# MOPLJO-MANET: MULTI-OBJECTIVE PARAMETER LESS JAYA OPTIMIZATION FOR MOBILE AD HOC NETWORKS

T. Sudhakar[1], H. Hannah Inbarani[2], Osamah Ibrahim Khalaf[3] and Sameer Algburi[4]

[1]Assistant Professor, Department of Computer Science, CHRIST (Deemed to be University), Bengaluru, India, sudhakar.t@christuniversity.in

[2]Associate Professor, Department of Computer Science, Periyar University, Tamilnadu, India, hhinba@gmail.com

[3]Professor, Department of Solar, AI-Nahrain Research Center for Renewable Energy, AI-Nahrain University, Jadriya, Baghdad, Iraq, usama81818@nahrainuniv.edu.iq

[4]Professor, AI-Kitab University, College of Engineering Techniques, Sameer.algburi@uoalkitab.edu.iq

**Abstract**

MANET (Mobile Ad hoc Network) is a self-contained network made up of mobile nodes that communicate without the use of a central controller. These nodes each function as a router, allowing packets to be delivered and received. In this research, routing and mobility issues have been examined. Since nodes can join or leave an ad hoc network at any time, it may be difficult to establish routing between any two nodes in an ad hoc network. This implies that a method that was perfect at one moment can suddenly become ineffective. In this research, a multi-objective parameter-less Jaya optimization is used to determine the best path between network nodes. The main contribution of this research work is to increase the PDR , throughput and minimize the delay of the network. In addition, the proposed technique was compared to current optimization techniques like Genetic Algorithm (GA) and Artificial Bee Colony (ABC), as well as well-known reactive protocols like DSR (Dynamic Source Routing) and AODV (Ad hoc On-Demand Distance Vector). The suggested solution surpasses others in terms of performance characteristics such as Packet Delivery Ratio (PDR) (92.72 %), Delay (0.01042 sec), and Throughput (89.74 %).

**Keywords:** optimization, Jaya Algorithm, Routing, Genetic, ABC, network

## 1. INTRODUCTION

To address the problem of mobility and routing issues in MANET, a novel multi-objective Jaya algorithm is employed in this work [7] [8]. An ad hoc network is a self-organizing mobile network that does not rely on any pre-existing infrastructure. A wireless network connects every mobile node. Every mobile node in this network will serve as a host node, linking a single source to a group of receivers [9]. In a wireless ad hoc network, node movement is also one of the major causes of data loss. This algorithm is compared to other mobility models and algorithms such as GA [14], ABC [14], random waypoint [3], random direction [3], and random walk [3]. In this paper, the efficiency of two ad hoc routing protocols is compared to three random mobility models. Furthermore, the impact of routing protocols on their behaviour is investigated [6].

A decentralized wireless ad hoc network is one in which mobile nodes connect over short distances and with limited battery capacity. Here, every single node act as a host node. Also, the hosts make the network change rapidly. In this field, several researchers are working to propose appropriate routing protocols to resolve these issues on an ongoing basis. Mobility scenario models are created in a popular Bonnmotion tool [1]. These scenarios are created under different values of pause time and topology size [4].

In a MANET, each mobile node serves as both a router and a host. There are drawbacks and downsides to both wired and cellular networks [2]. The wired network is much more reliable than a wireless network because the user can easily fix the problem of identification and resolve it [5]. However, the problem of identification is easy in the wired network but it cannot reach the entire region [15]. To overcome the problem of the wired network, wireless technology has been rolled out. In this study, soft computing methodologies were used to handle route selection and route classification in mobile ad hoc networks. Multi-path routing is one method of increasing the information's trustworthiness [16].

In recent years, swarm intelligence has become increasingly important in determining a path between source and destination nodes. The JA is designed in this paper to discover the best path in a wireless network and is compared to well-known reactive routing protocols such as DSR and AODV. The population-based algorithms can be divided into two categories [1]: Evolutionary Algorithms (EA) and Swarm Intelligence (SI). Genetic Algorithms (GA), Evolutionary Programming (EP), and others are examples of key evolutionary algorithms. Particle Swarm Optimization (PSO), Ant Colony Optimization (ACO), and Bee Colony Optimization (BCO) are examples of swarm intelligence algorithms [1]. Since many academics are working in this sector, optimization strategies to overcome numerous difficulties in wireless ad hoc networks have been gaining traction recently [2] [3].

***a) Objectives of the paper***

The main goal of this research is to use soft computing approaches to pick the best path in networks, resulting in more fruitful outcomes than existing methodologies. In experimental analysis, soft computing techniques are applied to select an optimal route for better communication between the source and destination nodes. Particularly, swarm intelligence has been applied to resolve routing and mobility problems in MANETs. In this work, the proposed JA has been compared with existing techniques such as GA, and ABC, besides routing protocols.

***b) Motivations of the paper***

The motivation to study wireless ad hoc networks is inspired by applying a real-time application and observing its characteristics and performance to cater to demands where slight carelessness can result in fatal outputs. The proposed work focuses on optimal route selection which would be much helpful in simplifying the networks. The inspiration for the current research comes from the potential of soft computing techniques and their application in the wireless ad hoc network field. This research work also analyses the performance of existing routing protocols with mobility models.

### *c) Contributions of the paper*

The accomplishments of this study are listed as follows:

1. To minimize delay and bandwidth in wireless ad hoc networks.

2. To Increase PDR, and throughput, then minimize the delay.

3. To use the route discovery strategy with different reactive routing protocols such as DSR and AODV.

4. To examine the performance of routing protocols under various mobility models, including RWP, RD, and RW.

5. Finally, in a simulation environment, assess the performance of the Jaya algorithm.

This paper is structured as follows: An overview of optimization methods is defined in Section 3.2., Section 3.3 and 3.4 to illustrate the JA which is proposed for the wireless network routing. The experimental results and simulation setup are discussed in Section 3.5. Section 3.6 presents a profound analysis of simulation work, and the summary of the chapter is presented in section 3.7.

## 2. REVIEW OF WORKS

In recent years, several MANET routing protocols have been created [10]. Proactive, reactive, and hybrid protocols are the most common types of such protocols. GSR (Global State Routing) [10], CGSR (Cluster-Head Gateway Switch Routing) [10], FSR (Fisheye State Routing) [10], OLSR (Optimized Link State Routing) [10], WRP (Wireless Routing Protocol) [10], and DSDV (Destination

Sequenced Distance Vector) [10] are examples of proactive routing. If the topology of the network changes in a dynamic environment like MANET, these routing tables are always required to be adjusted by periodically sending routing information (such as connection status and distance vector) between the nodes. Those designs do not include a routing table. A node must first discover the path before sending a data packet. As a result, while overhead routing is reduced, end-to-end data packet delivery improves. In this part, several authors provide a brief evaluation of similar works published in recent years. In this section, Evolutionary algorithms are used to solve network challenges such as routing, mobility, and mobility prediction, among others [17].

Ammar W. Mohemmed et al. (2008) have investigated the shortest path (SP) problem using Particle Swarm Optimization. The main problem of a network is finding the shortest path between the source and the set of receivers. PSO based approach finds the optimal route with high success rates and also finds closer sub-optimal ways. The proposed approach is opposed to the GA, and the proposed strategy gives a better result than others. An author addressing the SP problem uses a simple network topology size [10].

Jyoti Jain et al. (2011) have reviewed the Ant Colony algorithms for further modification. In this proposed work, ACO (Ant Colony Optimization) has been used in case of link failure situation. The path would be discovered by reactive routing, and sustained by the periodic generation of hello messages by all of the connection nodes. Additionally, all nodes in the chain must proactively consider an alternate route for next to the next node [21]. By utilizing this approach, the parameters of the throughput and low delay will possibly boost the overhead. The Overhead should raise the number of path errors in constructive path searching at the same time such that the parts available for alternate route seeking should decrease [11] [20].

Table 1 summarizes the various routing approaches proposed for route selection in the past decade.

**Table 1. Summary of routing approaches for MANETs**

| Sno | Year | Authors | Algorithm | Problem | Routing approach | Parameters |
|---|---|---|---|---|---|---|
| 1. | 2005 | Hui Liu et al. | Genetic Algorithm | Select optimal Multipath Routing | Genetic Fuzzy Multi-Path Routing | Link stability, queue occupancy rate, and Energy consumption rate. |
| 2. | 2007 | Mustafa Al-Ghazal et al. | Genetic Algorithm | Route optimization | CGSR (Cluster Gateway Switching Protocol) | Battery power and mobility |

| | | | | | | |
|---|---|---|---|---|---|---|
| 3. | 2008 | AmmarW. Mohemmed et al. | PSO algorithm | Shortest path problem | - | |
| 4. | 2011 | Jyoti Jain at al. | ACO | ACO modification | - | Throughput and End to End delay |
| 5. | 2014 | Gurpreet Singh et al. | Innovative Ant colony Algorithm | Rapid Change of Mobile nodes | Reactive Protocols (HOPNET, ANTLG and AODV) | Hop count, delay |
| 6. | 2015 | Shubhajeet Chatterjee et al. | Enhanced DSR algorithm | Reduce End to End Delay | Reactive Protocol (DSR) | Delay, Energy and Routing Overhead |

| | | | | | | |
|---|---|---|---|---|---|---|
| 7. | 2015 | Gin-Xian Kok et al. | Network coding Algorithm | Improve the performance of throughput by reducing the workload | - | Hop count, Routing overhead |
| 8. | 2015 | NaercioMagaia et al. | Multi-objective Routing Algorithm | Minimize Delay and Excepted Transmission Count | Reactive Protocol (DSR) | Delay and ETX and Bandwidth |
| 9. | 2016 | Roberto Magan-Carrion et al. | PSO algorithm | Improve Network Connectivity in Multi-hop Environment | Reactive Protocols | Node Hop count, Bandwidth |
| 10. | 2019 | D. MadhuBabu and M. Ussenaiah | Cuckoo Search algorithm | Provide optimal multi-cast routing | AODV | Distance, Destination Flag |

The optimization approach plays a vital role in mobile ad hoc networks, as this literature review shows. For route selection, many strategies have been proposed in the literature. However, they all have drawbacks, such as a low maximum packet delivery latency and a flat accuracy rate. As a result, in this research, a combination of diverse algorithms that can improve the system's efficiency is applied.

## 3. PROPOSED MULTI-OBJECTIVE JAYA ALGORITHM FOR MOBILE AD HOC NETWORKS

The Jaya algorithm [13] was proposed by Professor Venkata Rao, and it has been tested using benchmark functions. The given technique outperforms others like GA (Genetic Algorithm), PSO (Particle Swarm Optimization), ABC (Artificial Bee Colony), and ACO (Ant Colony Optimization), among others. Both constrained and unrestrained problems are basic and straightforward to answer. The Jaya method is built on the concept of obtaining a solution for a given problem, and it converges to global optimal results. This method simply needs general control parameters like population size and the number of design variables; it doesn't need any algorithm-specific control factors like mutation probability or crossover selection as GA does. It's a parameter-free algorithm for a wireless network that's algorithm-specific. The main purpose of this chapter is to reduce the network's mobile nodes' delay and bandwidth [2]. PSO [23] primarily uses the whole population of particles to identify the best solution (mobile node). One of the nature-inspired algorithms is Cuckoo Search [24]. Equations (1) and (2) are used to compute the Delay and Bandwidth as follows [14]:

$$Delay\big(P(s,t)\big) = \sum_{e \in P(s,t)} C_e \tag{1}$$

$$Bandwidth\big(P(s,t)\big) = min\{bandwidth(e), e \in P(s,t)\} \tag{2}$$

$$Minimize\ f(P(s,t)) = \delta_1 * delay\big(P(s,t)\big) + \delta_2 * bandwidth\big(P(s,t)\big) \quad (3)$$

Note that the parameters, $\delta_1$and $\delta_2$ are the objective weighting coefficients used to calculate the significance of these two objectives [14] Where $P(s,t)$ denotes the path from the source node 's' to destination node 't' of a multicast tree and $C_e$ is the transmission delay on communication link $(e)$. The bandwidth of $P(s,t)$ denotes the minimum link bandwidth in the whole route. The fitness function is generated for each path in topology based on the monitored parameters and estimated as Equation (3) is the name of this objective function [15] [14].

Subject to:

$$Delay\ (P(s,t)) \leq L_d \quad (4)$$

$$Bandwidth\ (P(s,t)) \geq U_b \quad (5)$$

Here, the scalarization technique [15] [16] [17] [18] produces a single solution for the multi-objective function and the weight is calculated before the optimization process. The equal weight [15] [16] [17] [18] technique can be used for calculating the weights from the following Equation (6).

$$w_i = \frac{1}{n} \quad (6)$$

Where, $i = 1, 2, 3, 4, \dots\dots.., n$, and $n$ are the number of objective functions. $L_d$ is a lower bound of delay and $U_b$ is an upper bound of every path [8].

### 3.1 Mapping of Jaya algorithm to mobile ad hoc networks

This section briefly illustrates how the Jaya algorithm is used to the problem of mobile ad hoc networks. Let $f(x)$ be the target value to be maximized or minimized, and this is solely a minimization problem in this study. Assume that there is m number of design variables$(j = 1, 2, 3, \dots., m)$. Only two design factors, bandwidth and latency, are used in this study [13]. At any iteration i, the population size is represented by the total number of mobile nodes in a network where $n$ is a candidate solution, i.e. $(k = 1,2,3, \dots, n)$. The best candidate solution is obtained from fitness function $f(x)$ i.e. $f(x)_{best}$. Also, the worst candidate solution is obtained from $f(x)$ i.e. $f(x)_{worst}$. If the $X_{j,k,i}$ value of $j^{th}$ design variable for the $k^{th}$ candidate solution is obtained during the $i^{th}$ iteration, then the value is changed as per Equation (7).

$$X'_{j,k,i} = X_{j,k,i} + r_{1,j,i}\left(X_{j,best,i} - \left|X_{j,k,i}\right|\right) - r_{2,j,i}\left(X_{j,worst,i} - \left|X_{j,k,i}\right|\right) \tag{7}$$

The value of the variable j for the best route is $X_{j,best,i}$ while the value of the variable $j$ for the worst route is $X_{j,worst,i}$. $r_{1,j,i}$ and $r_{2,j,i}$ are the two random values for the $j^{th}$ variable during the $i^{th}$ iteration in the range [0, 1], and $X'_{j,k,i}$ is the updated route value of $X_{j,k,i}$. The term "$r_{1,j,i}\left(X_{j,best,i} - \left|X_{j,k,i}\right|\right)$" denotes the solution's propensity to get closer to the correct solution (route), while "$r_{2,j,i}\left(X_{j,worst,i} - \left|X_{j,k,i}\right|\right)$" denotes the solution's inclination to avoid the worst solution. If $X'_{j,k,i}$ yields a better function value, it is allowed. All of the acceptable function values at the end of the iteration are kept and used as the input for the next iteration. Continue to seek better performance (i.e., finding the best answer) and avoid defeat (i.e. stay away from the worst solution). As shown in Figure 3, the Jaya algorithm is applied to mobile ad hoc networks in the following steps.

| ***Multi-objective Jaya Algorithm (MoJA)*** |
| --- |
| ***Input: As many pathways as feasible from source to destination.***<br>***Output: Optimal path solution as an output*** |
| ***Define node ()***<br>        *Node creation();*<br>        *Set mobility models();*<br>        *Set initial position();*<br>        *Set Neighborhood nodes();*<br>***End define ()***<br>***Main ()***<br>        *Path ();// all possible paths from a single source to* |

*destinations*

*Compute bandwidth and delay of the path using Equations (1) and (2).*

*Make an initial population set.*

*Initialize the population size, no. of design variables, and termination condition.*

*Compute fitness values using Equation (3).*

*Identify the best and worst solutions based on fitness value.*

*Modify the best and worst values based on the Equation (7)*

$$X'_{j,k,i} = X_{j,k,i} + r1_{j,i}\left(X_{j,best,i} - \left|X_{j,k,i}\right|\right) - r2_{j,i}\left(X_{j,worst,i} - \left|X_{j,k,i}\right|\right)$$

*If the solution is corresponding to* $X'_{j,k,i}$

*Accept and replace previous solutions.*

*Else*

*Keep the previous Solution.*

*This step is used to check the Termination Criteria, i.e. no. of iterations returning the optimal solution otherwise repeat the steps.*

***End main ()***

**Figure 3. Mapping of Jaya Algorithm**

## 5. EXPERIMENTAL RESULTS

In this experimental analysis, the proposed JA is compared with other optimization techniques, namely GA and ABC. In this segment, the environment for simulation setup, simulation parameters, and performance of QoS metrics like packet delivery ratio, delay, and average throughput under several nodes in the network are given.

### *a) Simulation Environment setup*

This simulation work is performed using Network simulator 2.34 [19]. This topology uses 200 nodes for simulation under various scenarios. The node movement patterns are created using Bonnmotion Tool [20], which is a Java-based tool. This tool has 21 types of mobility models for MANET environments.

Every node uses IEEE 802.11 [20] as the Medium Access Control (MAC) layer, which includes a distributed coordination function. Ad hoc On-Demand Distance Vector Routing is the routing protocol utilised by each network node (AODV). Each node creates 1000 seconds of constant bit rate traffic with 1 packet per source.

The cbrgen tool, which is included in the NS-2 package, is used to generate traffic. The cbrgen tool picked the number of sources and destinations at random. The NS-2 simulator is used to mimic data packet transmission and reception. A new trace format is used to store the transmission and reception traces. AWK scripts are used to calculate the end-to-end delay from such trace files. After that, the computed delay values are sent to NS2 for further analysis. The simulation settings utilized in this study are shown in Table 2.

**Table 2. MoJA simulation parameters**

| Simulation parameters | Specifications |
|---|---|
| **Channel Type** | Wireless |
| **Propagation model** | Two ray ground model |
| **Constant bitrate connections** | Five |
| **Network interface type model** | Ad hoc |
| **Antenna type** | Omnidirectional antenna |
| **Total simulation time** | 1000 s |
| **Simulation coverage area size** | (500 m, 500 m) |
| **Protocols** | AODV and DSR |
| **Mobility models** | RWP, RD, and RW models |
| **Transport Protocol** | UDP Protocol |
| **MAC Type** | IEEE 802.11 |
| **Number of nodes** | 50, 100, 150 and 200 |
| **Node transmission range** | 200 m |

### *b) Performance metrics*

***Packet Delivery Ratio (PDR):*** Multiple sources relay the ratio of the average amount of data packets obtained by the recipient node to the number of data packets. Equation (8) is used to measure the packet transmission ratio between both transmitted and obtained data packets.

$$Packet\ Delivery\ Ratio = \sum_i \frac{PD}{PS} * 100 \tag{8}$$

Note that, $PD$ – Packet Delivery, $PS$ – Packet Sent, $i^{th}$packet [21].

***Delay:*** Using Equation (9), the average delay of packet delivery is calculated between the arrival time of data packets and the sent time of data packets, followed by a total number of connections in topology.

$$Delay = \sum PAT_i - PST_i \tag{9}$$

Where, $PAT$ – Packet Arrival Time, $PST$ – Packet Start Time, $i^{th}$packet [21].

***Throughput:*** Throughput is calculated by data packets successfully delivered from one node to another node over a communication network. Normally, the below Equation (10) shows the calculation of throughput where n is the number of data packets which usually takes bits/second [10].

$$Throughput = \sum_i \frac{PD}{PAT-PST} \tag{10}$$

Where, $PD$ – Packet Delivery, $PAT$- Packet Arrival Time, $PST$ – Packet Start Time [21].

## 6. RESULT DISCUSSIONS AND ANALYSIS

The suggested task is simulated in this section using Network Simulator under various conditions (NS2). In this study, 200 mobile nodes (MNs) were used to simulate two well-known ad hoc routing protocols, AODV and DSR, versus three mobility models, namely random waypoint, random direction, and Random Walk models, and the number of nodes was also evaluated.

Three mobility models, the RWP, RD, and RW models were employed in simulation studies in this section. The study compares three mobility models based on the number of mobile nodes. Among these mobility models, the RWP model consistently outperforms other mobility models and optimization algorithms.

In comparison to state-of-the-art optimization methods, the experimental findings show how alternative mobility models based on routing protocols perform. To analyse the performance of mobility models based on routing protocols such as AODV and DSR, two optimization strategies were used: Genetic Algorithm (GA), Artificial Bee Colony (ABC), and traditional AODV.

Different measurement criteria such as PDR, latency, and throughput were used to assess the algorithms' efficiency. Tables 3, 4, 5, 6, 7, and 8 show the performance of the RWP, RD, and RW models under AODV and DSR, respectively. The proposed algorithms' performance was evaluated using several mobility models.

The suggested AODV-JA and DSR-JA algorithms outperform the other algorithms, according to the experimental results. When examining the overall performance of the proposed techniques, it is claimed that AODV-JA under RWP outperforms other mobility-based models and also produces the highest PDR, 81 %. It's also worth noting that the traditional AODV procedure has the lowest PDR, at 32 %.

Figures 5, 6, and 7 show how the average packet delivery ratio performs with different numbers of mobile nodes. The suggested Jaya algorithm is compared to two ad hoc reactive routing protocols in this paper. When compared to the other approaches, DSR-JA and AODV-JA produce superior results than AODV, DSR, DSR-GA, AODV-GA, DSR-ABC, and AODV-ABC, respectively. The PDR will have a reduced packet transmission rate and a longer pause interval as the number of nodes increases.

**Table 3. Performance Analysis of Random Way Point model based on AODV**

| Routing Protocols | Number of nodes | PDR (%) | Average Throughput (Kb/s) | Delay (Sec) |
|---|---|---|---|---|
| AODV | 50 | 40.40 | 70.58 | 0.01166 |
| | 100 | 42.58 | 60.48 | 0.02055 |
| | 150 | 45.20 | 55.00 | 0.01152 |
| | 200 | 30.45 | 54.42 | 0.01549 |
| AODV-GA | 50 | 42.54 | 76.41 | 0.02245 |
| | 100 | 40.21 | 65.21 | 0.01944 |
| | 150 | 41.75 | 60.45 | 0.02045 |
| | 200 | 32.71 | 55.47 | 0.01184 |
| AODV-ABC | 50 | 47.80 | 78.45 | 0.01512 |
| | 100 | 51.41 | 75.47 | 0.01777 |
| | 150 | 54.20 | 70.78 | 0.01974 |
| | 200 | 33.75 | 71.48 | 0.01328 |
| AODV-JA | 50 | 81.80 | 81.62 | 0.00142 |
| | 100 | 78.58 | 80.00 | 0.00985 |
| | 150 | 75.20 | 76.45 | 0.01078 |
| | 200 | 60.45 | 75.74 | 0.01074 |

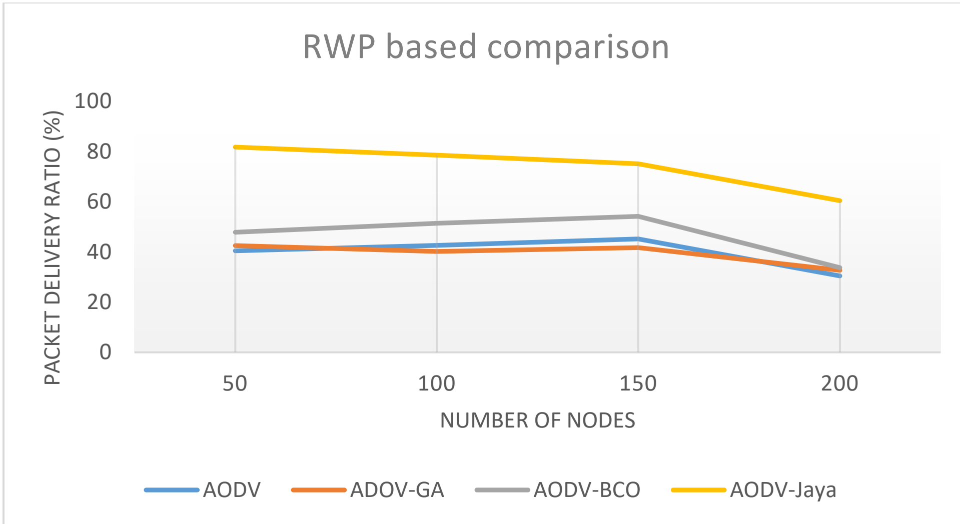


**Figure 5. PDR - Comparison of AODV based algorithm under RWP**

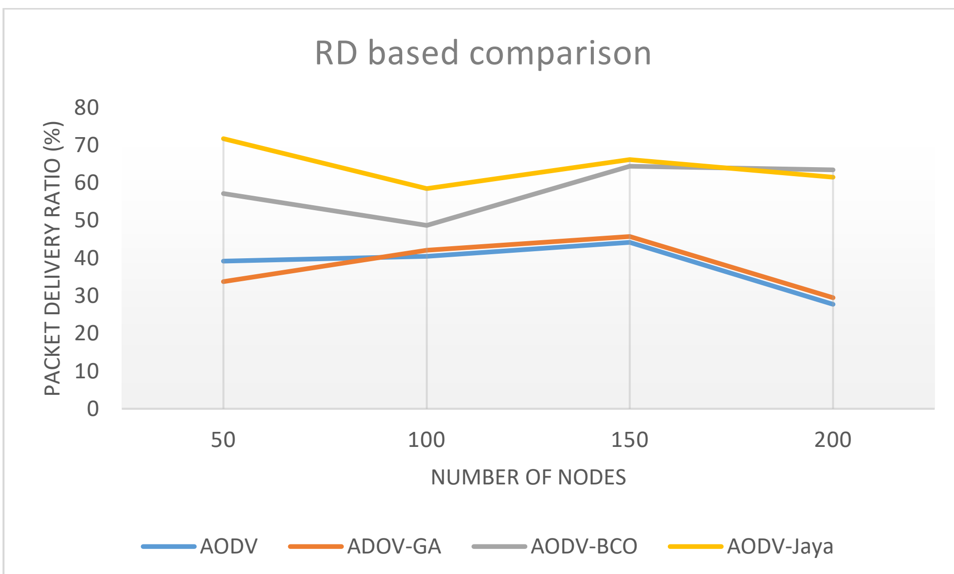


**Figure 6. PDR - Comparison of AODV based algorithm under RD**

**Table 4. Performance Analysis of Random Direction model based on AODV**

| Routing Protocols | Number of nodes | PDR (%) | Average Throughput (Kb/s) | Delay (Sec) |
|---|---|---|---|---|
| AODV | 50 | 39.21 | 68.18 | 0.01344 |

| | | | | |
|---|---|---|---|---|
| | 100 | 40.50 | 60.48 | 0.02266 |
| | 150 | 44.20 | 59.65 | 0.02344 |
| | 200 | 27.75 | 55.74 | 0.02374 |
| AODV-GA | 50 | 33.74 | 75.1 | 0.01245 |
| | 100 | 42.11 | 75.71 | 0.02016 |
| | 150 | 45.75 | 70.85 | 0.02146 |
| | 200 | 29.50 | 60.14 | 0.01564 |
| AODV-ABC | 50 | 57.18 | 79.5 | 0.01047 |
| | 100 | 48.71 | 75.78 | 0.01852 |
| | 150 | 64.42 | 71.73 | 0.01762 |
| | 200 | 63.45 | 69.54 | 0.01262 |
| AODV-JA | 50 | 71.74 | 83.42 | 0.00919 |
| | 100 | 58.48 | 81.92 | 0.00974 |
| | 150 | 66.22 | 80.24 | 0.00864 |
| | 200 | 61.50 | 78.66 | 0.008452 |

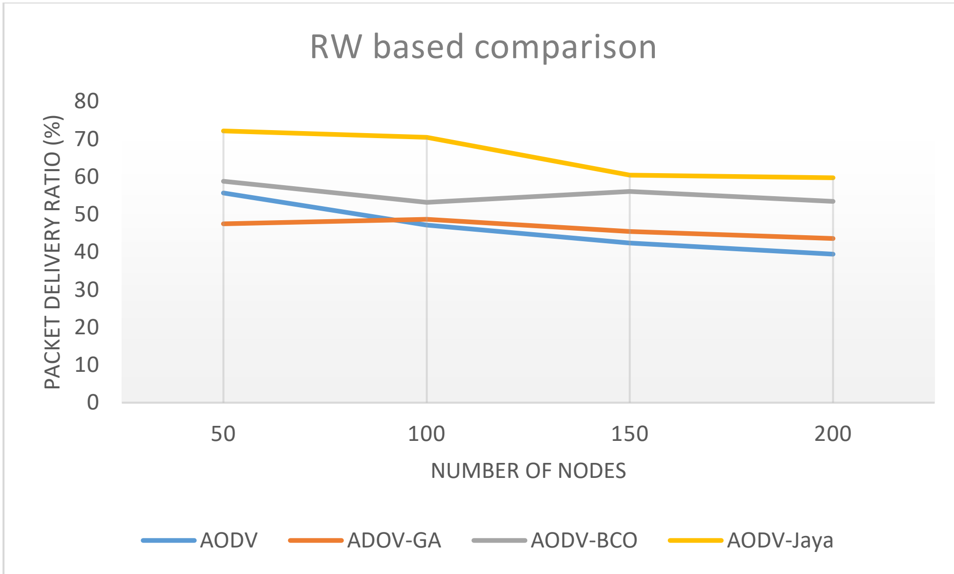


**Figure 7. PDR- Comparison of AODV based algorithm under RWP**

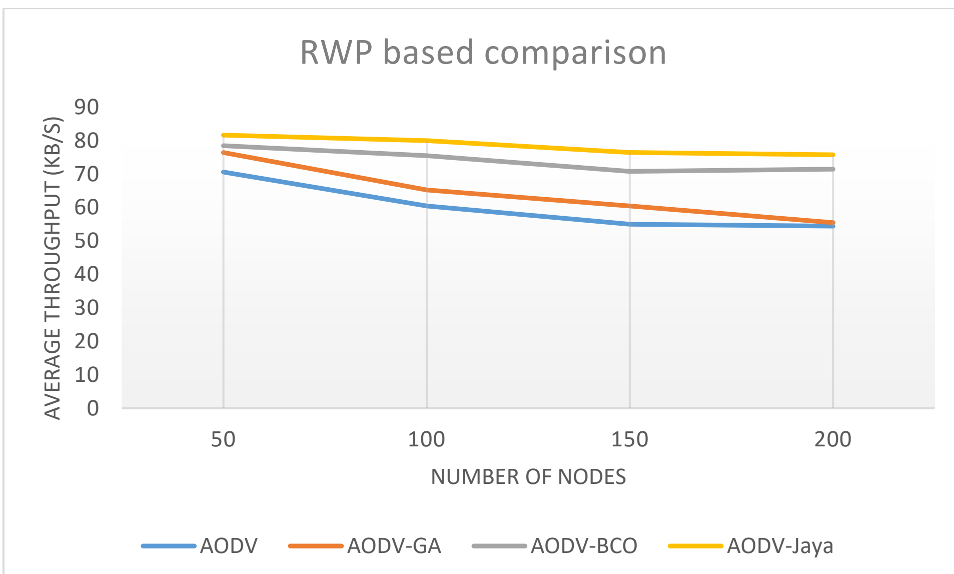


**Figure 8. Throughput - Comparison of AODV based algorithm by RWP**

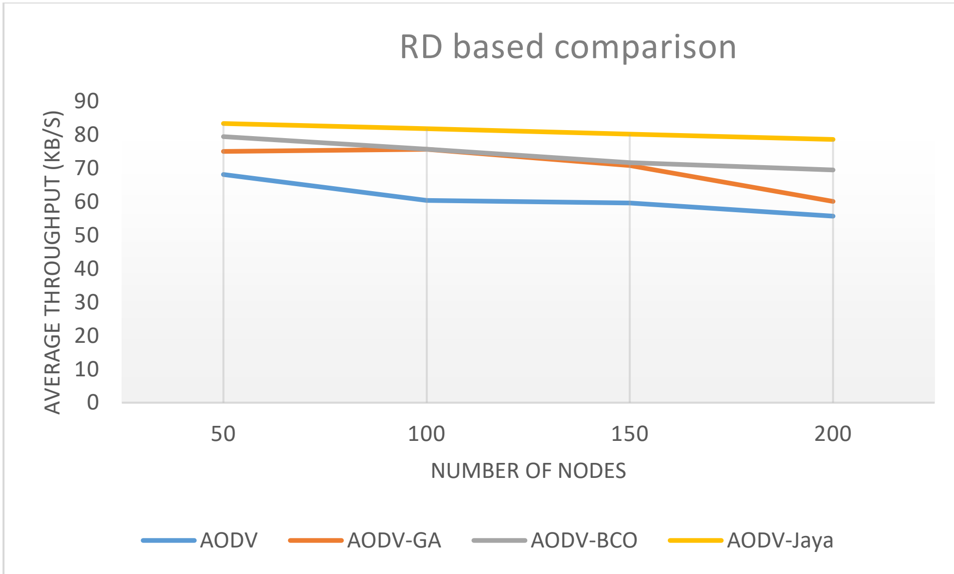


**Figure 9. Throughput - Comparison of AODV based algorithm under RD**

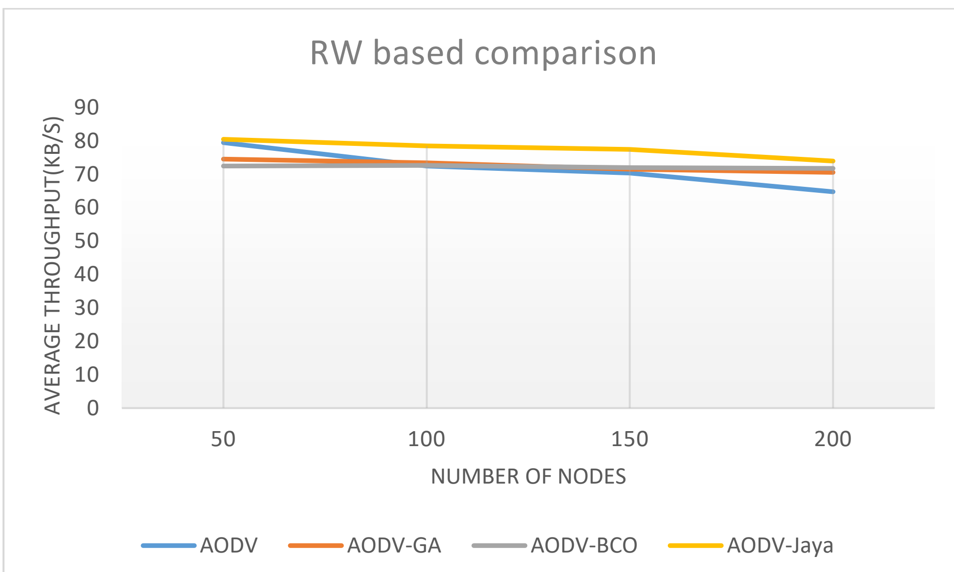


**Figure 10. Throughput - Comparison of AODV based algorithm under RW**

**Table 5. Performance Analysis of Random Walks model based on AODV**

| Routing Protocols | Number of nodes | PDR (%) | Average Throughput (Kb/s) | Delay (Sec) |
|---|---|---|---|---|
| AODV | 50 | 55.70 | 79.45 | 0.01652 |
| | 100 | 47.18 | 72.41 | 0.02264 |
| | 150 | 42.42 | 70.32 | 0.02364 |
| | 200 | 39.45 | 64.74 | 0.02374 |
| AODV-GA | 50 | 47.54 | 74.52 | 0.01841 |
| | 100 | 48.71 | 73.45 | 0.01755 |
| | 150 | 45.50 | 71.45 | 0.02175 |
| | 200 | 43.61 | 70.54 | 0.01344 |
| AODV-ABC | 50 | 58.81 | 72.45 | 0.01464 |
| | 100 | 53.21 | 72.64 | 0.01522 |
| | 150 | 56.12 | 72.00 | 0.01847 |
| | 200 | 53.45 | 71.74 | 0.01127 |
| AODV-JA | 50 | 72.18 | 80.45 | 0.00782 |
| | 100 | 70.45 | 78.47 | 0.00684 |
| | 150 | 60.40 | 77.41 | 0.01274 |
| | 200 | 59.74 | 73.94 | 0.01154 |

The mobile node varies as 50, 100, 150, and 200. When the node size is 100, the delay is less (0.01). If the number of nodes is increased, sometimes it gets packet loss. In this simulation, the Random Walk and Random Direction model gets high packet loss than RWP. The delay of three different mobility models for various routing protocols is shown in Figures 11, 12, and 13. Here, the mobility model with minimum delay is considered to be the high-performance mobility model.

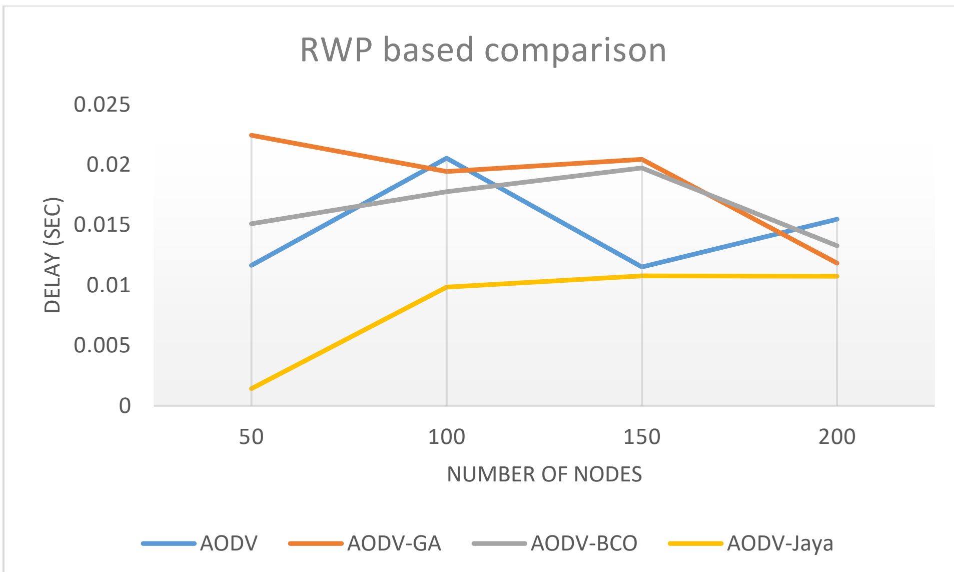


**Figure 11. Delay - Comparison of AODV based algorithm under RWP**

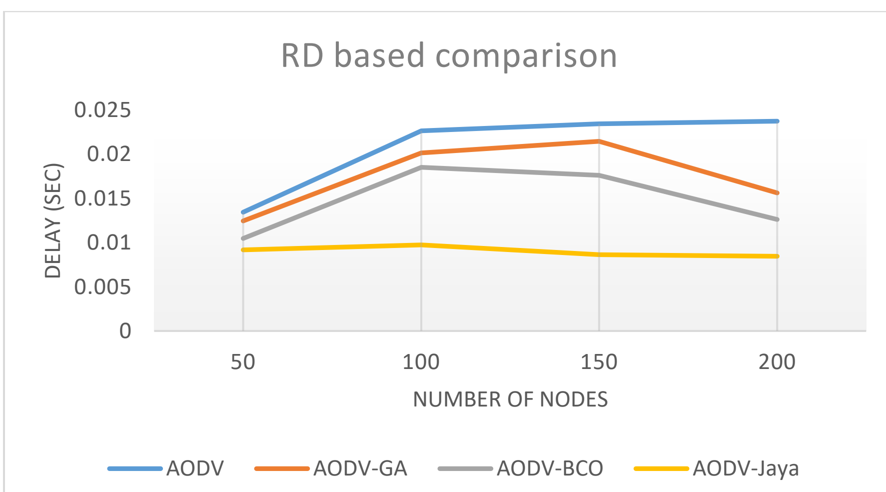


**Figure 12. Delay - Comparison of AODV based algorithm under RD**

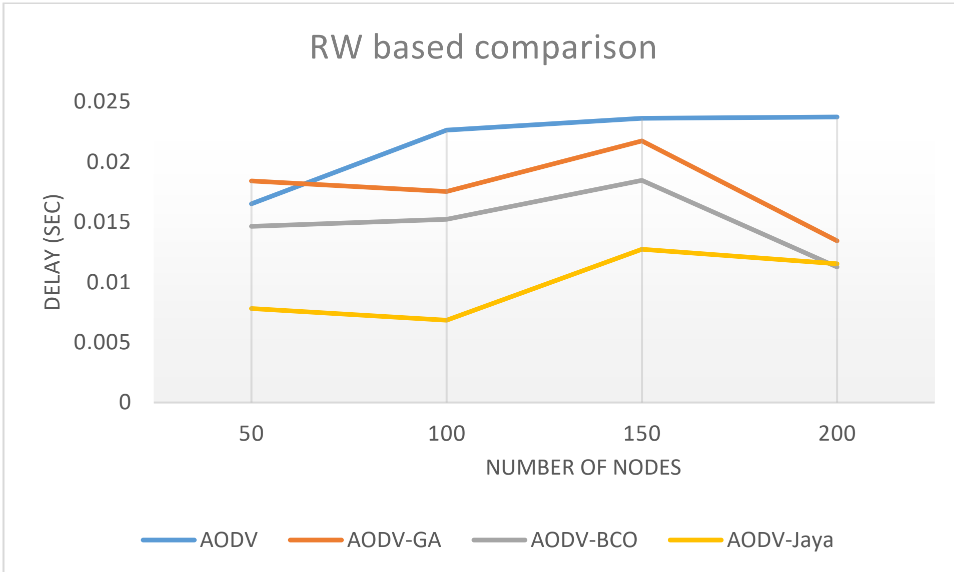


**Figure 13. Delay - Comparison of AODV based algorithm by RD**

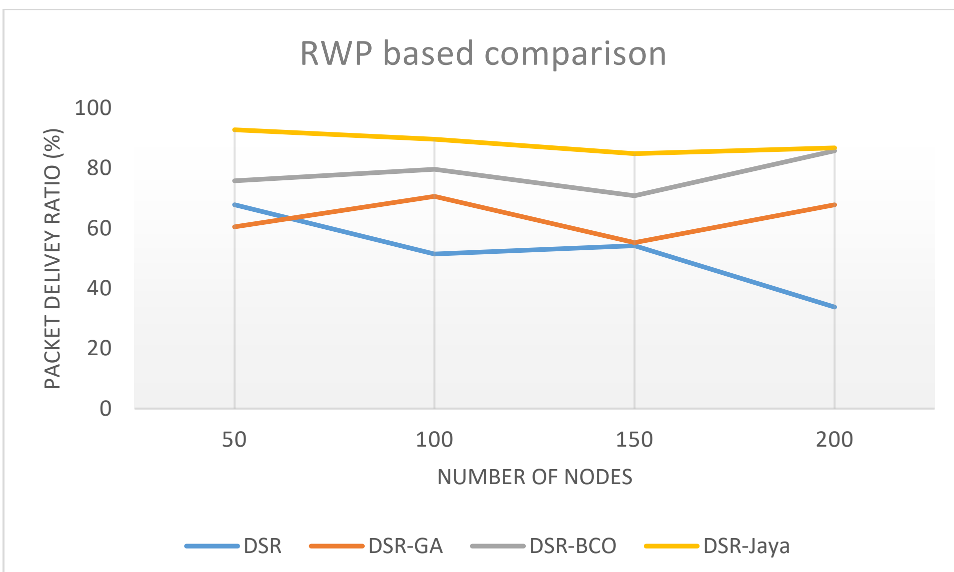


**Figure 14. PDR - Comparison of DSR based algorithm by RWP**

**Table 6. Performance Analysis of Random Way Point model based on DSR**

| Routing Protocols | Number of nodes | PDR (%) | Average Throughput (Kb/s) | Delay (Sec) |
|---|---|---|---|---|
| DSR | 50 | 67.8 | 78.41 | 0.04154 |
| | 100 | 51.41 | 77.25 | 0.04365 |
| | 150 | 54.2 | 75.75 | 0.04329 |
| | 200 | 33.75 | 71.48 | 0.04924 |
| DSR-GA | 50 | 60.48 | 79.45 | 0.03157 |
| | 100 | 70.58 | 76.47 | 0.03574 |
| | 150 | 55.2 | 73.45 | 0.03725 |
| | 200 | 67.75 | 72.66 | 0.03845 |
| DSR-ABC | 50 | 75.72 | 79.89 | 0.02168 |
| | 100 | 79.54 | 77.93 | 0.03074 |
| | 150 | 70.78 | 74.77 | 0.03584 |
| | 200 | 85.74 | 72.71 | 0.03654 |
| DSR-JA | 50 | 92.72 | 89.74 | 0.01042 |
| | 100 | 89.54 | 85.41 | 0.01485 |
| | 150 | 84.78 | 80.12 | 0.02034 |
| | 200 | 86.74 | 78.12 | 0.02975 |

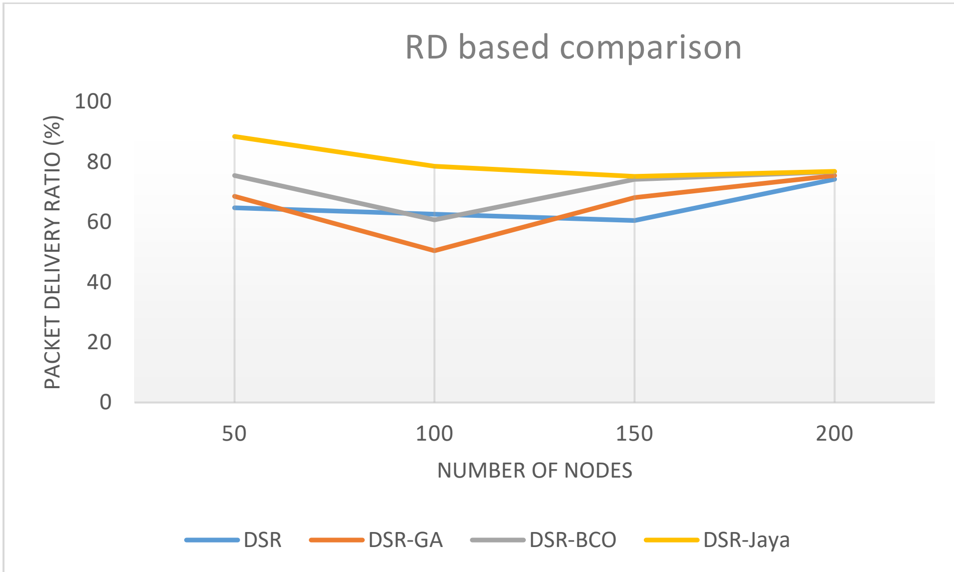


**Figure 15. PDR - Comparison of DSR based algorithm under RD**

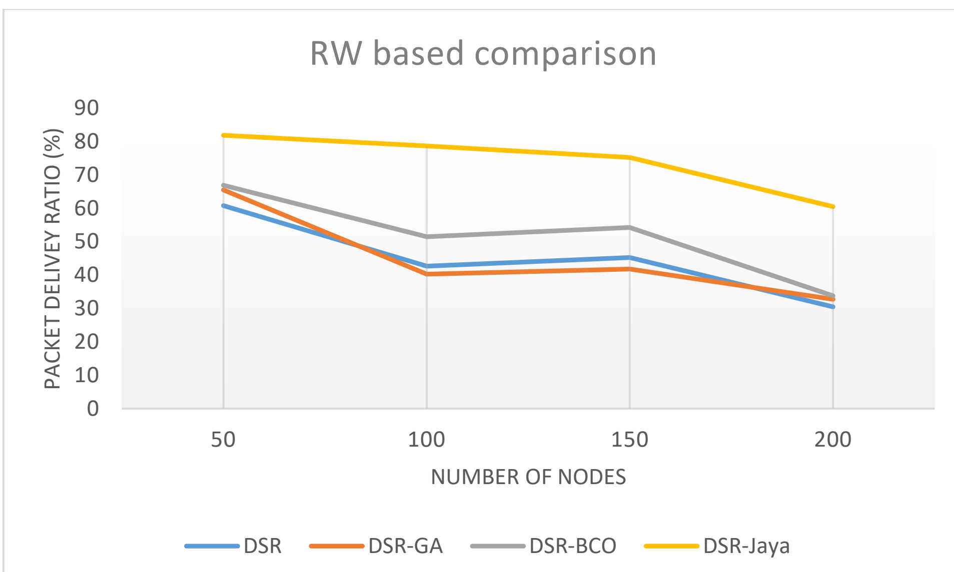


**Figure 16. PDR - Comparison of DSR based algorithm under RW**

**Table 7. Performance Analysis of Random Direction model based on DSR**

| Routing Protocols | Number of nodes | PDR (%) | Average Throughput (Kb/s) | Delay (Sec) |
|---|---|---|---|---|
| DSR | 50 | 64.74 | 75.87 | 0.04645 |
| | 100 | 62.58 | 74.95 | 0.04785 |
| | 150 | 60.52 | 70.84 | 0.04581 |
| | 200 | 74.25 | 65.21 | 0.05062 |
| DSR-GA | 50 | 68.55 | 77.84 | 0.03674 |
| | 100 | 50.47 | 75.47 | 0.04085 |
| | 150 | 68.13 | 73.86 | 0.04234 |
| | 200 | 75.48 | 74.87 | 0.04594 |
| DSR-ABC | 50 | 75.48 | 79.42 | 0.02763 |
| | 100 | 60.74 | 77.51 | 0.03524 |
| | 150 | 74.24 | 76.88 | 0.03674 |
| | 200 | 76.74 | 75.41 | 0.04032 |
| DSR-JA | 50 | 88.48 | 84.75 | 0.01285 |
| | 100 | 78.58 | 80.41 | 0.01634 |
| | 150 | 75.20 | 79.21 | 0.02085 |
| | 200 | 76.88 | 72.88 | 0.02521 |

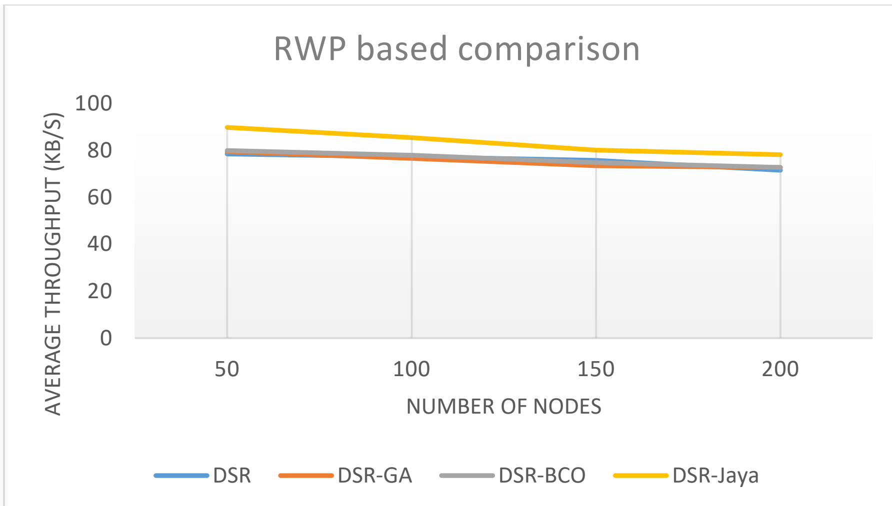


**Figure 17. Throughput - Comparison of DSR based algorithm under RWP**

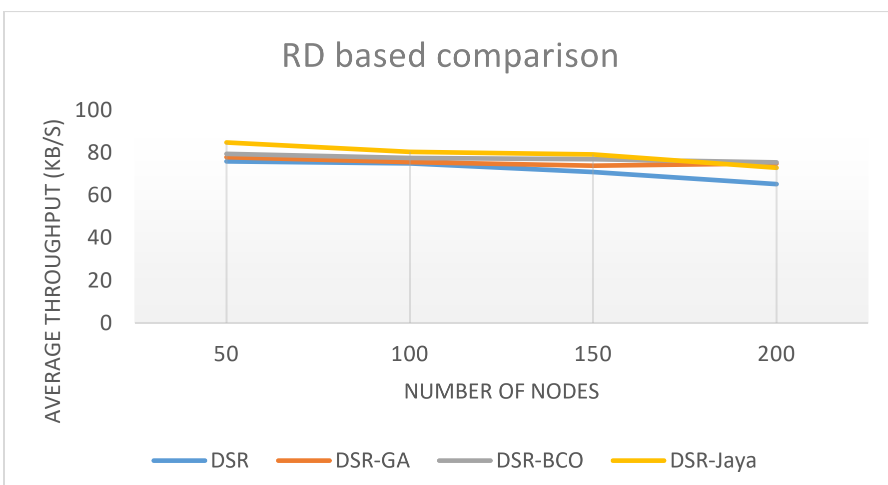


**Figure 18. Throughput - Comparison of DSR based algorithm under RD**

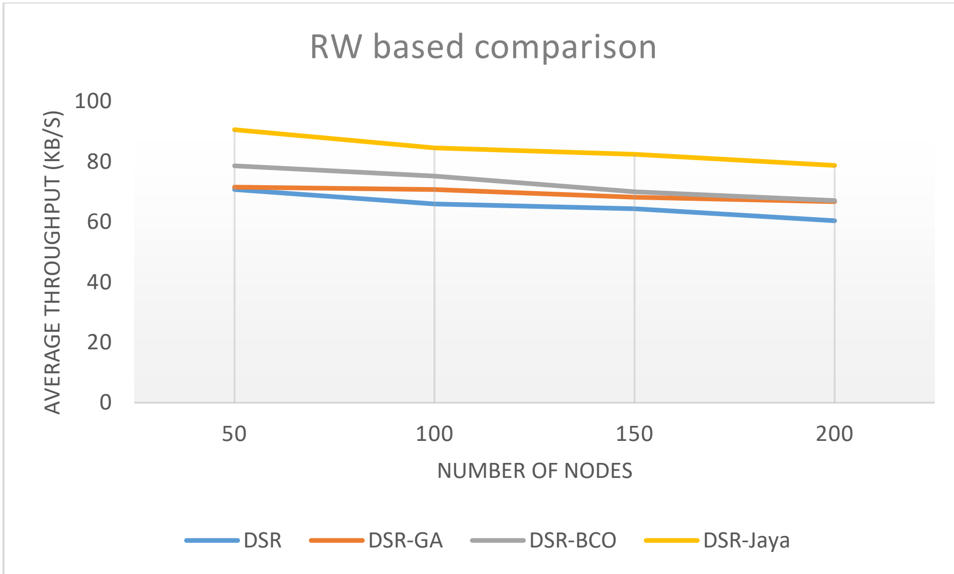


**Figure 19. Throughput - Comparison of DSR based algorithm under RW**

The performance of the algorithms AODV-JA, AODV-ABC, AODV-GA, and AODV are shown in Figures 3.4, 3.5, 3.6 for PDR and Figures 3.7, 3.8, 3.9 for Throughput, and Figures 3.10, 3.11, and 3.12 for delay respectively. In comparison with RD and RW mobility models, all the algorithms produce minimum PDR, Throughput, and maximum delay, i.e., below 70%. It is noted that in respect of DSR-JA and AODV-JA algorithms, some of the other algorithms (i.e., GA and ABC) give rise to maximum PDR and minimum delay than the AODV protocol. It is also inferred that the proposed AODV-JA and DSR-JA algorithms are more effective in finding the optimal path between mobile nodes. The proposed algorithm probably increases accuracy. To interpret the experimental results, we have empirically set the best accuracy to be above 82%.

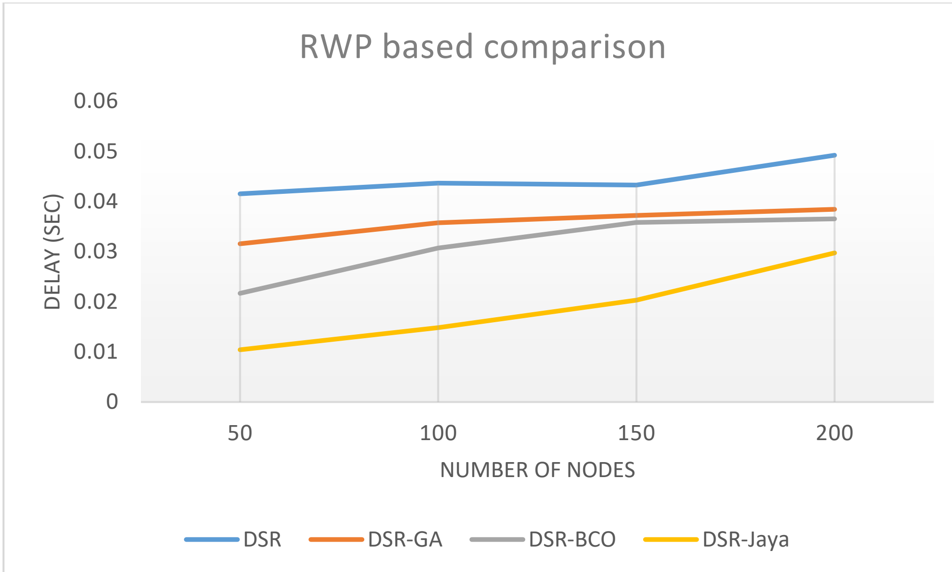


**Figure 3.19. Delay - Comparison of DSR based algorithm under RWP**

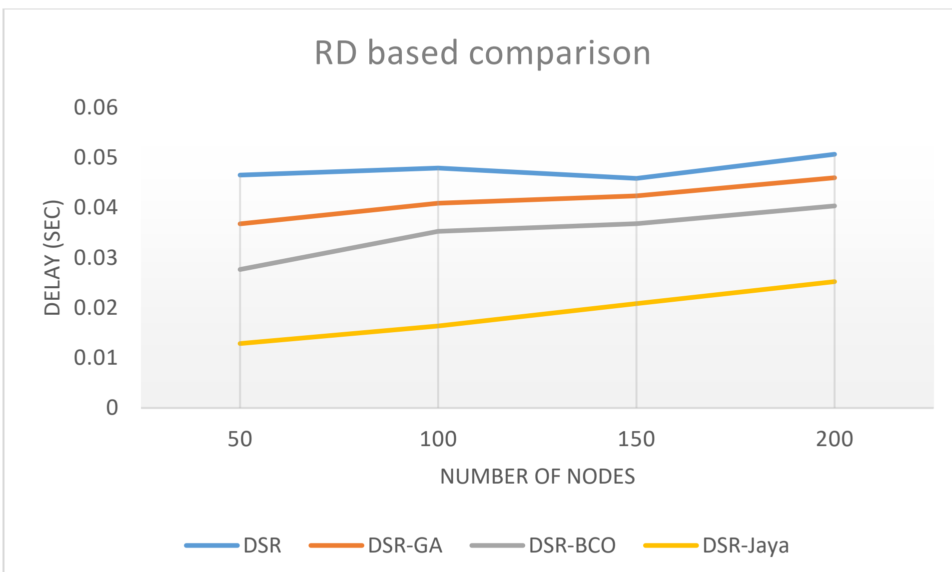


**Figure 3.20. Delay - Comparison of DSR based algorithm under RD**

**Table 3.7. Performance Analysis of Random Walks model based on DSR**

| Routing Protocols | Number of nodes | PDR (%) | Average Throughput (Kb/s) | Delay (Sec) |
|---|---|---|---|---|
| DSR | 50 | 60.74 | 70.78 | 0.05476 |
| | 100 | 42.58 | 65.95 | 0.05523 |
| | 150 | 45.2 | 64.36 | 0.05634 |
| | 200 | 30.45 | 60.41 | 0.05936 |
| DSR-GA | 50 | 65.44 | 71.56 | 0.04274 |
| | 100 | 40.21 | 70.82 | 0.04385 |
| | 150 | 41.75 | 68.25 | 0.04582 |
| | 200 | 32.71 | 66.74 | 0.04736 |
| DSR-ABC | 50 | 66.84 | 78.64 | 0.03632 |
| | 100 | 51.41 | 75.24 | 0.03884 |
| | 150 | 54.25 | 70.00 | 0.03984 |
| | 200 | 33.75 | 67.11 | 0.04084 |
| DSR-JA | 50 | 81.8 | 90.63 | 0.01222 |
| | 100 | 78.58 | 84.62 | 0.01584 |
| | 150 | 75.2 | 82.44 | 0.02035 |
| | 200 | 60.45 | 78.77 | 0.02567 |

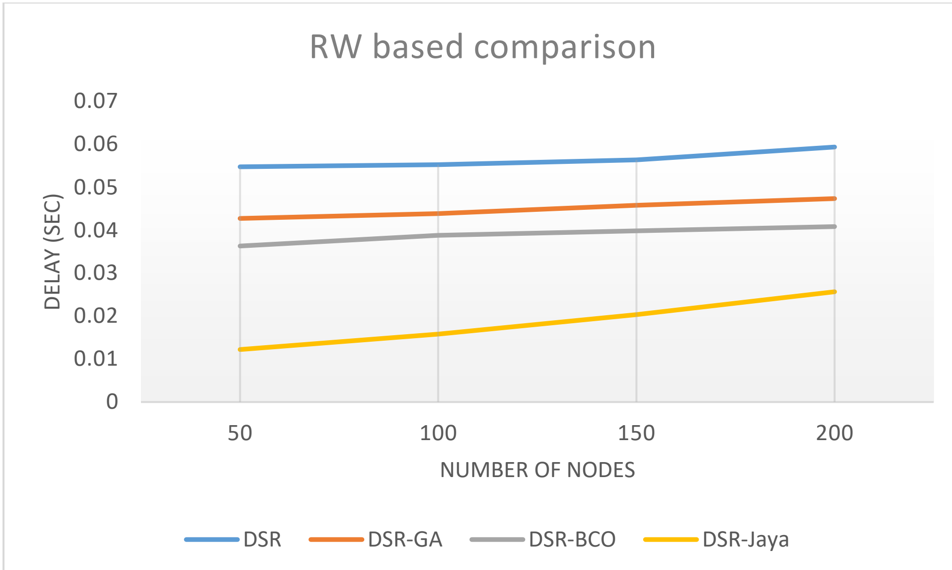


**Figure 3.21. Delay - Comparison of DSR based algorithms under RW**

The performance of DSR based algorithms are shown in Figure 3.13, 3.14, 3.15 for PDR, Figure 3.16, 3.17, 3.18 for throughput, and Figure 3.19, 3.20, 3.21 for the delay. From these figures, it is observed that the DSR-JA attain high throughput, PDR, and low delay.

## 7. Conclusion and future directions

A multi-constraint Jaya technique is presented in this paper to increase the performance of Ad hoc Networks. The suggested effort is solely focused on reducing a path's bandwidth and delay. This method chooses the best path from a network's available options. Because a single design measure is insufficient to forecast the path from source to many destinations, two design factors, bandwidth and delay, were employed in this study to discover an ideal path from accessible paths in a network. In addition, the performance of various mobility models, such as RWP, Random Direction (RD), and Random Walk (RW), is compared to two well-known ad hoc routing protocols. Throughput, packet delivery ratio, and latency are the performance metrics employed. The simulation findings show that the proposed DSR-JA and AODV-JA protocols outperform existing DSR and AODV protocols such as DSR-GA, DSR-ABC, AODV-GA, and AODV-ABC.

There are many fields for the expansion of this wireless network, based on the work discussed in this work. Some of the directions are mentioned for more study. In this work, soft computing techniques were employed to find a route between source and destination, and the performance of the algorithms was analyzed using evaluation metrics. It is suggested that in the future, an increasing number of mobile nodes could be developed to study the real scenarios more clearly and the route can be predicted using other soft computing methods. Moreover, it is possible to make use of optimization algorithms for better route selection in the wireless network.

## References


[1] T. Sudhakar and H. H. Inbarani, "Comparative analysis of indoor mobility scenarios creation (IMSC) in mobile ad hoc networks," *Indian J. Sci. Technol.*, vol. 9, no. 19, pp. 1–7, 2016, doi: 10.17485/ijst/2016/v9i19/93830.

[2] T. Sudhakar and H. H. Inbarani, "Intelligent Path Selection in Wireless Networks using Jaya Optimization," *Procedia Comput. Sci.*, vol. 133, pp. 976–983, 2018, doi: 10.1016/j.procs.2018.07.079.

[3] R. Magán-Carrión, R. A. Rodríguez-Gómez, J. Camacho, and P. García-Teodoro, "Optimal relay placement in multi-hop wireless networks," *Ad Hoc Networks*, vol. 46, pp. 23–36, 2016, doi: 10.1016/j.adhoc.2016.03.007.

[4] G. X. Kok, C. O. Chow, and H. Ishii, "Improving network coding in wireless ad hoc networks," *Ad Hoc Networks*, vol. 33, pp. 16–34, 2015, doi: 10.1016/j.adhoc.2015.04.002.

[5] F. Ducatelle, G. Di Caro, and L. M. Gambardella, "Ant agents for hybrid

multipath routing in mobile ad hoc networks," *2nd Annu. Conf. Wirel. On-Demand Netw. Syst. Serv. WONS 2005*, pp. 44–53, 2004, doi: 10.1109/WONS.2005.3.

[6] A. George *et al.*, "DOA: DSR over AODV routing for mobile ad hoc networks," *IEEE Netw.*, vol. 2, no. 1, pp. 46–55, 2004, doi: 10.1108/17427370910976052.

[7] T. Sudhakar, H. Hannah Inbarani, and S. Senthil Kumar, "Route classification scheme based on covering rough set approach in mobile ad hoc network (CRS-MANET)," *Int. J. Intell. Unmanned Syst.*, vol. 8, no. 2, pp. 85–96, 2020, doi: 10.1108/IJIUS-08-2019-0046.

[8] R. M. Ramadan, S. M. Gasser, M. S. El-Mahallawy, K. Hammad, and A. M. El Bakly, "A memetic optimization algorithm for multi-constrained multicast routing in ad hoc networks," *PLoS One*, vol. 13, no. 3, pp. 1–17, 2018, doi: 10.1371/journal.pone.0193142.

[9] N. Magaia, N. Horta, R. Neves, P. R. Pereira, and M. Correia, "A multi-objective routing algorithm for Wireless Multimedia Sensor Networks," *Appl. Soft Comput.*, vol. 30, pp. 104–112, 2015, doi: 10.1016/j.asoc.2015.01.052.

[10] A. W. Mohemmed, N. C. Sahoo, and T. K. Geok, "Solving shortest path problem using particle swarm optimization," *Appl. Soft Comput. J.*, vol. 8, no. 4, pp. 1643–1653, 2008, doi: 10.1016/j.asoc.2008.01.002.

[11] J. Jain, R. Gupta, and T. K. Bandhopadhyay, "Ant colony algorithm in MANET-review and alternate approach for further modification," *Commun. Comput. Inf. Sci.*, vol. 147 CCIS, pp. 344–347, 2011, doi: 10.1007/978-3-642-20573-6_60.

[12] D. Madhu Babu and M. Ussenaiah, "Cuckoo search and M-tree based multicast ad hoc on-demand distance vector protocol for MANET," *Int. J. Recent Technol. Eng.*, vol. 8, no. 2 Special Issue 3, pp. 876–879, 2019, doi: 10.35940/ijrte.B1164.0782S319.

[13] R. Venkata Rao, "Jaya: A simple and new optimization algorithm for solving

constrained and unconstrained optimization problems," *Int. J. Ind. Eng. Comput.*, vol. 7, no. 1, pp. 19–34, 2016, doi: 10.5267/j.ijiec.2015.8.004.

[14] S. Bitam and A. Mellouk, "Bee life-based multi constraints multicast routing optimization for vehicular ad hoc networks," *J. Netw. Comput. Appl.*, vol. 36, no. 3, pp. 981–991, 2013, doi: 10.1016/j.jnca.2012.01.023.

[15] N. Gunantara, "A review of multi-objective optimization: Methods and its applications," *Cogent Eng.*, vol. 5, no. 1, pp. 1–16, 2018, doi: 10.1080/23311916.2018.1502242.

[16] R. M. Dawes and B. Corrigan, "Linear models in decision making," *Psychol. Bull.*, vol. 81, no. 2, pp. 95–106, 1974, doi: 10.1037/h0037613.

[17] E. Science, G. Britain, and I. Engineering, "Flowshop scheduling problems are one of the most well known problems in the area of scheduling. The objective of minimizing the makespan is often employed as a criterion of flowshop scheduling since Johnson's work [1]. Various heuristic approaches (e.g., ," *Science (80-. ).*, vol. 30, no. 4, pp. 957–968, 1996.

[18] J. Jia, G. W. Fischer, and J. S. Dyer, "Attribute weighting methods and decision quality in the presence of response error: A simulation study," *J. Behav. Decis. Mak.*, vol. 11, no. 2, pp. 85–105, 1998, doi: 10.1002/(SICI)1099-0771(199806)11:2<85::AID-BDM282>3.0.CO;2-K.

[19] K. Fall and K. Varadhan, "The network simulator (ns-2)," *URL http//www. isi. edu/nsnam/ns*, pp. 2–3, 2007.

[20] M. G. And and C. de Waal, "A Mobility Scenario Generation and Analysis Tool," pp. 1–31, 2015, [Online]. Available: http://bonnmotion.net/.

[21] V. V. Mandhare and R. C. Thool, "Improving QoS of Mobile Ad-hoc Network Using Cache Update Scheme in Dynamic Source Routing Protocol," *Procedia Comput. Sci.*, vol. 79, pp. 692–699, 2016, doi: 10.1016/j.procs.2016.03.090.