IOSR Journal of Computer Engineering (IOSR-JCE)
e-ISSN: 2278-0661, p-ISSN: 2278-8727, Volume 27, Issue 3, Ser. 4 (May. – June. 2025), PP 27-40
www.iosrjournals.org

Designing Cooperative Communication Protocols for Improving Reliability in Wireless Ad Hoc Networks

DANIEL IDDRISU[1], DR. PRINCE CLEMENT ADDO[2], DR. ADASA KOFI NKRUMAH FRIMPONG[3]

(Department of Mathematics and ICT, McCoy College Of Education, Ghana)
(Department of Information Technology Education, Akenten Appiah-Menka University of Skills Training and Entrepreneurial Development, Ghana)
(Department of Information Technology Education, Akenten Appiah-Menka University of Skills Training and Entrepreneurial Development, Ghana)

Abstract

Ad-hoc networking is the process by which users create a temporary network without the supervision of a central administrator. Each node in the network functions as both a host and a router, therefore it must be willing to send packets to other nodes. A routing protocol is necessary to accomplish this task. Ad hoc networks have unique properties that throw additional demands on the routing system. The most important property is the dynamic topology produced by node mobility. Because nodes move so frequently, routing systems must respond fast to topology changes. Ad-hoc networks often use laptops and personal digital assistants with limited resources such as CPU, storage, battery life, and bandwidth. The routing protocol should minimize control traffic, which includes periodic update messages. The routing protocol should be reactive, determining routes only after receiving specific requests. Simulations show that a specific ad-hoc routing technique will be required when mobility grows. When there is a high degree of mobility, classical routing protocols like DSDV perform much poorer. The proposed protocols include DSR and AODV. They thrive in high-mobility settings. However, relying entirely on IP-level communications for routing does not provide optimal performance. Lower-layer support, such as connection failure detection and neighbor search, is required for high performance. DSR, as well as other source routing algorithms, are affected by network size and traffic demand. A large network with many mobile nodes and a high load considerably increases DSR's overhead. In these instances, a hop-by-hop routing system, such as AODV, is preferred.

Key words: Wireless Ad Hoc Networks, Mobility, AODV, DSDV, DSR, Protocols

---

Date of Submission: 12-06-2025 Date of Acceptance: 25-06-2025

I. INTRODUCTION

Wireless Ad Hoc Networks (WANETs) enable decentralized, self-configuring communication without relying on fixed infrastructure like base stations or routers. Each node is a host and a router, facilitating peer-to-peer communication. WANETs typically operate in clusters of mobile devices, such as laptops or embedded systems that communicate directly or act as repeaters and gateways (Bhandari, 2024). Unlike traditional networks, WANETs can be deployed in areas lacking infrastructure, making them ideal for disaster recovery, military operations, conferences, and mobile sensor networks (Kohlstruck, 2023). Advances in wireless LANs, modems, and portable devices have further driven their adoption (Veeraiah et al., 2021). WANETs are characterized by dynamic topology, as nodes frequently move, join, or leave the network. This demands constant route discovery and maintenance, relying on cooperative communication to ensure packet delivery. WANETs face bandwidth limitations, energy constraints, and mobility-induced disruptions despite their adaptability. Reliability is critical, given the risks of link failures, packet loss, and delays. Interference, congestion, and malicious activities exacerbate these issues (Tyagi et al., 2021). WANET routing objectives include improved packet delivery, prolonged network lifetime, and reduced delays (Unnikrishnan & Das, 2022). Protocols like Ad Hoc On-Demand Distance Vector (AODV) and Dynamic Source Routing (DSR) aid route discovery, while energy-efficient methods enhance network longevity. As WANET applications expand to IoT, vehicular ad hoc networks (VANETs), and smart cities, they demand robust communication protocols to support large-scale, dynamic environments. Cooperative communication protocols, which leverage surrounding nodes for reliability and efficiency, are promising solutions. These protocols allow intermediate nodes to act as relays, improving packet forwarding and overall stability. However, decentralization introduces challenges, such as establishing trust and ensuring uniform standards among nodes (Choudhury et al., 2008). This work focuses on developing cooperative communication protocols to address WANET reliability issues. By using cooperative diversity, dynamic relay selection, and intelligent routing algorithms, the proposed solutions aim to enhance reliability and efficiency in practical deployments. Although WANETs lack a standard routing protocol, ongoing research addresses current challenges and evaluates emerging protocols for reliable and scalable implementations (Nirmaladevi & Prabha, 2023).

1. 1 Problem Statement

WANETs have a dynamic topology due to node mobility and frequent structural changes, leading to unstable and transient connectivity. This results in frequent route breaks, requiring constant route discovery and maintenance, which increases control overhead and depletes network resources (Han, 2019). Shared wireless channels and limited bandwidth exacerbate interference and signal fading, raising error rates and reducing data delivery reliability (McDonald & Znati, 1999). Routing protocols like AODV, DSR, and OLSR focus on rapid route discovery but often neglect reliability. While basic cooperative communication solutions exist, comprehensive protocols addressing reliability across multiple layers (physical, MAC, and network) remain underdeveloped. This study evaluates WANET routing methods for reliability through conceptual analysis and simulation, comparing them with traditional wired network protocols.

1.2 Objectives

1. Design and evaluate routing protocols that can adjust dynamically to the high node mobility and regular topological shifts found in wireless ad hoc networks (WANETs).
2. Evaluate how well these protocols work in traffic and mobility circumstances.

1.3 Related Work

Few comparisons exist among proposed routing protocols for WANETs. The Monarch project at Carnegie Mellon University (CMU) evaluated various protocols using consistent metrics, publishing results in "A Performance Comparison of Multi-Hop Ad Hoc Wireless Network Routing Protocols". Other simulations exist but lack comparability due to differing metrics. Concurrently, a Gothenburg thesis studied AODV with five wireless PCs, enabling idea-sharing between the two projects.

II. GENERAL CONCEPTS

2.1 WIRELESS AD-HOC NETWORKS

A wireless ad-hoc network comprises mobile and semi-mobile nodes forming a temporary network without pre-existing infrastructure (Azgin et al., 2005). Each node, equipped with a wireless interface, communicates via radio or infrared (Cardei et al., 2006). Ad-hoc nodes include laptops, personal digital assistants, and stationary nodes like Internet access points (Adhikari & Setua, 2014). Semi-mobile nodes can serve as relay points in areas requiring enhanced connectivity. To improve network reach, intermediary nodes act as relays between sources and distant receivers (Marbach & Qiu, 2005). Ad-hoc networks are rapidly deployable, self-organizing, and function without fixed infrastructure. These wireless nodes collaborate to establish on-the-fly communication with minimal management (McDonald & Znati, 1999). They lack dedicated routing architecture, and transmission is influenced by channel effects, power constraints, and frequency reuse (Scaglione et al., 2006). Store-and-forward packet routing is essential for multi-hop paths, but routing becomes challenging due to mobile and independent endpoints (Azgin et al., 2005). Resource limitations and dynamic topology further demand efficiency in routing. Traditional algorithms designed for infrastructure networks perform poorly under dynamic conditions due to high overhead and limited adaptability (Cardei et al., 2006; McDonald & Znati, 1999). Wireless ad-hoc networks rely on nodes with transceivers, functioning as both routers and endpoints (Dipobagio, 2009). For example, in a basic three-node network, a central node can act as a router, enabling communication between two distant nodes.

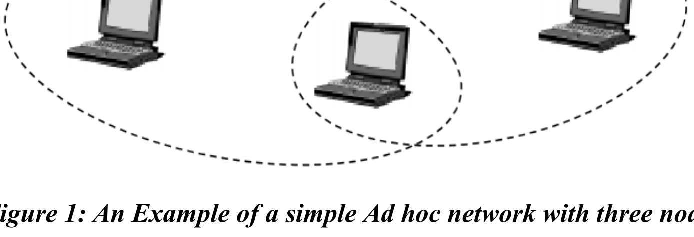

Figure 1: An Example of a simple Ad hoc network with three nodes

Figure 2 illustrates how the nodes form a transmission cloud. N1 desires to speak with N5. Since N5 is beyond N1's transmission range, N1 must forward the message to either N2-N3-N5 or N4-N2-N3-N5. The routing algorithm will determine the optimal path. Since N1 still has a path to N5, there won't be any issues if N4 exits the network.

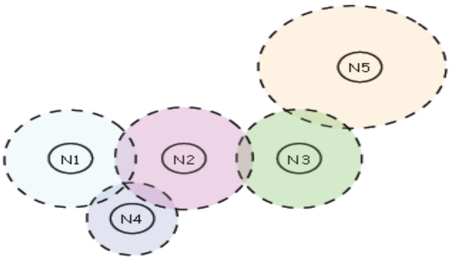

Figure 2: An Example of an Ad hoc network with more than three nodes

Ad hoc networks are therefore more reliable than infrastructure. This is to ensure that the network won't crash because a single mobile node goes out of the other nodes' transmission area. Nodes ought to have unrestricted access to get into and out of the network. It could take several hops to reach other nodes because of the nodes' constrained transmitter range. Any node that wants to be a part of an ad hoc network has to be prepared to send packets on behalf of other nodes. Every node thus serves as a router in addition to a host. A router and a group of connected mobile hosts make up a node, which can be thought of as an abstract object (Figure 3).

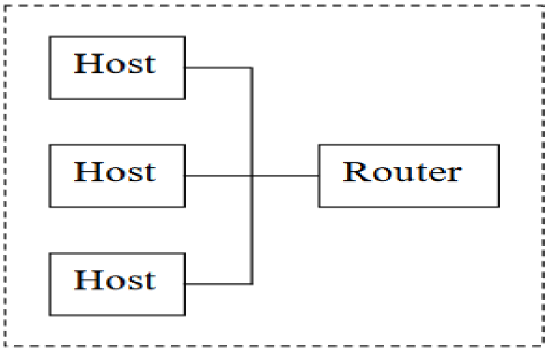

Figure 3: Block diagram of a mobile node operating as both host and router

A router is an entity that manages a routing protocol among other things. All an IP-addressable host or entity in the conventional sense, is a mobile host. Ad hoc networks have seen a sharp rise in popularity in recent years due to their support for network mobility and freedom. Without a cable, access point, or portable memory space, data can be shared (Dipobagio, 2009). Ad-hoc networks can additionally manage node breakdowns and changes in topology. Re-configuring the network resolves the issue. For example, if a node disconnects from the network and breaks links, the impacted nodes can simply request new routes, resolving the issue. The network will still function even if this will cause a little delay. Ad-hoc wireless networks capitalize on the characteristics of wireless communication. Stated differently, in a wired network, the actual cabling is completed beforehand, hence limiting the nodes' connection topology. In the wireless realm, this constraint does not exist, and an immediate link can develop between two nodes as long as they are within transmitter range of one another.

2.2 Cooperative Communication

In cooperative communications, a relay channel creates independent pathways between the user and the base station, supplementing the direct channel (Library & Core, 2019). The relay processes signals received from the source node, with different processing methods defining various cooperative protocols. These fall into fixed relaying and adaptive relaying categories (Perkins & Royer, 1999). Fixed relaying allocates channel resources deterministically between the source and relay (Choudhury et al., 2008). In amplify-and-forward (AF) protocols, the relay scales and forwards the received signal. In decode-and-forward (DF) protocols, the relay decodes, re-encodes, and transmits the signal (Bisnik, 2005). Fixed relaying is easy to implement but has low bandwidth efficiency, as the relay uses half of the channel resources, potentially wasting transmissions if the source-destination link is strong (Library & Core, 2019). Adaptive relaying strategies, such as selective and incremental relaying, address this inefficiency by dynamically adjusting relay operations based on channel conditions.

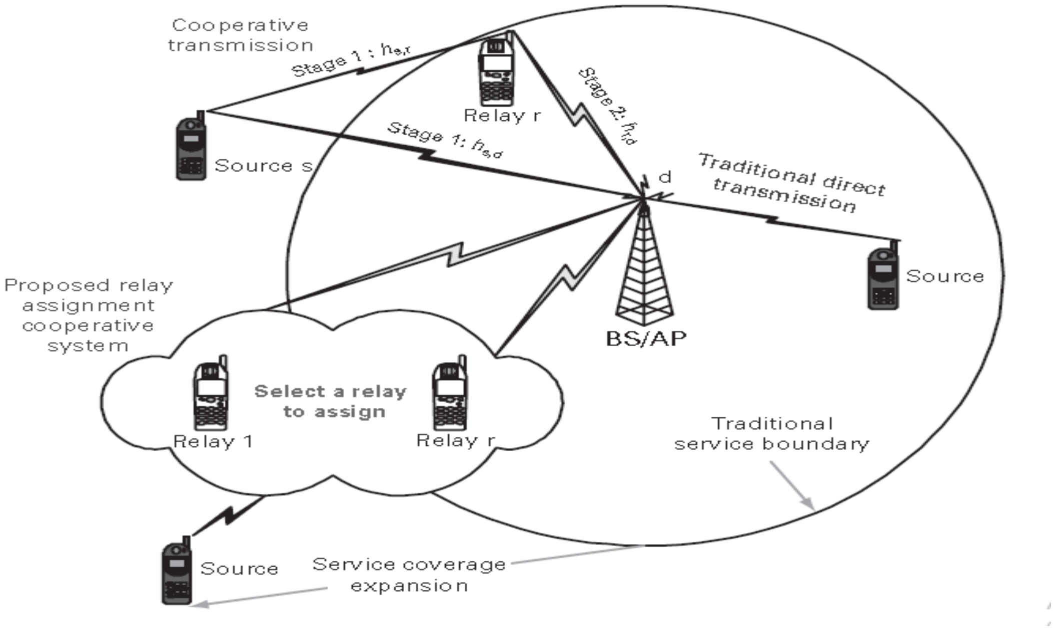

Figure 4: demonstrating the difference between direct and cooperative transmission techniques, as well as the coverage extension anticipated by cooperative transmission.

In selective relaying, the relay decodes and forwards the message if the signal-to-noise ratio of the received signal exceeds a predefined threshold. In contrast, the relay idles if there is sufficient fading in the channel between the source and the relay, resulting in a signal-to-noise ratio less than the threshold. Furthermore, suppose the source knows that the destination does not decode correctly. In that case, it may repeat transmissions of information to the destination, or it may employ incremental relaying, in which case the relay aids in the delivery of information. In this case, a feedback route between the destination, the source, and the relay is necessary.

2.3 Usage

Ad hoc networks have diverse applications, including IoT deployments where devices communicate without centralized infrastructure. This supports industrial automation, smart cities, smart homes, and environmental monitoring (Blakeway, 2015). These networks are also useful in areas without Internet access, such as military operations or disaster recovery, where traditional infrastructure is damaged or unavailable (Marbach & Qiu, 2005). In disaster management, ad hoc networks enable rapid communication for first responders in areas impacted by natural disasters like hurricanes, floods, or earthquakes (McDonald & Znati, 1999). Other uses include students communicating during lectures or business associates sharing files in airports. With wireless interfaces, mobile hosts can form ad hoc networks for peer-to-peer communication and access to network resources, such as printers or Internet connectivity (Han, 2019).

2.4 Characteristics

Nodes in ad hoc networks frequently move, resulting in a dynamic topology (Ali, 2011). Routing protocols that find routes dynamically are favored over traditional algorithms like link state and distance vector. These nodes, often "thin clients," have limited CPU, storage, battery, and bandwidth, requiring constrained emitter ranges to conserve power (Dipobagio, 2009). While routing must adapt to topology changes, efficiency is critical due to scarce and unstable network resources (Library & Core, 2019). Conventional routing algorithms perform poorly and incur high costs in highly dynamic topologies (Cadger et al., 2013). Ad hoc protocols must account for unique radio environment characteristics, such as unidirectional links caused by differing transmitter strengths or external disruptions (Unnikrishnan & Das, 2022). Multi-hop routing improves transmit capacity and reduces power usage, as nodes transmit packets with less output power by leveraging intermediate nodes (Kohlstruck, 2023).

2.5 Routing

A routing protocol is required since a packet may need to hop through multiple hops (multi-hop) before reaching its destination. Selecting routes for different source-destination combinations and ensuring that messages reach their intended recipient accurately are the two primary tasks of the routing protocol (Kohlstruck, 2023). A range of protocols and data structures, including routing tables, are used in the conceptually simple second function. Route selection and discovery are the main topics of this work.

2.5.1 Conventional Protocols

In computer networking, "conventional protocol" refers to widely used, well-tested communication protocols adhering to established standards for data transfer (Guan et al., 2012). These protocols, such as link state and distance vector, are industry standards and have been extensively applied in various networking scenarios. However, their suitability for ad hoc networks is limited due to the dynamic nature of these networks. Traditional protocols like link state and distance vector were designed for static topologies, where network changes are rare and convergence to a stable state is feasible (Almotairi & Shen, 2015). These protocols struggle to adapt and converge efficiently in ad hoc networks with frequent topology changes. They may work better in low-mobility ad hoc networks where topology changes are infrequent (Nandan et al., 2004). A significant limitation of these protocols in ad hoc networks is their reliance on periodic control messages. Maintaining routes to all reachable destinations involves frequent data exchanges between nodes. This behavior conflicts with the resource constraints of wireless ad hoc networks, where bandwidth, battery life, and CPU power are limited, and all communications are airborne, making updates costly. Additionally, conventional protocols assume bidirectional links, where communication between two hosts is equally effective in both directions. This assumption does not hold in wireless radio environments, where unidirectional links can occur due to differences in transmitter power or external interference. Understanding the principles of traditional protocols like link state, distance vector, and source routing is essential, as many proposed ad hoc routing protocols build upon these classic foundations. However, designing efficient protocols for ad hoc networks requires addressing wireless communication's dynamic topology, resource constraints, and unique characteristics.

2.5.2 Link State

In link-state routing, each node collects data about its directly connected links and shares it with all network nodes, creating a "link-state database" (Almotairi & Shen, 2015). Nodes maintain a full topology map, periodically broadcasting outbound link costs using flooding. Upon receiving this data, nodes update their network view and calculate next-hop destinations via the shortest path algorithm. Temporary routing loops may occur due to partitioned networks or propagation delays but typically resolve within the time it takes for a message to traverse the network (Khalid et al., 2011).

2.5.3 Distance Vector

In distance vector routing, each node tracks the cost of its outgoing links and periodically broadcasts the shortest path estimates to neighbors (Unnikrishnan & Das, 2022). Nodes update routing tables using this data. While distance vector is simpler, more efficient, and requires less storage than link-state, it can lead to transient or permanent routing loops due to distributed next-hop decisions based on outdated information (Guan et al., 2012).

2.5.4 Source Routing

According to source routing, every packet must contain the entire path it should follow over the network (Srinivasan et al., 2005). Thus, the choice of routing is decided upon at the source. This method has the benefit of being quite simple to eliminate routing loops. The drawback is that there is a small overhead required for each packet.

2.5.5 Flooding

Broadcasting is a common method used by routing systems to communicate control information from an origin node to every other node (Kohlstruck, 2023). Flooding is a popular broadcasting technique that works as follows. The origin node communicates with its neighbors, or all nodes within transmitter range in the case of a wireless connection. The packet travels through the neighbors' neighbors and so on until it reaches every single network node. A sequence number can be used to guarantee that a node only relays a packet once. With every new packet a node sends, this sequence number is raised (Moh & Yu, 2012).

2.5.6 Classification

Routing protocols can be categorized based on their characteristics:

- **Reactive vs. Proactive:** Proactive protocols continuously evaluate network routes, ensuring they are ready for immediate use when a packet needs forwarding. The Distance-Vector protocol family is an example. Reactive protocols, like flooding algorithms, determine routes only when needed, often using a global search process. Proactive methods minimize delays in forwarding but require time to stabilize, which can be an issue in highly dynamic topologies (Adhikari & Setua, 2014).
- **Centralized vs. Distributed:** In centralized algorithms, a single node makes all routing decisions, while distributed algorithms rely on collaboration among network nodes to compute routes (Wang et al., 2014).
- **Static vs. Adaptive:** Static protocols maintain fixed routes for source-destination pairs, only adjusting for node or link failures. This limits their performance under varying traffic patterns. Adaptive protocols, commonly used in large packet networks, adjust paths dynamically based on traffic congestion, enabling higher throughput.

III. MANET

A mobile ad hoc network, or MANET for short, is a kind of wireless ad hoc network in which mobile nodes connect independently of centralized management or fixed infrastructure. With the help of MANETs, temporary networks may be created on-the-fly and nodes can communicate with one another as they move throughout the network. MANETs facilitate node mobility. As long as the target is accessible we can still converse with our handheld devices (Dipobagio, 2009). The main goal of a MANET is to enable the rapid deployment of a data communications network in situations where an infrastructure is either non-existent, restricted, or lacks predefined components, or where using the existing infrastructure may raise security concerns (Blakeway, 2015). In situations where there is no preset infrastructure in place or if infrastructure has been damaged by malicious intent, natural disasters like earthquakes or floods, attacks, intentional sabotage, or explosions, a MANET would enable the development of a communications network. It may additionally be applied when maintaining the current infrastructure's security is a worry, especially in a combat zone (Blakeway, 2015).

3.1 Destination Sequenced Distance Vector – DSDV

The proactive routing protocol known as Destination-Sequenced Distance Vector (DSDV) is employed in wireless ad hoc networks. The goal of its development was to overcome the difficulties presented by the dynamic and movable nature of ad hoc networks by improving upon the traditional distance-vector routing algorithm. Based on the Distributed Bellman-Ford algorithm, DSDV uses a distance vector routing approach. This routing protocol behaves horribly in networks with changing topologies. The count-to-infinity issue exists in this protocol. The nodes must continuously exchange their routing tables to obtain information about the actual topology (Dipobagio, 2009).

3.1.1 Description

DSDV is a hop-by-hop distance vector routing protocol where each node maintains a routing table with the next hop and total hops for every reachable destination. Like traditional distance-vector protocols, DSDV periodically disseminates routing information. However, DSDV ensures loop freedom by tagging routes with sequence numbers, indicating route freshness. Routes with higher sequence numbers or fewer hops (if sequence numbers are equal) are preferred. When a route to a destination breaks, the node increases the sequence number and advertises it with an infinite hop count. DSDV modifies distance-vector protocols for ad hoc networks by introducing triggered updates to handle topology changes between broadcasts. Two update types are used: full dumps (containing all routing information) and incremental dumps (only changes since the last dump), reducing data in update packets.

3.2 Ad-Hoc on Demand Distance Vector – AODV

When mobile nodes want to create and manage an ad hoc network, they can do so by using the Ad Hoc On-Demand Distance Vector (AODV) routing protocol, which allows multi-hop routing. Distance vector algorithms serve as the foundation for AODV. AODV differs from proactive protocols like DSDV in that it is reactive, meaning nodes do not have to maintain routes to destinations that are not being used for communications; instead, AODV only requests a route when necessary. AODV is irrelevant as long as there are legitimate routes connecting the ends of a communication link. A traditional distance vector routing algorithm is used by AODV.

Additionally, it distributes the routes that DSR finds on demand. To provide loop-free paths, AODV is used to repair link breaks. As long as there is a route that connects the source and the destination, it does not increase packet overhead. This lessens the consequences of stale routes and the requirement for underused routes to have their routes maintained. AODV's ability to facilitate broadcast, unicast, and multicast communication is among its best characteristics. A broadcast is used by AODV's route discovery method, while a unicast response is used for replies (Ali, 2011).

3.3 Dynamic Source Routing (DSR)

Within the class of reactive protocols is Dynamic Source Routing (DSR), which enables nodes to dynamically find a path to any destination via a series of network hops. When a packet is routed by source routing, the entire ordered list of nodes that it has to travel through is contained in the packet's header. By not using periodic routing messages (such as router ads), DSR saves battery life, lowers network bandwidth overhead, and prevents significant routing adjustments across the ad-hoc network. Rather, DSR depends on MAC layer support, which should notify the routing protocol of link failures. Route discovery and route maintenance are the two fundamental DSR operating modes.

3.4 Comparison

The protocols have up to this point only undergone theoretical analysis. Table 1 presents a summary and comparison of the findings from these qualitative and theoretical assessments, outlining the characteristics that the protocols possess and lack.

	DSDV	AODV	DSR
Requires reliable or sequential data	No	No	No
Periodic broadcasts	Yes	Yes	No
Power conservation	No	No	No
Security	No	No	No
Multicast	No	Yes	No
QoS support	No	No	No
Unidirectional link support	No	No	Yes
Reactive	No	Yes	Yes
Loop-free	Yes	Yes	Yes
Multiple routes	No	No	Yes
Distributed	Yes	Yes	Yes

Table 1: A Comparative Analysis of Ad Hoc Routing Protocols

Table 1 highlights that none of the protocols currently support quality of service or power conservation, though these features may be added in future updates. All protocols are distributed, avoiding reliance on centralized nodes and adapting well to topology changes. DSDV, the only proactive protocol, closely resembles traditional wired routing systems and uses sequence numbers to ensure loop-free paths. While effective in stable networks, frequent movement can hinder its performance. AODV, a reactive version of DSDV, adds multicast capabilities, improving performance for communication with multiple nodes. DSR, another reactive protocol, shares similarities with AODV, including route discovery through request messages. However, DSR uses source routing, enabling it to learn more routes and support unidirectional links. A drawback is the added overhead of carrying source paths in each packet, making it less suitable for QoS. Zone-based protocols combine proactive routing within zones and reactive routing between zones, resembling DSR and AODV. None of the protocols adapt to traffic load, often routing packets through the shortest or quickest paths, potentially overloading certain nodes.

IV. SIMULATION STUDY

The DSDV, AODV, and DSR protocols were the ones we simulated. The purpose of DSDV is to compare the relative improvements or decreases in performance between the MANET protocols and a standard proactive protocol.

4.1 Mobility

Mobility is a key parameter for evaluating ad hoc networks and can be defined in various ways. For instance, the CMU Monarch project used pause time at waypoints, where low pause times indicate high mobility and high pause times indicate low mobility. However, this approach is insufficient, as nodes moving slowly in the same direction would still appear highly mobile. We define mobility based on relative movements between nodes, offering a clearer picture of their interactions. Mobility is calculated as the average change in distance between nodes over time TT. It depends on the movement pattern and speed and is measured using a specific sampling rate. In our simulations, a 0.1-second sample rate was used, aligning with the default logging interval. Table 2 lists the variables used in the mobility calculation.

Variable name	**Description**
Dist(n_x, n_γ)$_t$	Distance between nodes x and y at time t
n	The number of nodes
i	Index
A_x(t)	Average distance between node x and all other nodes at time t
M_x	Node x's average mobility compared to all other nodes over time.
T	The simulation time
Δt	Granularity, simulation step
Mob	The mobility for entire scenario

Table 2: Mobility Variables

The average distance between each node and every other node must first be determined. For times t = 0, t = 0+X, t = 0+2X... t = simulation time, this needs to be completed. The formula for the node x at time t is:

A_x (t) $= \sum_{i-1}^{n} Dist(n_{x,}n_i)$ (5.1)

n-1

Next, the average mobility for that specific node needs to be computed using (5.1). This represents the mean variation in distance durin~~g the simulation. For node~~ x, the mobility is:

(5.2)

$$M_{x=} \sum_{t=0}^{T-\Delta t} \frac{|\left(A_{x\,(t)-A_{x\,(t+\Delta t)}}\right)|}{T-\Delta t}$$

Lastly, the total mobility for all nodes (5.2) divided by the total number of nodes equals the mobility for the entire scenario:

(5.3)

$$\frac{Mob_{=} \sum_{i=1}^{n} Mi}{n}$$

First and foremost, one of the key elements of an ad hoc network is its mobility, which is why it was chosen as a simulation parameter. Furthermore, individuals generally find it easy to understand the concept of mobility. Everyone's understanding of what happens if mobility is improved is rather good.

We have examined the impact of the mobility factor on the dynamic topology through testing. The number of link changes is directly correlated with the mobility factor. In essence, a link change occurs when a connection goes from an up/down state to a down/up state. The plot displays the average. Results of all the simulations we ran with 50 nodes and a 1000 × 1000 meter setting.

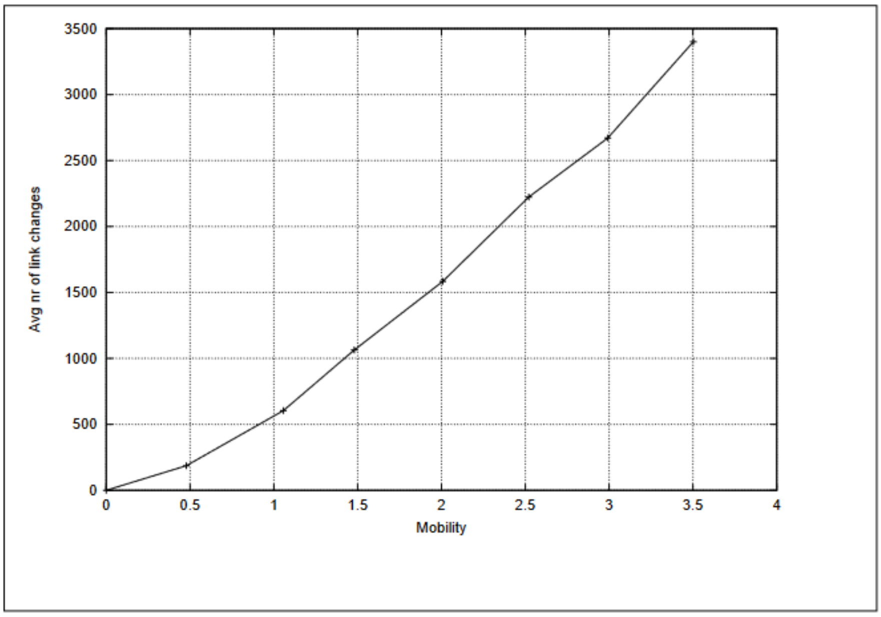

Figure 5: The Relationship between the Number of Link Changes and Mobility

4.2 Mobility Simulation

4.2.1 Setup

When we changed the mobility, we used randomization of scenario files in the simulations. Because we cannot specify in advance during scenario generation that we want a mobility factor of precisely X, this strategy is particularly difficult to implement. Instead, we managed the scenario by using the maximum speed option. Table 3 displays the simulation parameters that were applied to the mobility simulations.

Parameter	Value
Number of flows	15
Packet size	64 byte
Packet rate	5 packets/s
Traffic rate	Constant Bit Rate
Environment size	1000x1000 m
Pause time	1 s
Number of nodes	50
Simulation time	250 s
Transmitter range	250 m
Bandwidth	2 Mbit

Table 3: Parameters Used During Simulation

The simulation scenario is crucial, so we collected 10 measurements for each mobility factor: 0.5, 1.0, 1.5, 2.0, 2.5, 3.0, and 3.5, with intervals set at 0.1. Scenarios with high percentages of unreachable hosts were excluded, as network partition was not a focus. Mobility increases with the scenario's maximum speed, where a speed of 20 m/s corresponds to a mobility factor of 3.5. Randomized simulations used speeds ranging from 0 to 20 m/s, representing high mobility. Each mobility simulation followed the same communication pattern with 15 CBR sources starting at different times. TCP was excluded to avoid studying features like flow control and retransmission, focusing instead on routing protocol behavior. Communication patterns were randomized, setting parameters like source count, packet size, transmission rate, and duration. A modest load was used to examine mobility effects, transmitting 64-byte packets at 5 packets per second over links with 2 Mbit bandwidth.

4.3.2 Fraction of Received Packets

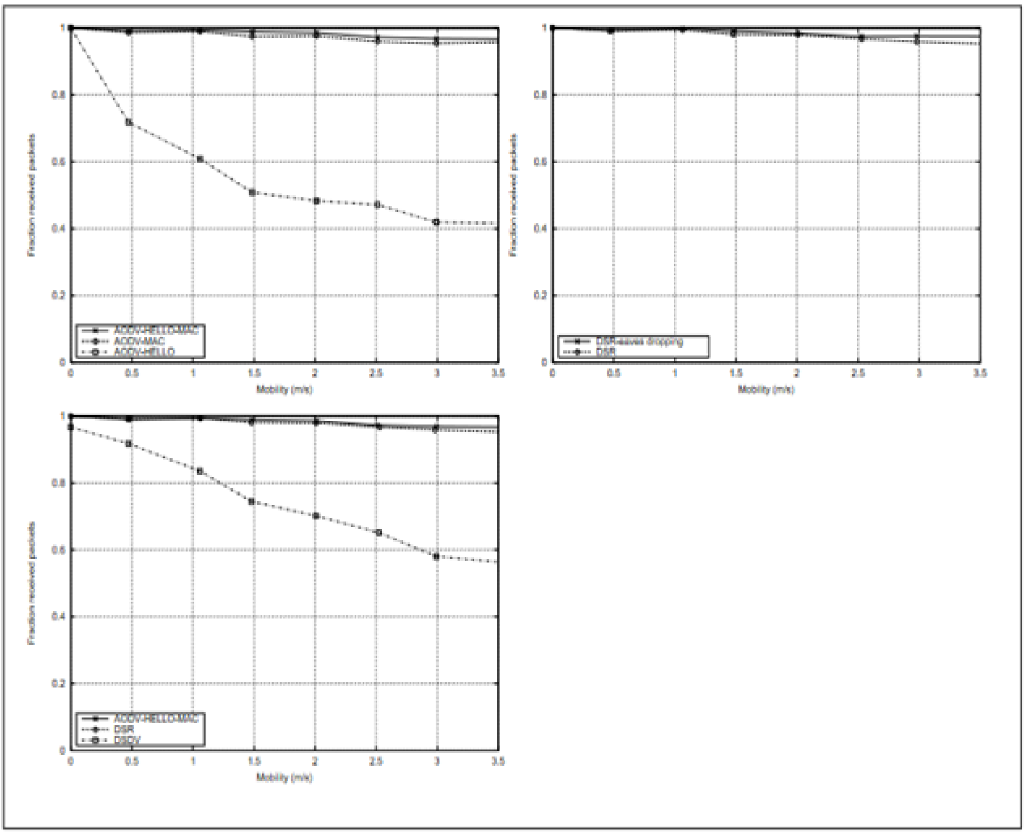

Figure 6: Mobility Simulation – Fraction of Received Packets

What percentage of data packets that are sent are received, and why are the packets that are dropped deleted? Figure 6 illustrates that, out of the several AODV versions, the two that support the MAC layer receive nearly all of the packets that are delivered. The version of AODV that supports both the MAC layer and hello messages is marginally superior to the version that only supports the MAC layer. The same explanation as before explains this: the hello messages receive some advance notice about link failures. However, as mobility grows, AODV with merely hello messages is losing a significant fraction of the packets. Naturally, a large fraction of missed packets is unacceptable, and the hello message interval is the cause of this loss. To detect link breakages, the time gap between hello messages and the total number of permitted hello message losses are essential. Reduced intervals allow for the early detection of connection failures but can increase network control overhead. Trying to determine the best values for these parameters is the problem at hand. The intended behavior—a higher percentage of received packets, a high throughput, a short delay, or a low overhead—influences the choice of these parameters as well.

Even with great mobility, a significant portion of the packets for the DSR versions are received. The percentage of received packets is marginally lower in the DSR version that does not use eavesdropping. Nonetheless, this difference is so slight as to be insignificant. For the simple reason that it has a bit more information while computing the routes, DSR with eavesdropping produces superior results. The fact that DSR enables packets to remain in the send buffer for up to 30 seconds whereas AODV only allows for 8 seconds (in our implementation) may be the cause of the greater fraction of received packets for DSR when compared to AODV. However, it should be noted that the length of time a packet is permitted to remain in the send buffer is not specified in the AODV draft.

It is evident from a comparison of these results with the DSDV results that a proactive strategy is completely unacceptable as mobility rises. The percentage of packets received drops sharply to 56–57 percent. Nonetheless, this figure is for a very high mobility factor (vehicles). However, as with all other protocols, the percentage of packets received is not even 100% when the mobility is 0. This occurs because packets are dropped. After all, they are transmitted before the routing tables have had a chance to converge.

The protocol discarding packets is mostly caused by two factors: congestion and timeouts in the buffers, and the protocol delivering packets on broken routes it believes to be genuine. Only a small portion of the packets have been discarded due to collisions under this low load. It is also evident that using IP-based greeting messages as the sole method of link failure detection is not a smart concept. Even DSDV just marginally outperforms in terms of results. When a link breaks, the upper layer routing protocol is notified by the link layer feedback, which enables it to respond considerably faster.

4.3.3 End-to-End Delay

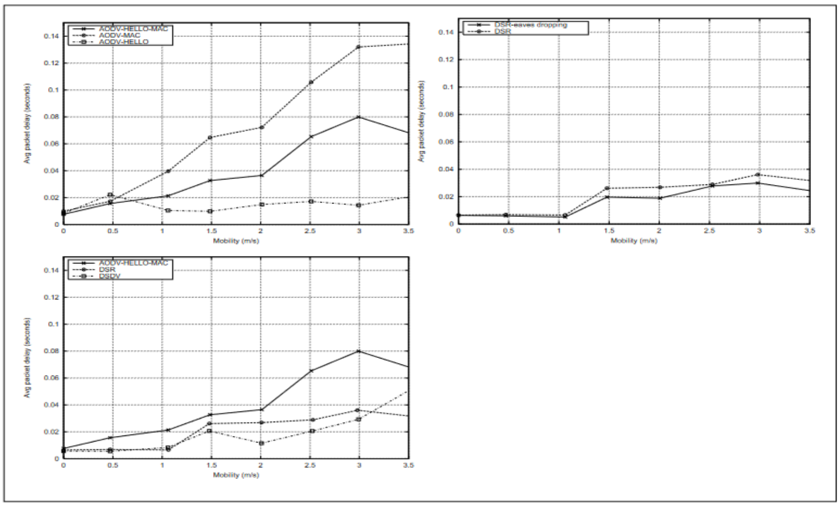

Figure 7: Mobility Simulation-Delay

Figure 7 illustrates that among the various AODV versions, the AODV that solely transmits greeting messages experiences the least latency in receiving data packets. This is not because it identifies routes more quickly, or because the routes are shorter or more ideal; rather, the AODV version that sends out only hello messages is the one that moves the fewest packets across the network. It successfully traverses the network with packets around the same latency as the other AODV versions. The other AODV versions differ in that some of the packets have a bigger delay (having been in a buffer for a long period but still being sent over the network).

Long-lived packets in the buffers are discarded in AODV when hello messages are all that are sent. The cause is that broken links are not identified quickly enough by ADOV, which makes it possible for a source to continue transmitting packets on a broken connection under the mistaken impression that it is operational. The latency of AODV with both MAC layer support and welcome messages is somewhat less than that of AODV with only MAC layer support. This is because, as was previously explained, AODV, which only supports the MAC layer, renders the protocol entirely on-demand; it only identifies connection failures when packets are being sent. Because they are buffered while waiting for a new route to be found, packets sent after this breakage is detected will arrive with a larger latency.

On the other hand, AODV which has both Mac-layer support and hello messages will be aware of the link breakage ahead of time and will have an opportunity to choose a new route before any more packets are delivered. When mobility is increased, both DSR versions tend to get higher delays. At about a 1.0 mobility factor, the tipping point occurs. When comparing DSR with eavesdropping to DSR without, the latter has a noticeably longer latency. According to these findings, the protocol with the least amount of latency is DSDV. However, because DSDV drops so many packets that it cannot be considered legitimate, the results are somewhat misleading. When utilizing DSR, for example, the packets missed in DSDV will pass through, but with a little increased delay due to buffering durations and other factors. The average delay for DSR will increase because of these higher delay packets. With simple greeting messages, AODV can be regarded to be in the same boat. The reason for the slight delay between the other two AODV versions and the DSR versions is likely related to the source routing concept of DSR.

Since DSR learns so much from source routes, it will be able to learn routes to a much greater number of destinations than a distance vector protocol like AODV. This means that, even though DSR already knows a route for a particular destination, AODV will need to send an individual request for that destination, meaning that packets will remain in a buffer until a valid route is found, which will take some time and raise the average delay. The packet delays in a packet-based radio network will differ significantly in the absence of quality of service. Until a route is found, the packets without a route will be buffered. How long a packet should be allowed to remain in the buffer before being discarded is a crucial consideration in this situation.

The following scenario may arise if the packets are left in the buffer for an extended period: A packet is sent, but it is buffered and a route request is made because there isn't a route to that location. Nevertheless, no route reply is sent back to the transmitting node because the destination node cannot be reached. The packet is transmitted when, after a considerable amount of time, the destination node is suddenly reachable. There will be a significant delay for this packet. Is it appropriate to allow this to continue? Do we want every packet to get over the network, even though there may be a significant delay, or should the packet be removed from the buffer much sooner? Since no acknowledgment was received in the case of TCP, the retransmit process will most likely retransmit the packet sooner nonetheless. In DSR, packets can remain in the send buffer for up to 30 seconds, whereas in AODV, the maximum time is only 8 seconds. A packet received thirty seconds after it was transmitted will result in a somewhat longer average delay

4.3.4 End-to-End Throughput

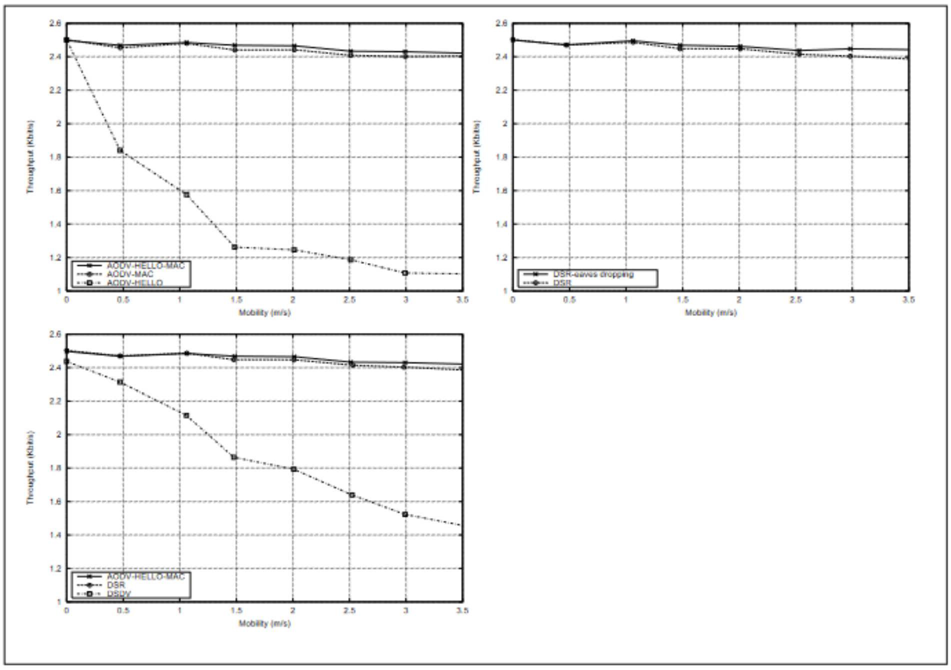

Figure 8: Mobility Simulations–Throughput

The throughput curves for the various protocols with 64-byte packet sizes are displayed in Figure 9. However, it should be noted that the curves in this instance are only noteworthy when viewed in the context of a relative comparison of the protocols. We have merely attempted to ascertain the relative differences in throughput for the various protocols about the mobility factor and the particular load that we have employed; we have not attempted to maximize throughput. All of the protocols' throughput curves resemble the fraction received packet curves in a fairly similar way. This makes sense because higher packet drops will inevitably result in decreased throughput.

The throughput of the DSR and link layer-supported AODV versions is nearly the same. Similarly, roughly constant, its throughput starts to decline with mobility levels as high as 2.5–3.5. The throughput of DSDV and AODV with merely greeting messages drops sharply as mobility rises. AODV with just hello yields a pretty subpar outcome. When mobility is zero, the throughput curve quite instantly falls to half of its original value.

4.3.5 Overhead

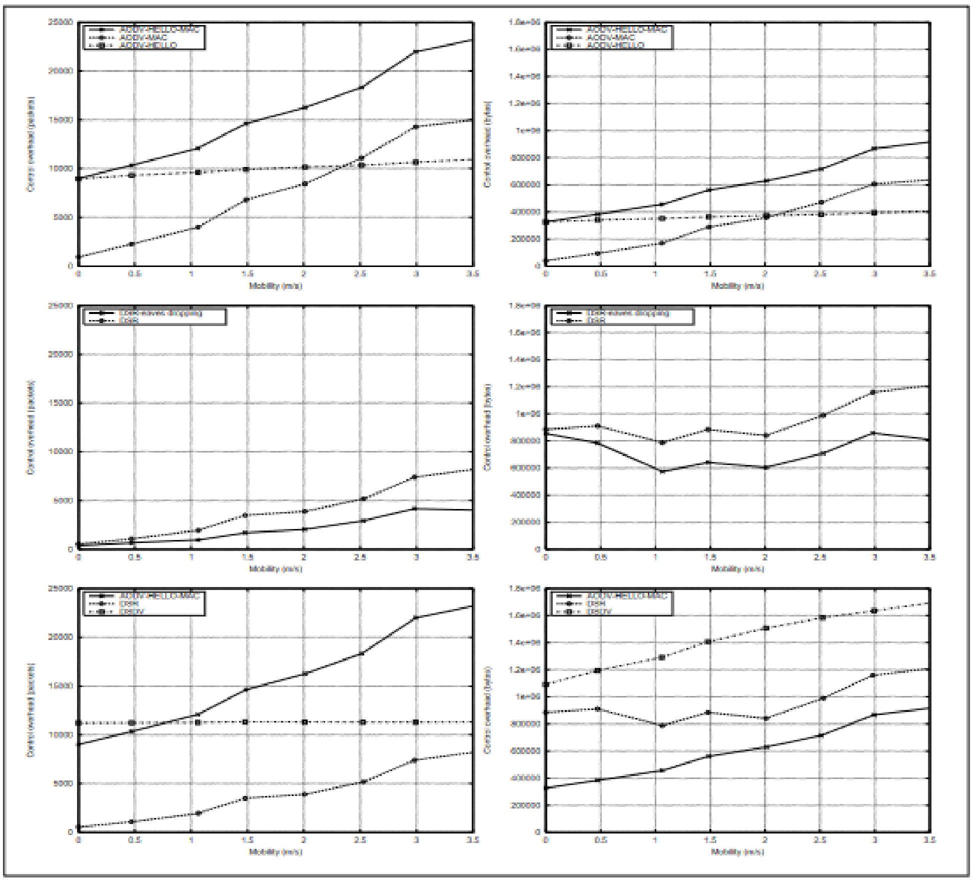

Figure 9: Mobility simulations–Overhead

It's interesting to observe how much control information is delivered for each protocol since discovering routes requires the routing protocol to send control information. There is a trade-off of sorts between the quantity of control information packets sent and the byte overhead. Naturally, a higher byte overhead would result in more bandwidth being lost. However, a large number of tiny control information packets would result in a higher frequency of acquisition of the radio medium used to send the packets. The cost of power and network usage for this can be high. No MAC layer overhead or physical layer framing is included in the data that we have plotted. Only the IP-level overhead has been examined. The overheads stated above would also be included in a fair comparison. For the straightforward reason that a real-world implementation's MAC layer may differ, we have opted not to provide these. Our focus was on examining the overall cost rather than the overhead that is specifically related to the IEEE 802.11 MAC protocol. Figure 18 presents the findings. Overhead is measured in packets in the first column, and overhead in bytes is shown in the second.

For all of the simulations run in which we have different mobility, the total number of control packets and byte overhead is given. The curves for the various AODV versions with link layer support have a similar appearance among them, as can be observed. The difference between the visible 270000-290000 bytes and about 8000-9000 packets corresponds to the hello packets. The curve is far more stable in the AODV version that only contains greeting messages. The triggered route replies that are sent in the event of a link failure and the new requests that are issued in the event of a route failure are the minor uptick that is apparent.

Thus, a route failure results in both new requests and triggered answers. For the straightforward reason that link layer support in AODV versions detects connection failure considerably sooner, resulting in a greater number of messages, this growth is substantially greater for those versions. DSR only accounts for the extra byte overhead from these packets; it does not include the data packets in the computation of the number of control packets. When comparing the DSR versions, it's important to keep in mind that the version without eavesdropping has around twice as much control overhead measured in messages and about 400000 extra bytes of overhead than the version with eavesdropping at the greatest mobility of 3.5. The byte overhead for DSR exhibits peculiar behavior, which can be attributed to the combination of delivered control messages and packets. Fewer packets will pass via the network when mobility rises. There is less byte overhead in the packets' source path when there are fewer packets. More mobility also translates into more topological changes, which raises the quantity of update messages. As a result, the byte overhead is falling; however, at roughly mobility 1.5, the number of control messages will rise, increasing the byte overhead.

Even in cases where mobility is exceptionally high, the amount of control messages in DSDV remains rather constant. This is how a proactive protocol that relies on recurring broadcasts works. Conversely, when mobility rises, so will the byte overhead. This is because as the number of link updates rises, so does the amount of information delivered in each update message.

V. Conclusion

5.1 Results

Simulations indicate the necessity for a specialized ad-hoc routing protocol as mobility increases. Feedback from link-layer protocols, such as IEEE MAC 802.11, is essential for monitoring link status and identifying neighbors. Using solely periodic IP-level messages can lead to significant packet losses, even when mobility rises slightly. Simulations indicate that traditional protocols, such as DSDV, perform poorly with increased mobility, making them unsuitable for mobile ad-hoc networks. AODV and DSR have overall performed well, even when mobility is high. DSR relies on source routing, which means that byte overhead in each packet can significantly impact overall network overhead as network demand and size increase. In these scenarios, a hop-by-hop routing system such as AODV is preferable. The source routing strategy has the advantage of learning more routes during route discovery. Source routing is not ideal for forwarding data packets due to its high byte overhead. Combining AODV with DSR may provide superior results compared to either alone.

REFERENCES

[1]. Adhikari, S., & Setua, S. K. (2014). Cooperative network intrusion detection system (CNIDS) in mobile adhoc network based on DSR protocol. *Proceedings of 2013 3rd International Conference on Computer Science and Network Technology, ICCSNT 2013*, 929–935. https://doi.org/10.1109/ICCSNT.2013.6967257

[2]. Ali, M. A. (2011). *Security Issues regarding MANET (Mobile Ad Hoc Networks): Challenges and Solutions*. *March*.

[3]. Almotairi, K. H., & Shen, X. S. (2015). A distributed multi-channel MAC protocol for Ad Hoc wireless networks. *IEEE Transactions on Mobile Computing*, *14*(1), 1–13. https://doi.org/10.1109/TMC.2014.2316822

[4]. Azgin, A., Altunbasak, Y., & Alregib, G. (2005). Cooperative MAC and routing protocols for wireless ad hoc networks. *GLOBECOM - IEEE Global Telecommunications Conference*, *5*, 2854–2859. https://doi.org/10.1109/GLOCOM.2005.1578280

[5]. Bhandari, R. (2024). *Journal of Recent Innovations in Computer Science and Technology Load Balancing in Wireless Mobile Ad Hoc Networks Journal of Recent Innovations in Computer Science and Technology*. *01*(01), 8–14.

[6]. Bisnik, N. (2005). Protocol Design for Wireless Ad hoc Networks : The Cross-Layer Paradigm. *Teknik Rapor, Rensselaer Polytechnic Institute*, 1–10.

[7]. Blakeway, S. J. (2015). *AN INVESTIGATION OF MOBILE AD-HOC NETWORK PERFORMANCE WITH COGNITIVE ATTRIBUTES APPLIED STEWART JOHN BLAKEWAY A thesis submitted in partial fulfilment of the requirements of Liverpool John Moores University for the degree of Doctor of Philosophy*. *June*.

[8]. Cadger, F., Curran, K., Santos, J., & Moffett, S. (2013). A survey of geographical routing in wireless Ad-Hoc networks. *IEEE Communications Surveys and Tutorials*, *15*(2), 621–653. https://doi.org/10.1109/SURV.2012.062612.00109

[9]. Cardei, M., Wu, J., & Yang, S. (2006). Topology control in ad hoc wireless networks using cooperative communication. *IEEE Transactions on Mobile Computing*, *5*(6), 711–724. https://doi.org/10.1109/TMC.2006.87

[10]. Choudhury, S., Roy, S. D., & Singh, S. A. (2008). Trust management in ad hoc network for secure DSR routing. *Novel Algorithms and Techniques in Telecommunications, Automation and Industrial Electronics*, 495–500. https://doi.org/10.1007/978-1-4020-8737-0_89

[11]. Dipobagio, M. (2009). An overview on ad hoc networks. *Inf.Fu-Berlin.De*, 1–9. https://www.inf.fu-berlin.de/groups/ag-tech/teaching/2008-09_WS/S_19565_Proseminar_Technische_Informatik/dipobagio09overview.pdf

[12]. Guan, Q., Yu, F. R., Jiang, S., & Leung, V. C. M. (2012). Joint topology control and authentication design in mobile ad hoc networks with cooperative communications. *IEEE Transactions on Vehicular Technology*, *61*(6), 2674–2685. https://doi.org/10.1109/TVT.2012.2196061

[13]. Han, L. (2019). Wireless Ad-hoc Networks 2 . Characters Challenges of Networks and Fundamental Wireless Ad-hoc. *Networks*, 1–6.

[14]. Khalid, M., Wang, Y., Ra, I. H., & Sankar, R. (2011). Two-relay-based cooperative MAC protocol for wireless ad hoc networks. *IEEE Transactions on Vehicular Technology*, *60*(7), 3361–3373. https://doi.org/10.1109/TVT.2011.2159872

[15]. Kohlstruck, C. (2023). *Development and Evaluation of Protocols for the Operation of Wireless Ad-Hoc Networks with Quality-of-Service Requirements PhD Thesis to*.

[16]. Library, T. L. H., & Core, C. (2019). *4.1 Cooperative communications*.

[17]. Marbach, P., & Qiu, Y. (2005). Cooperation in wireless ad hoc networks: A market-based approach. *IEEE/ACM Transactions on Networking*, *13*(6), 1325–1338. https://doi.org/10.1109/TNET.2005.860109

[18]. McDonald, A. B., & Znati, T. (1999). A path availability model for wireless ad-hoc networks. *IEEE Wireless Communications and Networking Conference, WCNC*, *1*, 35–40. https://doi.org/10.1109/WCNC.1999.797781

[19]. Moh, S., & Yu, C. (2012). A cooperative diversity-based robust MAC protocol in wireless ad hoc networks. *IEEE Transactions on Parallel and Distributed Systems*, *23*(3), 353–363. https://doi.org/10.1109/TPDS.2010.99

[20]. Nirmaladevi, K., & Prabha, K. (2023). A selfish node trust aware with Optimized Clustering for reliable routing protocol in Manet. *Measurement: Sensors*, *26*(October 2022), 100680. https://doi.org/10.1016/j.measen.2023.100680

[21]. Nandan, A., Das, S., Pau, G., Gerla, M., & Sanadidi, M. Y. (2004). Co-operative downloading in vehicular ad-hoc wireless networks. *2nd Annual Conference on Wireless On-Demand Network Systems and Services, WONS 2005*, 32–41. https://doi.org/10.1109/WONS.2005.7

[22]. Perkins, C. E., & Royer, E. M. (1999). Ad-hoc on-demand distance vector routing. *Proceedings - WMCSA'99: 2nd IEEE Workshop on Mobile Computing Systems and Applications*, *May*, 90–100. https://doi.org/10.1109/MCSA.1999.749281

[23]. Srinivasan, V., Nuggehalli, P., Chiasserini, C. F., & Rao, R. R. (2005). An analytical approach to the study of cooperation in wireless ad hoc networks. *IEEE Transactions on Wireless Communications*, *4*(2), 722–733. https://doi.org/10.1109/TWC.2004.842950

[24]. Tyagi, S., Som, S., & Khatri, S. K. (2021). Reliability-based dynamic multicast group formation provisioning local adjustment ensuring the quality of service globally in MANETs. International Journal of Parallel, Emergent and Distributed Systems, 36(2), 144-158.

[25]. Unnikrishnan, A., & Das, V. (2022). Cooperative Routing for Improving the Lifetime of Wireless Ad-Hoc Networks. *International Journal of Advances in Signal and Image Sciences*, *8*(1), 17–24. https://doi.org/10.29284/ijasis.8.1.2022.17-24

[26]. Wang, X., Li, J., & Tang, F. (2014). Network coding aware cooperative mac protocol for wireless Ad Hoc networks. *IEEE Transactions on Parallel and Distributed Systems*, *25*(1), 167–179. https://doi.org/10.1109/TPDS.2013.22