CHAPTER 2
LITERATURE REVIEW

2.1 Introduction
The main objective of this chapter is to provide background information for understanding of the subsequent chapters. The current chapter provides a critical review of literature on mobile ad-hoc networks and the routing protocols. Section 2.2 discusses the concept and characteristics of mobile ad-hoc networks. The features and classification of the ad-hoc routing protocols are discussed in Sections 2.3, and 2.4 respectively. Section 2.5 explains the description of Ad-hoc on demand distance vector (AODV) protocol with its mechanisms. Section 2.6 discusses the link failure in mobile ad-hoc networks, whereas the link state prediction is illustrated in Section 2.7. The related work is discussed in Section 2.8.

2.1.1 Concept and Characteristics of Mobile Ad-Hoc Networks
A mobile ad-hoc network is considered as a special type of wireless networks which contains a collection of wireless devices such as tablets, iPad, and laptops. These devices organize a temporary network without depending on fixed infrastructures. Basically, mobile ad-hoc networks are peer to peer multihop mobile wireless networks according to which each packet information is sent from the source to the destination node through an intermediate node as demonstrated in Figure 2.1. The communication of mobile nodes with each other is performed through the wireless medium. Mobile ad-hoc networks may operate independent or with few selected routers to communicate with an infrastructure network such as the Internet as mentioned in [10]. The topology of mobile ad-hoc network is dynamic because the mobile nodes can leave or join the network at anytime and anywhere. The mobile nodes that are located within the transmission range of other nodes are called (neighbors). This means each neighbor node can transfer the data packets directly with other nodes. However, when a mobile node needs to transmit the data packets to other neighbor nodes, the data is routed via a sequence of multiple hops with intermediate nodes which act as routers. Mobile ad-hoc networks have been
considered a robust wireless communication network where its robustness has been attributed to its area of deployment and configurable. According to Das et al. [11] one of the major advantages of mobile ad-hoc networks is the attractive choice for military field. However, the freedom and flexibility of mobile ad-hoc networks emerge with some complication and challenges. The lack of pre-existing infrastructure, frequently changing topology, multi-hop nature, bandwidth constraints, energy consumption, and network scalability, add the challenges and complexities to the traditional of wireless networks.

![Dynamic Topology in Mobile Ad-Hoc Networks](image)

**Figure 2.1: Dynamic Topology in Mobile Ad-Hoc Networks**

2.1.2 **Mobility and Network Topology**

Node mobility is the main factor that affect the performance of MANET as stated in Tonguz [12] and Lenders [13]. The fast unpredictable movement of the intermediate nodes in MANET environment dynamically change the network topology. This in turn causes a break in the discovered communication links. When the links are disrupted, the result is dropping of large amount of the data packets that were sent to the next hop during the forwarding process. As a result, the network forces the underlying protocol to recover the failed links or starts discovering new routes to continue transmission the data packets. However, the increas of many activities in mobile ad-hoc
networks, causes a frequent exchange of routing information over the bandwidth constrained communication channel. According to Bandana [14], this leads to increase in overall routing overhead of the network.

![Network Topology Diagrams](image-url)

**a. Network state at time t**

**b. Network state at time t+1**

Figure 2.2: Dynamically Changing Network Topology
Figure 2.2 illustrates the dynamic changes of topology under node movement and multi-hop nature of mobile ad-hoc networks. Figures 2.2 (a) and 2.2 (b) represent the network topology at time t and t+1 respectively where the mobile node (6) presents the source and mobile node (1) presents the destination. The dotted line indicates the active link and the arrow line indicates the speed of the mobile nodes. At time t, the active route is <6, 4, 2, 1> where the mobile nodes (4) and (2) are intermediate nodes. However, the mobile nodes (4) and (6) tend to move in different positions as if they are not in the same transmission range. Simultaneously, mobile node (3) in moving towards node (6), the node (4) appears within the transmission range of the source and the destination nodes. At time t+1, the weaker links fail and fresher links are established. Therefore, a new active route < 6, 3, 4, 1> is created with the node (4) and node (3) as an intermediate node. However, these events result in a large number of network activities that has to be accomplished by the underlying routing protocol.

2.1.3 Effect of Mobility and Link Stability

Mobility directly affects number of the link facts such as the number of established links, number of disrupted links and the duration of links in MANET as mentioned in Lenders et al. [13]. As a result, the amount of data packets that can be carried over the wireless link is reduced. Mobility of the individual mobile node causes a dynamic change of the network topology, thereby prompting the routing protocol to perform network reconfiguration continuously. Furthermore, mobility affects the end-to-end performance measurements such as network throughput, the amount of control routing overhead, packet delivery ratio, average delay, and allocation of resources as stated in Mullen and Huang [15]. High node mobility makes the communication route unstable due to occurrence of more link breakages within the communication route. In addition, the wireless links might also be broken due to various sources of signal interference and the packet collision. It has been discovered that the node mobility and interference/collision have totally different effects on the lifetime of the routes as stated in Lenders et al. [13]. However, the node mobility is more likely to be the main cause of link failure in case of longer activation time. Due to the ad-hoc network characteristics, the most difficult and important issue in MANET is designing of dynamic routing protocol that efficiently determines the routes between pairs of mobile nodes. The routing protocol in mobile ad-hoc network must be able to keep up with
the node's mobility. The mobile ad-hoc routing protocols and their features are discussed in the next section.

2.2 Routing Protocols in Mobile Ad-Hoc Networks

The primary goal of using routing protocols in ad-hoc networks is to discover and establish routes between mobile nodes to send the data packets from the source to destination so that messages will be delivered timely as stated in Saito et al. [16]. Routing protocols for mobile ad-hoc networks usually call for mobility management and scalable design. Mobility management is performed by exchanging the routing information between moving hosts in mobile ad-hoc networks. However, frequent information exchanges can be costly in terms of bandwidth and power consumption. By minimizing the frequent information changes, these costs will be decreased. The scalable design works for the large size networks which needs both routing protocols and resource consumption to be scalable. Designing a reliable and efficient routing protocol is very challenging in MANET due to the dynamic topology that results from mobility of nodes and limited network bandwidth as well as different types of wireless communication restraints as mentioned in [17, 18]. These restraints are variable link quality, energy constrained nodes, interference, and exposed terminal problems. However, the issues of the routing protocol performance in mobile ad-hoc networks are divided into two areas. The first area focuses on the limitation of the environments such as wireless, limited bandwidth, power consumption, and security. The second area concerns about the way of transferring the data communication which might take place leading to the desired qualitative features of mobile ad-hoc routing protocols as stated in [19]. The MANET working group determines some desirable qualitative properties of ad-hoc routing protocol as stated in RFC 2501 [20]. It is necessary that the proposed protocol in this research meets some of the properties in the list summarised below:

- Distributed operation: The routing protocol should provide multiple routes from the source to the destination to solve the problems of congestion. The decentralized nature of mobile ad-hoc networks requires executing its operation in a distributed fashion.
- Demand-based operation: The routing protocol should be capable of adapting the network traffic pattern on a demand fashion instead of assuming that the routes are always active. It should be capable of updating the routing information between mobile nodes at all times as
well. The on-demand-based operation leads to reduce the power consumption and save the bandwidth resources more efficiently.

- Low routing overhead: The routing protocol should have a low routing overhead because of the exchanges of the routing information when the network topology changes occur to be adapted quickly to such changes.
- Optimal routing: The routing protocol should allow quick establishment of routes in order to be used before they become invalid (non fresh routes).

2.3 Classification of Mobile Ad-Hoc Routing Protocols

The classification of ad-hoc routing protocols can be divided according to their approaches for creating and maintaining routes. These protocols can be classified into proactive, reactive, and hybrid. The classification depends on the reaction of the mobile nodes in the routing determination process as stated in Djenouri et al. [21]. Figure 2.3 shows the taxonomy of ad-hoc routing protocols.

![Figure 2.3: Taxonomy of Ad-Hoc Routing Protocols](image-url)
2.3.1 Proactive Routing Approach

In this category of the routing protocols, the route is always available for the data communication from the source node to the destination node. Every mobile node in proactive routing approach keeps updating the routing information periodically in its routing table. The main advantage of table driven routing approach is when the source node requires to send the data packets to a desired destination. The route is already available and there is no latency for establishing the routes. On the other hand, the disadvantage of this approach is some routes in some situations may never be used for communication which leads to consume network bandwidth and node energy.

The most popular proactive routing protocols are the Destination Sequenced Distance Vector Routing (DSDV) [22], Optimized Link State Routing Protocol (OLSR) [23], and Wireless Routing Protocol (WRP) [24]. Figure 2.4 demonstrates the concept of proactive routing approach [25]. For example, if the source node (A) needs to send the data packets to the destination (D), the node A should search in a previously prepared topology table (stored on a source A itself) to find the destination D.

![Route Discovery for Proactive Approach](image)

**Figure 2.4: Route Discovery for Proactive Approach (Adopted from [23])**
2.3.2 Reactive Routing Approach

Reactive routing protocols are also called "On-Demand Routing". On-demand routing protocols means that a route is established exclusively when the data packets are in need to be delivered to the destination. On-Demand routing protocols takes a different approach for routing than table driven protocols such as Temporally Ordered Routing Algorithm (TORA) [26], Ad hoc On-Demand Distance Vector Routing (AODV) [27], and Dynamic Source Routing (DSR) [28]. The routes to the destination are discovered when they are actually needed. When a source node requires to send data packets to the destination, it starts checking the routing table to determine whether it has a route to the destination or not. If there is no existing route, the source node starts route discovery procedure in order to find a path to the destination.

Reactive routing protocols are based on Distance Vector concept (DV) which can significantly decrease the routing overhead and the power consumption. These protocols do not need to keep searching and maintaining the routes because there is no data traffic to send as stated in Royer and Toh [17].

Figure 2.5 demonstrates that when there is a request from the source node (A) to send the data packets to the destination node (D), a route discovery process is initiated by broadcasting a route request packet to all the neighbor nodes searching for the destination (D) [25]. When the node (D) receives this message, it sends a route reply to build a route to the source node (A).

![Figure 2.5: Route Discovery for Reactive Approach (Adopted from [23])]
The works in [17, 18, 29, 19,12] have evaluated the comparison of proactive and reactive with multi-hop routing protocols such as DSDV, TORA, DSR, ABR, and AODV. Through their simulation results, they have observed that the reactive routing protocols outperform proactive protocols in terms of packets delivery ratio, routing overhead, energy efficiency, and stability. Another main motivation behind using reactive routing approach in this research is the reduction of the routing load. In fact, high routing load normally has a significant performance effect on low bandwidth wireless network. Therefore, this research mainly focuses on the reactive routing approach.

The next sections will discuss the details of the basic operation of Ad hoc On-Demand Distance Vector Routing (AODV) protocol. In addition, they will present the relevant works that have been done in single path routing protocols regarding the link failure problem for the extensions of AODV protocol.

2.4 Ad hoc On-Demand Distance Vector Routing (AODV)

In Ad hoc On-demand Distance Vector Routing (AODV) to find a route to the destination, the source broadcasts a route request packet. This broadcast message propagates through the network until it reaches an intermediate node that has recent route information about the destination or until it reaches the destination. When intermediate nodes forwards the route request packet it records in its own tables which node the route request came from. This information is used to form the reply path for the route reply packet as AODV uses only symmetric links. As the route reply packet traverses back to the source, the nodes along the reverse path enter the routing information into their tables. Whenever a link failure occurs, the source is notified and a route discovery can be requested again if needed. It is based on standard Distance Vector Algorithm. Nodes maintain route cache and uses destination sequence number for each route entry does nothing when connection between end points is still valid Route Discovery Mechanism is initiated when a route to new destination is needed by broadcasting a Route Request Packet (RREQ).Route Error Packets (RERR) is used to erase broken Links.
Figure 2.6: AODV Protocol (a) Propagation of RREQ, (b) Path of the RREP to the source
When the network entity needs to send a message to another node, it calls upon AODV to determine the next-hop. Whenever an AODV router receives a request to send a message, it checks its routing table to see if a route exists. Each routing table entry consists of the following fields:

- Destination address
- Next hop address
- Destination sequence number
- Hop count

If a route exists, the router simply forwards the message to the next hop. Otherwise, it saves the message in a message queue, and then it initiates a route request to determine a route. The following flow chart illustrates this process:

![Figure 2.7: AODV](image)

Upon receipt of the routing information, it updates its routing table and sends the queued message(s). AODV nodes use four types of messages to communicate among each other. Route Request (RREQ) and Route Reply (RREP) messages are used for route discovery. Route Error (RERR) messages and HELLO messages are used for route maintenance.
The following sections describe route determination and route maintenance in greater detail. AODV Route Discovery When a node needs to determine a route to a destination node, it floods the network with a Route Request (RREQ) message. The originating node broadcasts a RREQ message to its neighboring nodes, which broadcast the message to their neighbors, and so on. To prevent cycles, each node remembers recently forwarded route requests in a route request buffer (see next section). As these requests spread through the network, intermediate nodes store reverse routes back to the originating node. Since an intermediate node could have many reverse routes, it always picks the route with the smallest hop count. When a node receiving the request either knows of a “fresh enough” route to the destination (see section on sequence numbers), or is itself the destination, the node generates a Route Reply (RREP) message, and sends this message along the reverse path back towards the originating node. As the RREP message passes through intermediate nodes, these nodes update their routing tables, so that in the future, messages can be routed through these nodes to the destination. Notice that it is possible for the RREQ originator to receive a RREP message from more than one node. In this case, the RREQ originator will update its routing table with the most “recent” routing information; that is, it uses the route with the greatest destination sequence number.

2.4.1 Route Discovery Mechanism of AODV Protocol

The Route Request Buffer

In the flooding protocol described above, when a node originates or forwards a route request message to its neighbors, the node will likely receive the same route request message back from its neighbors. To prevent nodes from resending the same RREQs (causing infinite cycles), each node maintains a route request buffer, which contains a list of recently broadcasted route requests. Before forwarding a RREQ message, a node always checks the buffer to make sure it has not already forwarded the request. RREQ messages are also stored in the buffer by a node that originates a RREP message. The purpose for this is so a node does not send multiple RREPs for duplicate RREQs that may have arrived from different paths. The exception is if the node receives a RREQ with a better route (i.e. smaller hop count), in which case a new RREP will be sent. Each entry
in the route request buffer consists of a pair of values: the address of the node that originated the request, and a route request identification number (RREQ id). The RREQ id uniquely identifies a request originated by a given node. Therefore, the pair uniquely identifies a request across all nodes in the network. To prevent the route request buffers from growing indefinitely, each entry expires after a certain period of time, and then is removed. Furthermore, each node’s buffer has a maximum size. If nodes are to be added beyond this maximum, then the oldest entries will be removed to make room.

**Sequence Numbers**

Each destination (node) maintains a monotonically increasing sequence number, which serves as a logical time at that node. Also, every route entry includes a destination sequence number, which indicates the “time” at the destination node when the route was created. The protocol uses sequence numbers to ensure that nodes only update routes with “newer” ones. Doing so, we also ensure loop-freedom for all routes to a destination. All RREQ messages include the originator’s sequence number, and its (latest known) destination sequence number. Nodes receiving the RREQ add/update routes to the originator with the originator sequence number, assuming this new number is greater than that of any existing entry. If the node receives an identical RREQ message via another path, the originator sequence numbers would be the same, so in this case, the node would pick the route with the smaller hop count. If a node receiving the RREQ message has a route to the desired destination, then we use sequence numbers to determine whether this route is “fresh enough” to use as a reply to the route request. To do this, we check if this node’s destination sequence number is at least as great as the maximum destination sequence number of all nodes through which the RREQ message has passed. If this is the case, then we can roughly guess that this route is not terribly out-of-date, and we send a RREP back to the originator. As with RREQ messages, RREP messages also include destination sequence numbers. This is so nodes along the route path can update their routing table entries with the latest destination sequence number. Link Monitoring & Route Maintenance Each node keeps track of a precursor list, and an outgoing list. A precursor list is a set of nodes that route through the given node. The outgoing list is the set of next-hops that this node routes through. In networks where all routes are bi-directional, these
lists are essentially the same. Each node periodically sends HELLO messages to its precursors. A node decides to send a HELLO message to a given precursor only if no message has been sent to that precursor recently. Correspondingly, each node expects to periodically receive messages (not limited to HELLO messages) from each of its outgoing nodes. If a node has received no messages from some outgoing node for an extended period of time, then that node is presumed to be no longer reachable. Whenever a node determines one of its next-hops to be unreachable, it removes all affected route entries, and generates a Route Error (RERR) message. This RERR message contains a list of all destinations that have become unreachable as a result of the broken link. The node sends the RERR to each of its precursors. These precursors update their routing tables, and in turn forward the RERR to their precursors, and so on. To prevent RERR message loops, a node only forwards a RERR message if at least one route has been removed.

The following flow chart summarizes the action of an AODV node when processing an incoming message.
Event received by the Process

Checking of message type?

RREQ message

Update route to originator (if found better than the existing)

RREP message

Update route table, outgoing list & precursor

RERR message

Remove the affected routes

Is destination?

Yes

Send RREP

No

Has fresh enough routes?

Yes

Forward RREP to next hop

No

If not in buffer, forward RREQ to neighbors

Is originator?

Yes

Send queued messages

No

Forward RERR to precursors

At least one remove?

Yes

Stop

No

No

No

No

Yes

Yes

Yes

Yes

No

Figure 2.8: Flow chart Route Request

Figure 2.9: Route Reply in AODV
2.4.2 Route Maintenance Mechanism of AODV Protocol

In an ad hoc network, links are likely to break due to the mobility of the nodes and the ephemeral nature of the wireless channel. Hence, there must be a mechanism in place to repair routes when links within active routes break. An active route is defined to be a route that has recently been utilized for the transmission of data packets. When such a link break occurs, the node upstream of the break (i.e., the node closer to the source node), invalidates in its routing table all destinations that become unreachable due to the loss of the link. It then creates a Route Error (RERR) message, in which it lists each of these lost destinations. The node sends the RERR upstream towards the source node. If there are multiple previous hops that were utilizing this link, the node broadcasts the RERR; otherwise, it is unicast. In the Figure 2.10, the link between nodes N2 and N5 on the path from Source to Destination is broken. Node N2 invalidates its route table entries for both nodes N5 and N8, creates a RERR message listing these nodes, and sends the RERR upstream towards the source. When a node receives a RERR, it first checks whether the node that sent the RERR is its next hop to any of the destinations listed in the RERR. If the sending node is the next hop to any of these destinations, the node invalidates these routes in its route table and then propagates the RERR back towards the source. The RERR continues to be forwarded in this manner until it is received by the source. Once the source receives the RERR, it can re-initiate route discovery if it still requires the route.

Figure 2.10: Link breaks Notification
2.4.3 Route Cache of AODV Protocol

We saw that AODV finds new routes by making a route request broadcast which travels through various intermediate nodes before reaching the destination node. These requests carry a lot of information about the network topology as they pass through different node but due to lack of caches at intermediate nodes, this information cannot be tapped by the nodes to be use later. So by providing all the nodes with an extra cache and by making changes in the RREQ packets such as to enable them to carry the information about the nodes through which the byepass, intermediate nodes can save the information about the network topology contained in the RREQ packets. This reduces the time and overhead to find new routes in cases of route failure. From now no, we will call the AODV with cache enabled as AODV–WC and AODV without caching as AODV-WOC.

Each node now has a separate queue (apart from the queue which AODV has for maintaining routing information) which acts as a cache for the routes. For this purpose, we have used the same queue structure which AODV uses for maintaining its routes. To reduce the problem of stale caching, a cache timer is introduced in the caches and an appropriate cache timeout value is found to get the maximum efficiency from the cache even in the case of high mobility (low pause time). So any route that does not get updated within the cache timeout period from the time of its addition to the cache, is discarded as stale Route request packets (RREQ) should be able to carry the node addresses and latest sequence numbers (It is the same sequence number as used by AODV to check the freshness of a route) of the intermediate nodes they have passed before reaching the destination node. For this purpose, we have implemented a special data structure in the AODV RREQ packet header which forms a link list of node addresses and sequence numbers of the nodes through which the packet has crossed. All the nodes on receiving a route request packet should, apart from doing their already specified tasks, read the node addresses and sequence numbers in the packet and add them to their caches as the nodes reachable from the last node through which the packet is coming. Then before broadcasting the packet to the neighboring nodes, the nodes should append their own address and a latest unused sequence number into the packet.

As AODV is not a source routing protocol like DSR, so caching of routes can cause the
problem of looping of data packets because of deletion of routes due to cache timeout. In order to avoid the looping of packets, a packet sequence number is generated by the source node before sending the packet by incrementing by one the last sequence number used by that node. This sequence number is attached with each packet so that the packet sequence number along with the source id uniquely determines a packet and so nodes can detect the packets forwarded by themselves. On encountering a packet which has looped, the node drops the packet and deletes the path on which it was last forwarded which resulted in the loop and informs the source of the packet that the path to the destination does not exist anymore. Caching of routes enables intermediate nodes to salvage data packets as alternate routes may be available with every node. So if the older route breaks, the intermediate node which detects the route failure first looks for an alternate route in its cache and if it finds any route, it sends the packet on that route. But it informs the source that the old route has expired so that the source can initiate route discovery to find a new route. This prevents over lengthening of routes after many routes have expired.

2.5 Link Failure in Mobile Ad-Hoc Networks

The routes between the source and destination pair in mobile ad-hoc networks frequently become unavailable due to link failure that occurs somewhere on the route. The link failure arises from many reasons, such as node mobility, power consumption, fading in the communication channel and errors in the noisy wireless medium. Thus, mobile ad-hoc networks are normally highly dynamic and the routing algorithms should be developed in order to deal with link failures effectively.

According to [30], there are two kinds of failures in MANET: node failure, and link failure. The link failure occurs more often than node failure because the node failure usually carries multiple link failures since the link will fail on a failed node. The link failure is classified into two types in terms of the number of broken links: single link failure and multiple link failure. These failures can be classified based on the recovery time from the link failure, into permanent, transient or intermittent failure as described in Wu et al. [31]. Permanent failure occurs when the failure cannot be repaired automatically within a period of time. Permanent failures, that indefinitely make the path non operational are frequently found due to the physical damages of the node, battery
depletion or long term malicious attack. In transient failure or intermittent, this type of failure occurs for short periods of time. A transient failure is more frequent than permanent failure.

This research focuses on transient (or intermittent) link failures that are caused by node mobility and which are quite common in mobile ad-hoc networks. How to deal with the failure of routing efficiently and reliably is an extremely necessary part of wireless ad-hoc routing protocol research. The link or node failures, which make the paths unreliable either because they effected the link or because the node cannot perform its job for forwarding the incoming packet, are normally experienced in mobile ad-hoc routing protocol research. These failures results in packet loss and increases the end-to-end delay in the network. Figure 2.11 shows an example of link failure in mobile ad-hoc networks.

![Example of Link Failure in MANET](image)

Figure 2.11: Example of Link Failure in MANET

### 2.6 Link State Prediction

The link state prediction is an approach used to predict the link failure in advance. There are three main prediction methods used to predict the link status for the near future as stated in Aguayo et al. [32]. These are:

- Signal strength-based methods: This method attempts to predict the link breakage
based on the received signal strength from the predecessor node as mentioned in [33, 34, 35, 36]. Each signal power keeps a minimum threshold and compares the received signal strength with this value. If the received signal is less than the threshold value, then the link to the next hop will be disconnected soon. In this aspect, the current node tries to find an alternative route to forward the data flow to the desired destination. The signal strength-based method has a good performance in any environment, especially in urban environments.

- Beacon packet-based methods: In this method, each mobile node broadcasts a Hello message to its next hop neighbors where it introduces itself to them and knows them. In some cases of the beacon packet based method, two mobile nodes receive a minimum threshold number of Hello messages from each other. In this respect, it can be concluded that they have stable state relatively. Therefore, in the route discovery process, they try to select such mobile nodes to determine the stable routes. Finding stable routes can postpone the link breakage as far as possible. This method suffers from the stale routes in the route cache as written by Heissenbuttel et al. [37]. However, to update the routing information in the route cache, this method has to reduce the broadcasting period as mobile nodes speed increases. This results in a large routing overhead in the network and this overhead may become larger than the overhead of underlying routing protocol invocation to find a new route in some scenarios.

- Position information-based methods: This method depends also on the packet received signal in order to compute its distance from the previous hop and accordingly takes an action. If the distance becomes larger than the maximum allowable threshold, then the current node considers the link breaks soon. In this respect, the new route discovery process is initiated in order to find a new route before the current one fails. This method requires a positioning system like Global Positioning System (GPS) to estimate two consecutive nodes distance. This method has good performance in open areas, whereas in urban and similar environments are not too precise. As result, the mobile nodes may fail in their prediction process as stated in [36]. However, based on the three methods above, it seems that the signal strength based method is more suitable than the two other methods in terms of suitability to urban areas as well as the power consumption. Hence, this research utilizes the first method (signal strength based method).
2.7 Literature Review

Chang J et. al. [38] and Stojmenovic I et. al. [39] have proposed power aware Ad hoc routing protocol in which main focus on minimizing the total power needed to route packets or maximizing the lifetime of all nodes. Research focus was based on individual nodes in the system instead of the system as a whole.

Sesay S et. al. [40] have proposed a combine Adaptive load energy balancing and hotspot mitigation scheme that aims at evenly distributing network traffic load and energy, mitigate against any possible occurrence of hotspot and provide some form of security to the network. This combine approach is expected to yield high reliability, availability and robustness, that best suits any dynamic and scalable Ad hoc network environment.

Bouhorma M et. al. [42] performs the comparison of AODV and DSR. Their simulated results show that DSR performs well in small networks with low node speeds and AODV performs better if mobility increases. If we want to get the optimize results then we should have to use combination of both the protocols.

Lin-zhu WANG et. al. [43]. Min have compared two routing protocols. In which DSR outperforms AODV in less “stressful” situations, i.e., smaller number of nodes and lower load and/or mobility. AODV, however, outperforms DSR in more stressful situations, with widening performance gaps with increasing stress.

Emmanouil S et. al. [41] have used DSR to route packets in Mobile Ad hoc Network. They study the management of routing data stored in nodes’ route caches by optimizing the cached route lifetime using a route Time-To-Live (TTL) parameter. The idea is to purge cache entries after some Time-to-Live (TTL) interval. If the TTL is small, valid routes are likely to be discarded prematurely, on the other hand if the TTL value is large, invalid route-caches are likely to be used, and in both cases additional routing delay and traffic overhead may result before a broken route is discovered. But there is a drawback as the transmission range becomes higher the throughput improvement drops significantly.
Biradar S.R et.al. [44] analyzed the MANET popular routing protocols DSR and AODV. The DSR performs better in high mobility, and average delay is better in case of AODV for increased numbers of groups. DSR Protocol produces higher control traffic during high mobility, due to its aggressive caching [44].

Huang Tsung-Chuan et.al. [45] proposed backup routing scheme which utilized 2-hop neighbor knowledge to establish backup paths. These backup paths are geographically close to the primary path in order to provide efficient recovery from route failure and reduce the number of route discovery procedure. But there is a problem, they propose backup routing scheme which utilized 2-hop neighbor knowledge to establish backup paths only for single link breakup. They do not propose any backup path for multiple link breakups.

Mohseni M et. al. [46] propose a new algorithm in which they uses the information location of intermediate nodes to forward the route request packets. The frequency of route breaking increased at high speeds, overhead reduction is more visible.

Mittal S et. al. [47] perform the comparison of AODV, DSR and ZRP Routing Protocols in MANET. Their results AODV is better than the other two and delivers almost 90 percent of packets.

Barakovic Sabina. et. al. [48] do the comparative performance evaluation of MANET routing protocols. Their result shows DSR is better in high mobility.

Agrawal R et. al. [49] also performance the evaluation and comparison of AODV and DSR under adversarial environment. Simulation result shows that performance of AODV is better for small number of nodes but performance degrades if number of nodes increases.

In the last years, lots of people are pushed to study the performance in wireless networks because of the increasingly use of wireless networks. Some researcher’s or users advices to change routing protocols to dynamic protocols or source routing protocol. But there are some researcher’s make a difference of the cause of packets lost or they can change
the congestion control protocol of TCP.

Table 2.1: Summary of literature review

<table>
<thead>
<tr>
<th>Author</th>
<th>Protocol</th>
<th>Drawback</th>
</tr>
</thead>
<tbody>
<tr>
<td>Chang J et. al. [38] and</td>
<td>power aware Ad hoc routing protocol</td>
<td>focus was based on individual nodes in the system instead of the system as a whole</td>
</tr>
<tr>
<td>Stojmenovic I et. al.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sesay S et.al. [40]</td>
<td>combine Adaptive load energy balancing</td>
<td>provided only for dynamic system, not concern with static system.</td>
</tr>
<tr>
<td></td>
<td>and hotspot mitigation scheme</td>
<td></td>
</tr>
<tr>
<td>Huang Tsung-Chuan et.al.</td>
<td>proposed backup routing scheme which</td>
<td>propose backup routing scheme only for single link breakup not for multiple link breakups.</td>
</tr>
<tr>
<td>[45]</td>
<td>utilized 2-hop neighbor knowledge to</td>
<td></td>
</tr>
<tr>
<td></td>
<td>establish backup paths</td>
<td></td>
</tr>
<tr>
<td>C.David,G.Alessandro[66]</td>
<td>proposed secure A-SAODV</td>
<td>consider only some improvements</td>
</tr>
<tr>
<td>W.Zhiguo,R.Kim et.al[67]</td>
<td>Proposed unobservable secure on-demand</td>
<td>warmhole attacks cannot be prevented with USOR</td>
</tr>
<tr>
<td></td>
<td>on-demand routing (USOR) protocol</td>
<td></td>
</tr>
<tr>
<td>W.JIANXI[68]</td>
<td>AODV based anycast protocol</td>
<td>not efficient if traffic load is unevenly distributed</td>
</tr>
</tbody>
</table>