cost of transmitting sensed data is often not
considered and network lifetime is rarely a metric of interest.

While the bulk of published work envisioned SNs to be stationary, some
investigated the possibility of attaching SNs to moveable entities such as
robots [20, 21]. Sensor's mobility has been exploited to boost the
dependability of WSNs. For example, SNs can re-spread in the area to ensure
uniform coverage, move closer to loaded SNs in order to prevent bottlenecks,
increase bandwidth by carrying data to the BS and so on [22–26]. Proposed
schemes for dynamic SNs positioning in the literature can be categorized into
two groups based on when relocation is exploited: (1) post-deployment and (2) on-demand relocation.

3.8.1 Post-Deployment Relocation
This type of relocation is pursued at the conclusion of the sensor deployment phase when the SNs are being positioned in the area. In the WSN applications, sensor deployment is performed randomly due to the inaccessibility of the monitored areas. However, this random configuration usually does not provide an adequate coverage of the area without deploying an excessive number of SNs. Alternatively, the coverage quality can be improved by moving the SNs if they are able to do so. In that case, the SNs can be relocated to the regions that do not have the desired level of coverage or even are not covered at all.

Wang et al. [19] utilizes sensor’s ability to move to distribute the SNs as evenly as possible in the region. The goal is to maximize the area coverage within the least time duration and with minimal overhead in terms of travel distances and inter-sensor message traffic. The main idea is that each sensor assesses the coverage in its vicinity after deployment and decides whether it should move to boost the coverage. Nonetheless, this process still can be very slow and hence prolong the deployment time. With the objective of reducing the overall deployment time, Wu and Yang [24] proposed another solution to the same problem based on two-dimensional scan of a clustered network, called SMART.

3.8.2 On-Demand Repositioning
Instead of relocating the SNs at the deployment phase, sensor relocation can be used on demand to improve certain performance metrics such as coverage, network lifetime and so on. This can be decided during the network operation based on the changes in either application-level needs or the network state. For instance, in some applications, there can be an increasing number of dysfunctional SNs in a particular part of the area necessitating the redistribution of available SNs. In addition to improving coverage, the energy consumption can be reduced through on-demand relocation of SNs in order to reach the best efficient topology. The approach presented in [18] performs sensor relocation to counter holes in coverage caused by SNs failure.
3.9. Summary

In this chapter, we have presented a state of the art literature survey on energy-efficient, secure, fault tolerant, mobility management schemes for WSNs. In the next Chapter a reconfigurable network management protocol is presented which is energy efficient and fault tolerant in nature.