Ad hoc routing comparison of DSDV, DSR, and GPSR

Johannes Kung David Larsson
Alrik Munoz Linus Roos
Linköping University
Linköping, Sweden

ABSTRACT

Ad hoc networks, consisting of decentralised wireless nodes with high mobility, provide flexible on-the-fly networks with application areas such as vehicular networks and communication in emergency situations. To achieve efficient communication over an inherently dynamic topology, a routing protocol needs to be carefully chosen depending on the purpose and topology of the network. The goal of this paper is to compare three routing protocols (Destination-Sequenced Distance-Vector, Dynamic Source Routing, and Greedy Perimeter Stateless Routing) used in ad hoc networks, and identify their respective advantages and disadvantages. This is achieved by running simulations of the presented routing protocols over example topologies and analysing the resulting data. The performance of a protocol is evaluated with the metrics throughput, routing overhead, average packet delay, and packet delivery ratio. This paper shows that the routing protocol GPSR outperforms both DSR and DSDV overall, but needs a highly connected network in order to scale well. DSDV is good for throughput and average packet delay if the network is small in size, while DSR performs poorly in all metrics in comparison to the other two protocols.

CCS CONCEPTS

• **Networks** → **Ad hoc networks**; **Routing protocols**.

KEYWORDS

Ad hoc networks, routing algorithms, DSDV, DSR, GPSR

1 INTRODUCTION

Ad hoc networks are decentralised wireless networks where each node participates in the routing process without the need for infrastructure such as access points. Most of the ad hoc nodes are mobile and free to move around in the network. Compared to conventional wireless networks, ad hoc networks can provide mobility and flexibility thanks to the short time required to set up and tear down the networks [2]. Ad hoc networks may also be more robust due to their distributed and non-hierarchical characteristics, further lowering economical costs in certain cases [2]. Moreover, as ad hoc networks support multi-hop communication, links between nodes are typically short which reduces interference and energy consumption compared to sending data in one hop over a large distance [2]. Because of this, ad hoc networks have great potential for areas such as emergency search-and-rescue situations where there is no time to set up infrastructure, as well as in vehicular networks for sharing crucial traffic information between moving cars on a long road [7]. However, with a constantly changing topology, ad hoc network protocols need to be carefully designed in order to successfully and reliably deliver data across the network.

In particular, the choice of a routing protocol has significant impact on the efficiency with which hosts can forward packets through the network. As both the purpose and topology of ad hoc networks differ depending on the situation, so does the most suitable choice of routing protocol. The challenge is to evaluate the performance of different routing protocols in different topologies using networking metrics such as throughput and latency.

In this paper three types of ad hoc routing protocols are compared – Destination-Sequenced Distance Vector (DSDV), Dynamic Source Routing (DSR) and Greedy Perimeter Stateless Routing (GPSR). These are simulated on different network topologies and the performance of each protocol is evaluated, taking several different factors into consideration in order to figure out the strengths and weaknesses of each protocols and when to use which one. The simulation shows how much congestion the different protocols create in a given network, as well as how well they handle host mobility. Knowing which protocol to use for which task based on network topology is important for network performance.

Similar analyses have been done before but with different sets of routing protocols [8]. Hamid and Mokhtar compared the performance of the three routing protocols AODV, DSDV and OLSR [1]. However, their simulation considered exclusively vehicular ad hoc networks (VANETs) which are characterised by incredibly high host mobility and churn, while this paper investigates networks that have a much lower level of volatility.

The remainder of this paper is structured as follows. Firstly, an introduction to the routing algorithms evaluated are given in section 2. Next, section 3 discusses the methodology. Section 4 contains the results and analysis of the findings. Lastly, section 5 contains the conclusion.

2 ROUTING PROTOCOL OVERVIEW

In general, most routing protocols for ad hoc networks can be categorised into two types – reactive routing protocols and proactive routing protocols [7]. The latter involves hosts periodically sharing routing information among each other, in order to keep their routing tables up-to-date and to always reflect the currently available routes. Reactive routing protocols on the other hand, discover the most appropriate route only when needed. Thus, no traditional routing table is used in these types of protocols. As reactive protocols do not require constant updates of routing information among hosts, overhead traffic will typically be smaller compared to proactive protocols over time.

However, not all routing protocols can be categorised into these two types. There exists routing protocols that can be seen as hybrids between the two categories, as well as those that do not fit into either of them. An example of a protocol that cannot be categorised as reactive, proactive, or anything in between, is GPSR.

2.1 DSDV

The DSDV protocol is a proactive protocol [7], that requires that each node in the network stores a routing table and that they periodically transmit their routing tables to neighbouring nodes [9]. An entry in a routing table at a node contains the destination node of the route, the next-hop node through which the destination can be reached, a metric which usually is the hop count to the destination, and a sequence number initially created by the destination node. A DSDV table contains entries for all available destination nodes.

The protocol works by having each node send its routing table to its neighbours. The neighbour then uses this routing table to update its own table. Typically, when node A receives the table of its neighbour B, it will check if B's hop count to destination x is less than A's current hop count. If it is, A will update its entry for destination x to forward along the new route through node B instead of its old route and update it with the new hop count. This approach would typically have a problem with loops in the topology graph, but DSDV solves this problem with the aforementioned sequence numbers. Whenever a node sends its routing information, it increments the sequence number in the entry for itself by two. When a node receives a table from a neighbour, it will compare the table to its own to see if any route should be updated. Before updating its routing table, a node will also check the current sequence number, and an entry is updated only if the received sequence number for the destination is newer.

When a node discovers that it is no longer receiving routing updates from a neighbouring node, it considers the link between them as broken. The cost to the neighbour is set to infinity and the sequence number of the entry is incremented by one. The node then immediately announces the change to its neighbours. In order to reduce communication overhead in the face of a dynamically changing topology, a node will only occasionally broadcast its full routing table, such as whenever there is a significant update to the table, and instead usually sends smaller incremental updates to keep routing information consistent.

2.2 DSR

DSR is a reactive protocol, which means that instead of maintaining a constantly updating routing table like DSDV does, it only discovers new routes as they are needed. When host A needs to send a packet to host B, it broadcasts a Route Request packet (RREQ) to all its neighbours [3]. The neighbours in turn forward the RREQ by broadcasting it to all of their neighbours, even back the way it came. The RREQ contains a unique ID, a source node, a destination node, and a list of the traversed path. When a host receives a RREQ, it first checks an internal cache of recently handled RREQs. If it has already handled the RREQ (like when a neighbour sends it straight back immediately) or if the host is already listed in the traversed path, it simply drops the RREQ. If it has not handled the RREQ, the host checks whether it is the intended destination. If it is, it sends a Route Reply packet (RREP) back to the source using the RREQ list of traversed nodes to provide a complete route. If the host is not the intended destination, it simply adds itself to the list of traversed nodes and broadcasts the RREQ along to its neighbours [4].

2.3 GPSR

The GPSR protocol is neither a proactive nor reactive protocol as unlike DSDV or DSR it does not have a routing table and does not maintain any information about other hosts outside of a host's direct neighbours. GPSR instead relies on the geographical positions of the hosts in the network when sending packets [5]. The protocol primarily uses *greedy forwarding* when forwarding packets, which means a host always tries to send the packets to its neighbour with the closest geographical location to the destination.

The simplicity of greedy forwarding means that it can also easily fail, if for example there is an obstacle between the source and the destination, or this specific host in the topology happens to not have any neighbours closer to the destination. In such a case, GPSR switches to its secondary forwarding mechanism, perimeter forwarding. In perimeter forwarding the packet is sent to the first neighbour sequentially in a counter-clockwise rotation from the previous host. Basically if the packet cannot move "forward", closer to its destination, it moves "sideways" in the network in search of other routes.

3 METHODOLOGY

To measure the performance of the examined protocols, the four following metrics have been used: throughput, routing overhead, average packet delay, and packet delivery ratio. In order to collect data for each metric, each protocol is simulated on the same set of topologies. The simulations consist of four different topologies to evaluate if, and by how much, the measurements change depending on the network layout. The following paragraphs explain each metric and how it is measured, as well as a summary describing the details of the simulation and the simulated scenarios.

Throughput: The throughput of a network measures the rate at which data passes through the network [6], usually measured from one end to another. Since packets can drop or change route through the network, throughput has to be measured with regard to only the data that actually arrives at the final destination.

Routing overhead: A protocol may require routing information in packet headers, or packets dedicated solely to the communication of routing information. As headers for routing are generally small in size and do not noticeably affect throughput at a particular host, routing overhead will in this paper only take packets dedicated strictly to routing into consideration. These packets usually involve announcing or finding new routes, or maintaining information about already existing ones. The number of these packets and the frequency at which they are sent define the routing overhead of the protocol. The routing overhead of a protocol is calculated by the amount of routing information traffic in proportion to the total traffic (including actual data) on the network over a period of time.

Average packet delay: A packet experiences many different types of delay on its journey through a network. The most significant of these are transmission delay (how fast a host can push a packet onto an outgoing link), propagation delay (the time it takes for a packet to be transmitted over a link), queuing delay (the time a packet spends in a host buffer), and processing delay (the time it takes for a host to process a packet). These delays are present at every host throughout the network. The total average packet delay is the average of the sum of these delays for all data packets.

Given name	Node count	Node transmission range
Sparse small	50	120 m
Dense small	50	160 m
Sparse large	250	40 m
Dense large	250	80 m

Table 1: Parameters for topologies used

Packet delivery ratio: Occasionally, packets in the network are lost. This can for instance happen due to congestion or link failure. The packet delivery ratio describes the ratio of packets successfully delivered to their end destinations out of the total amount of packets sent. To measure this ratio, the number of packets sent and the number of packets that arrive at their destination are recorded. Only data packets will be considered for this metric, meaning routing information packets (such as DSR RREQ packets) will not be counted towards this metric.

To collect data about these metrics, a simulator[1] has been constructed which can simulate an ad hoc network in operation. For a given topology, a corresponding ad hoc network has been simulated using one of the routing protocols. The fundamental characteristics of an ad hoc network, where geographical locations of the nodes determine the layout of the network in terms of connections, make both random graphs and scale free graphs unsuitable for modelling them [2]. Instead, simply randomly distributing a number of nodes over a certain area (in our case, 500 metres by 500 metres) and letting nodes close to each other have a link between them will result in a graph that better represents an ad hoc network. Using this method to generate appropriate topologies, a set of topologies with specific characteristics have been chosen to be used in the simulations. Increasing the maximum possible transmission range of hosts makes the network connectionally denser, meaning more links form between hosts in the network, while increasing the number of hosts makes the network larger. Table 1 shows the exact attributes of the four chosen topologies for the simulation and Figure 1 shows these topologies graphically.

The simulator uses a predetermined set of events when running. An event is either a `send` event, meaning a given host shall attempt to send a packet to another specified host in the network, or a `move` event, meaning a given host shall attempt to move to a specified location. For each topology, five sets of such events were generated, each with 200 `send` events and 10 `move` events in random order. The reason for using five different sets of events for each topology is to ensure a higher accuracy in the final results. The simulator runs the next event a certain amount of time after the previous one, until there are no more events to run. A packet created as a result of a `send` event is referred to as a *data packet* while a packet created by a host for routing purposes is a *routing packet*. The simulation exits when all data packets have either arrived at their destination or been dropped.

To reduce the complexity of the simulation, it has been limited to homogeneous ad hoc networks only. In a homogeneous ad hoc network, properties such as maximum distance to a neighbour does not differ between nodes. The simulation also does not alter the rate at which data is sent over links in order to further simplify the implementation. The simulation focuses on the routing aspect of ad hoc networks using a simple model of packets, rather than implementing a real world internet stack. This is to say functionality at for instance the link layer for forwarding packets from a host to its neighbour is not simulated, instead a packet simply arrives at a host after spending an appropriate amount of time on the link to it.

[1]Simluator code: https://github.com/TheSolarz/ad-hoc-routing-simulator

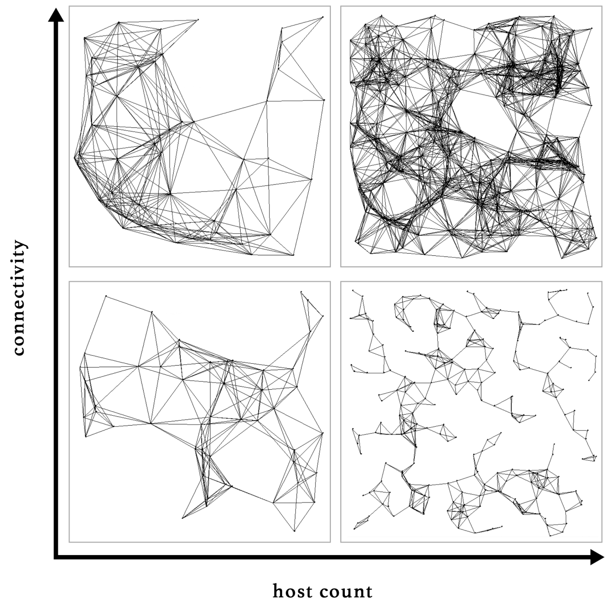

Figure 1: The four topologies used for the simulations

4 RESULTS AND ANALYSIS

The simulations have been executed without any pre-installed routing information in any host. As a result, DSDV and DSR which rely heavily on pre-cached information have a slow start-up time and can instantaneously flood the network upon startup with route discovery or broadcasts, causing unrealistic packet loss for data packets. At any meaningful point of observation, a real ad-hoc network will already have existed for a period of time and in the case of DSDV and DSR have already (partially) established routing caches. This is somewhat mitigated in DSDV's case by a join-phase at the start of each simulation, where all hosts join the network one by one, building links and getting to know their neighbours before send events are started.

For each metric, Figures 2, 3, 4, and 5 show the average result over the five sets of events for each topology and routing protocol. A 95% confidence interval has been calculated and is shown for each result.

4.1 Throughput

Figure 2 shows the measured throughput of the three protocols for each topology. GPSR and DSDV have a roughly equal throughput with a low decrease as host count or connectivity grows. Also shown is that DSR has a consistent throughput regardless of topology that is considerably lower than that of GPSR and DSDV. A notable edge case shown in the figure is the large sparse topology, where the throughput of GPSR is substantially lower than for other topologies. The reason for why this could be the case is discussed in the Average packet delay subsection ahead. In the same topology, DSDV appears to have a substantially higher throughput compared to in other topologies and other protocols. This however, is likely only the case because of its low delivery ratio, as can be seen in Figure 5. The network gets congested and the few packets that do arrive, do so quickly, because the more time the packets spend out in the network, the higher the risk it is for them to be dropped due to congestion in DSDV's case.

4.2 Routing overhead

Figure 3 shows the observed routing overhead of the three protocols for each topology. The figure shows that GPSR always has a routing overhead of 0%, DSR has a routing overhead close to 100% in all topologies, and DSDV has a routing overhead around 90% that grows towards 100% as either host count or connectivity increases.

GPSR is observed to keep a negligible routing overhead compared to that of DSR and DSDV. Whereas there are no packets sent purely for routing in GPSR, DSR and DSDV on the other hand produce a large amount of pure routing packets during route discovery even when sending only a few data packets. Per the simulations that have been run, DSR and DSDV had a routing overhead of above 90% in all four topologies, making for a lot of unnecessary overhead for very little actual data sent. However, this ratio can be expected to decrease over time in more lengthy simulations as the hosts in the network will have learnt more routes. Not taken into consideration in the graph is the overhead for all three protocols caused by the need for storing extra information in the headers of data packets in order to route them correctly.

4.3 Average packet delay

The observed average delay of the three protocols for each topology is shown in Figure 4. It can be seen in the figure that GPSR has a low average packet delay in most topologies except in the large, sparse one. DSDV is shown to have a low and roughly equal average delay regardless of topology, on par with the average delay of GPSR. As for DSR, the figure shows an average delay around two to three times larger than that of GPSR and DSDV in each topology. No certain patterns can be seen in the figure for DSR when either host count or connectivity increases.

The outlying case of GPSR occurs on the topology shown in the bottom right of Figure 1 – a tree-like topology with long narrow branches of connected hosts leading to nowhere. In such a topology, GPSR experiences massive delays when forwarding packets in perimeter mode. In the worst case scenarios, packets are forced to traverse all the way out and then back in again on most branches. Although GPSR performs well under most metrics in the tested topologies, it is clear that GPSR needs a well connected topology as the number of hosts increases in order to maintain a good performance. This flaw will remain the same throughout the whole lifetime of a GPSR network. Meanwhile in a DSR or DSDV network, the average packet delay will decrease over time as frequently used routes become cached. The caching of routes results in hosts needing to send fewer routing packets and a smaller average packet delay as the network ages. However, a high degree of host mobility can negate this.

4.4 Packet delivery ratio

Figure 5 shows the packet delivery ratio of the three protocols for each topology. GPSR has a near 100% delivery ratio in all topologies except for the large, sparse topology where the ratio is around 60%. In addition, the figure shows that DSR has a low delivery ratio around 10% which slowly decreases as either host count or connectivity increases. As for DSDV, the figure shows a delivery ratio of around 25% for the small topologies but a significant drop to around 5% in topologies with a higher host count.

In the small topologies as well as the large, dense one, the networks stay mostly connected and GPSR should thus always succeed to deliver packets under the premise that no packet will be dropped due to congestion. Since GPSR has very low routing overhead, congestion is kept to a minimum when there is little traffic over the network, meaning GPSR will have close to a 100% delivery ratio in most cases. DSDV and DSR however, are not as reliable and can easily fill the buffers of hosts with their routing packets. Dense topologies amplify this effect, causing more congestion and loss of packets. Packet drops due to congestion in DSDV and DSR will on the other hand decrease and the average packet delay will improve as the network ages, assuming the movement of hosts is kept to a minimum.

In the large, sparse topology, movement of hosts often lead to many disconnected parts of the network, preventing some packets from ever reaching their destination. This decrease in delivery ratio is most evident for GPSR but can also be seen for DSDV.

5 CONCLUSION

While ad hoc networks are desirable due to their distributed and flexible on-the-fly properties, the routing protocol chosen for a particular network determines how well the network scales as the number and connectivity of hosts grow in a dynamic network topology. This paper has compared the three ad hoc routing protocols GPSR, DSDV, and DSR through simulations under topologies of varying host count and connectivity to determine which protocol is the most suitable for each case. The outcome of the simulations has shown that GPSR performed well overall, but specifically needed a highly connected topology in relation to the network size in order to scale well. DSDV generally performed on par with GPSR (and even surpassed it in a large, sparse network) in terms of throughput and average packet delay, but it suffered greatly in terms of overhead and packet delivery ratio – especially in large topologies. DSR had comparatively bad performance in all metrics. Both DSDV and DSR metrics were impacted by network flooding during the early stages of the network's lifetime, but the impact was more evident in the overall poor performance of DSR.

Figures 2, 3, 4, and 5, show that GPSR performed well in all cases, with one exception: a large network with low connectivity. This is due to the perimeter forwarding of GPSR, which suffered from the fact that the packet can be routed into a "dead end" of the topology and had to be routed back the way it came. This issue became more evident in large, sparse topologies. Figure 4 shows that the average packet delay increased dramatically for GPSR in such a topology, and Figure 2 shows that the throughput also decreased considerably. Thus, GPSR is not well suited for networks with low connectivity if throughput and average packet delay are important

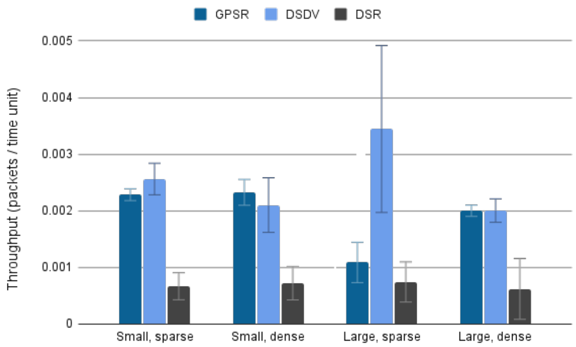

Figure 2: Throughput of the routing protocols GPSR, DSDV and DSR on four different topologies.

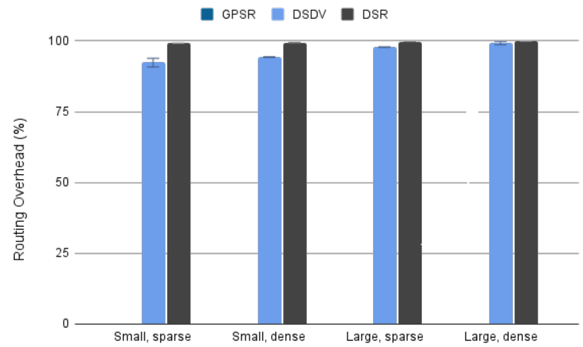

Figure 3: Routing overhead of the routing protocols GPSR, DSDV and DSR on four different topologies.

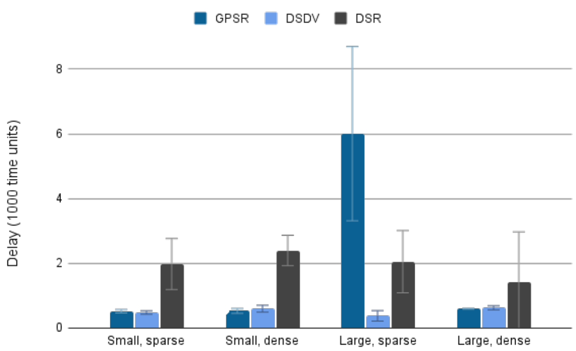

Figure 4: Average packet delay of the routing protocols GPSR, DSDV and DSR on four different topologies.

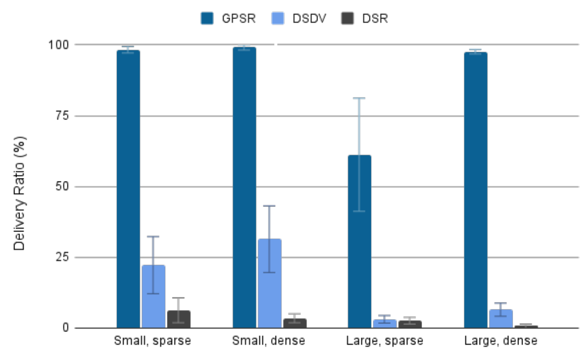

Figure 5: Packet delivery ratio of the routing protocols GPSR, DSDV and DSR on four different topologies.

metrics. However, if the only important metric is packet delivery ratio or a low routing overhead, GPSR is well suited for the task regardless of topology.

Future work may benefit from more detailed experimentation with the different variables in the simulation, to try and find more optimal values for each of the protocols in different kinds of topologies. Factors such as the rate at which packets are sent between hosts, the time to live for a cached route in DSR, or how often DSDV hosts broadcast their routing tables, might impact the conclusions about the performance of each protocol and when to use which. A shorter interval between DSDV routing table broadcasts would for example cause more traffic over the network, but allow the network to more quickly detect and recover from broken links caused by host mobility. In some network topologies, detecting broken links with a shorter broadcast delay could increase the overall delivery ratio despite the increased congestion. Future work should also take into consideration how the prevalence and frequency of host mobility impact the performance of different routing protocols and topologies. For more realistic data, the simulation may also be expanded to cover heterogeneous networks, three-dimensional network topologies, and the volatility of wireless communication (such as jitter). The simulation sends data at a constant rate, which is not completely representative of a real network. Future simulations should be expanded to also contain occasional bursts of data alongside the steady data streams, as that better represents real ad hoc networks.

REFERENCES

[1] B. Hamid and E.-N. E. Mokhtar. Performance analysis of the vehicular ad hoc networks (vanet) routing protocols aodv, dsdv and olsr. In *Proceedings of International Conference on Information Communication Technology and Accessibility (ICTA)*, pages 1–6. IEEE, 2015.

[2] R. Hekmat. *Ad-hoc networks. fundamental properties and network topologies.* Springer, 1 edition, 2006. ISBN 9781402051661.

[3] D. B. Johnson, D. Maltz, and J. Broch. Dsr: The dynamic source routing protocol for multi-hop wireless ad hoc networks. In *Ad Hoc Networking*, pages 139–172. ACM, March 2001.

[4] M. D. Johnson D, Hu Y. The dynamic source routing protocol (dsr) for mobile ad hoc networks for ipv4. RFC 4728, IETF Datatracker, February 2007.

[5] B. Karp and H. T. Kung. Gpsr: Greedy perimeter stateless routing for wireless networks. In *Proceedings of the Annual International Conference on Mobile Computing and Networking*, MobiCom, page 243–254, 2000. ISBN 1581131976.

[6] J. Kurose and K. Ross. *Computer Networking: A Top-Down Approach.* Pearson, 2016. ISBN 9780133594140.

[7] J. Loo, J. Lloret Mauri, and J. H. Ortiz. *Mobile ad hoc networks : current status and future trends.* CRC Press, 2012. ISBN 9781439856512.

[8] A. A. Mahamune and M. M. Chandane. Evaluating routing protocols for mobile ad hoc networks under varying network scenarios. In *Proceedings of International Conference on Intelligent Communication Technologies and Virtual Mobile Networks (ICICV)*, pages 220–225, 2021.

[9] C. E. Perkins and P. Bhagwat. Highly dynamic destination-sequenced distance-vector routing (dsdv) for mobile computers. *ACM SIGCOMM Computer Communication Review*, 24(4):234–244, 1994. ISSN 0146-4833.