Mobility-based Candidate Selection and Coordination in Opportunistic Routing for Mobile Ad-Hoc Networks

by

Mohammad Tahooni

Thesis submitted to the
Faculty of Graduate and Postdoctoral Studies
In partial fulfillment of the requirements
For the M.A.Sc. degree in
Electrical and Computer Engineering

School of Electrical Engineering and Computer Science
Faculty of Engineering
University of Ottawa

Abstract

Opportunistic Routing (OR) is an effective and enhanced routing scheme for wireless multihop environment. OR is an approach that selects a certain number of best forwarders (candidates) at each hop by taking the advantage of the broadcast nature of the wireless medium to reach the destination. When a set of candidates receive the packet, they coordinate with each other to figure out which one has to forward the packet toward the destination. Most of the research in this area has been done in mesh networks where nodes do not have mobility.

In this survey, we propose a new OR protocol for mobile ad hoc scenarios called as Enhanced Mobility-based Opportunistic Routing (EMOR) protocol. To deal with the node mobility, we have proposed a new metric which considers the following: geographical position of the candidates; the link delivery probability to reach them; the number of neighboring nodes of candidates; and the predicted position of nodes using the motion vector of the nodes. We have compared EMOR with five other well-known routing protocols in terms of delivery ratio, end-to-end delay, and expected number of transmissions from source to the destination. Our simulation results show that proposed protocol improves delivery ratio and number of expected transmission in terms of different type of mobility models.

Acknowledgements

I would like to express my special appreciation to my supervisor Professor Azzedine Boukerche, who has been a tremendous mentor to me. I would like to thank him for encouraging my research and for allowing me to grow as a research member at PARADISE research laboratory.

I would also like to thank Amir Darehshoorzadeh, who often shared his knowledge and experience during this research. In addition, a special thanks goes out to my friends at PARADISE research laboratory for all of the advice and feedback. It was my pleasure to study among them.

Last but not least, my deepest gratitude to my father, mother, and my family for all of their support. Absolutely, there are not enough words to express my appreciation to them.

Publication

Conference Paper:

- M.Tahooni, A.Darehshoorzadeh, A.Boukerche, Mobility-based Opportunistic Routing for Mobile Ad-Hoc Networks. in: PE-WASUN'14 (ACM), September 2014, Montreal, QC, Canada.

Table of Contents

List of Tables

List of Figures

Abbreviations

ACK	Acknowledgement	**CS**	Candidate Set
CTS	Clear To Send	**DP**	Distance Progress
EAX	Expected Any-path Transmission	**EDC**	Estimated Duty Cycled wake-up
EDP	Expected Distance Progress	**EOT**	Expected One-hop Throughput
EPP	Expected Predicted Progress	**ETT**	Expected Transmission Time
ETX	Expected Number of Transmission	**GPS**	Global Positioning System
GP	Geographical Progress	**MAC**	Media Access Control
MANET	Mobile Ad Hoc Network	**NC**	Network Coding
NS-2	Network Simulator 2	**OR**	Opportunistic Routing
RTS	Request To Send	**SIFS**	Short Inter Frame Space
STR	Successful Transmission Rate	**VANET**	Vehicular Ad Hoc Network
XOR	Exclusive-OR		

Chapter 1

Introduction

In this chapter, we introduce a brief definition and history of Opportunistic Routing over wireless ad hoc networks, the challenges and issues, and effect of mobility over performance of routing algorithms. Then, we explain our motivation and objective. Finally, we mention the thesis organization.

1.1 Background and Overview

In traditional Mobile Ad hoc Networking (MANET) routing protocols the next hop forwarders are chosen before transmitting, and packets are traversed on a pre-specified path toward the destination [5][8][42][9][19]. In this manner, wireless links have been treated as point-to-point wired links and the broadcast nature of the wireless medium has been ignored. In contrast, Opportunistic Routing (OR) uses the broadcast feature of the wireless environment to enhance the performance of routing. Generally, the basic idea of OR is overhearing of transmission of the nodes in wireless medium and coordination of them in such a way that the packets reach the destination. Due to this fact, OR is a suitable technic for wireless multihop networks such as mobile ad hoc network, vehicular ad hoc network (VANET), and sensor network. Increasing the network reliability and reducing

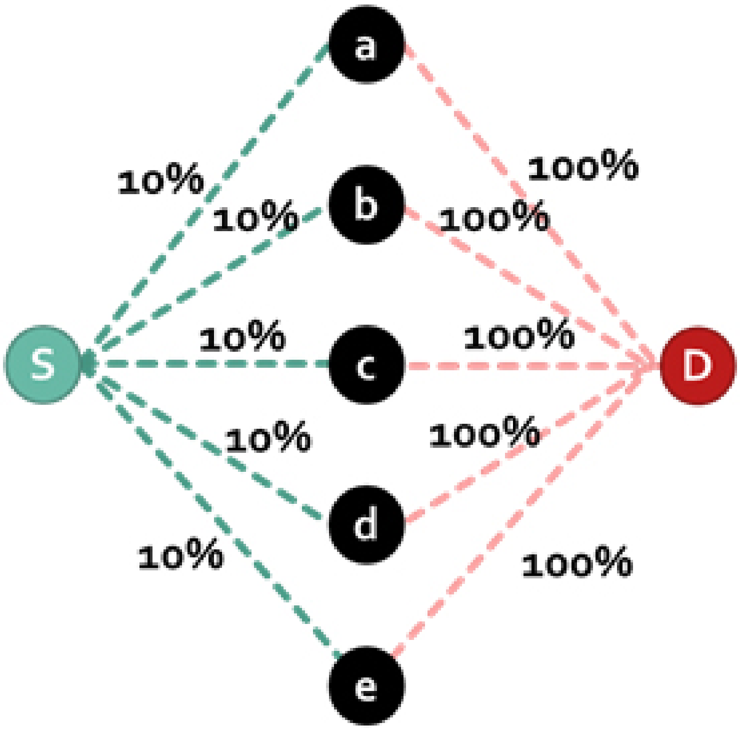


Figure 1.1: Comparison of end-to-end delivery probability

the number of transmissions are the main advantages of using OR. Both of the analysis results [52][37] and recent experiments [4][13][43] show that OR has the potential to perform better than traditional routing in most scenarios [25][6], especially in the environments with high percentage of packet loss.

Figure 1.1 shows an illustration of increasing reliability by using OR concept rather than traditional routing. As one can see on Figure 1.1 end-to-end delivery probability in terms of opportunistic routing concept is: $P_{OR} = (1–(1–0.1)^5) \times (1) \approx 0.4$ which means the probability that at least one of the forwarders routes the packet toward the destination. It is almost four times better than the probability of traditional way routing which is: $P_{Tradition} = (0.1) \times (1) = 0.1$

Increasing reliability and reducing the number of transmissions as an other advantages of OR are shown in Figure 1.2. For example, consider that node S wants to send a packet to node D and node b is supposed as a next hop forwarder because it is closer to the

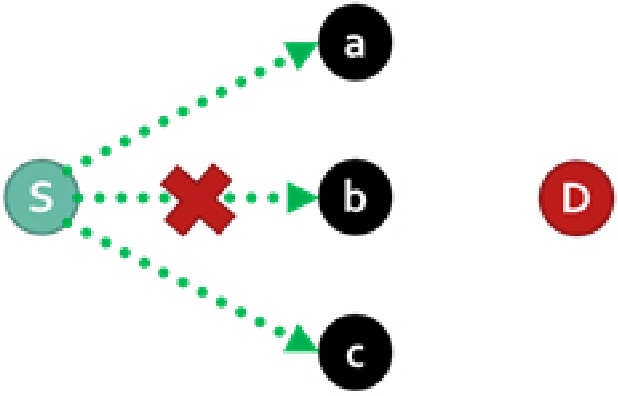


Figure 1.2: Increasing reliability by employing OR

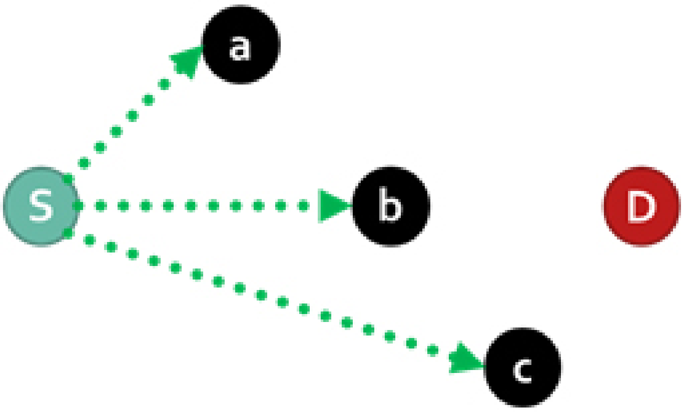


Figure 1.3: Enhancing transmission range and latency by employing OR

destination. In terms of the OR concept, if b cannot receive the packet for any reason, node a or c can do its task by overhearing the packet. Hence, there is no need for re-transferring the packet by the source. Meanwhile, reliability of transmission has been raised.

In Figure 1.3, the effect of OR over enhancing transmission range is illustrated. Suppose that b is considered as a next hop forwarder. However, the packet may be received by c as well when the sender transmits the packet toward the b. In this case, node c, which is closer to the destination, may relay the packet and total latency and transmission range will be improved.

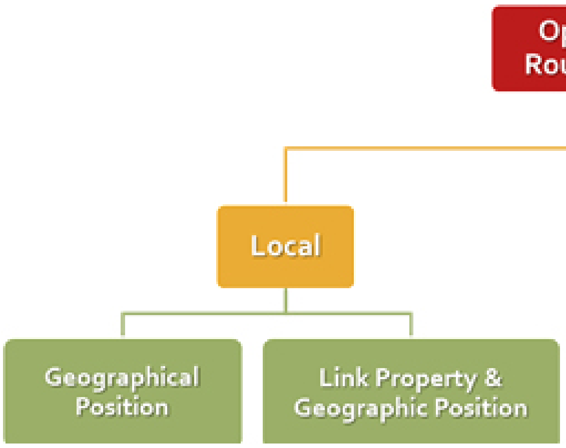


Figure 1.4: Classification of OR metrics

OR consists of three main parts: metric calculation, candidate selection and prioritization, and candidate coordination. Metric calculation mechanisms generally are divided into local and end-to-end methods, as shown in Figure 1.4. Local or hop-by-hop metrics consider only the information of the local neighboring nodes to forward the packets. On the other hand, in end-to-end approaches, the information and states of the all nodes are considered to select the best route from source to the destination. Although the end-to-end method is more efficient and lead to the optimal result, it is difficult to carry the information and state of the whole topology in a mobile scenarios, which have frequent topology changes. Therefore, it is obvious that the local trend is more suitable for dynamic networks. In the local approaches, beacon messages are broadcasted regularly so that each node gets some information about its neighbors. In this case, nodes only consider the information of their local neighbors for making the decision to select the candidates. We will discuss metric in more details in chapter 2.

As we have mentioned above, the other part of OR is candidate selection. In this phase, the sender node specifies a set of capable nodes as the candidates set to forward the data packet and sorts them according to a metric. The sender puts the candidates set in the header of the data packet and broadcasts it. Note that the selected candidates can be prioritized based on expected transmission count (ETX) [4], hop count to the destination, geo-distance and so on (see [2][7][11][10] for more information). When the data packet is received by the candidate, it checks the packet header to figure out whether its ID exists in the header of the packet or not. In case of existence, it will enter the coordination phase. Several coordination methods have been proposed to handle this part, such as timer-based, Acknowledgment-based, RTS-CTS, and network coding [2]. Figure 1.5 illustrates the OR coordination classification. One of the most used approaches for the candidate coordination is the timer-based coordination, whereby each candidate waits for a specific time before forwarding the packet. Then, each candidate will transmit the packet if it does not hear the same packet from other higher priority candidates during the waiting time. Otherwise, the candidate simply discards the data packet. This procedure continues until the data packet received by the destination.

As another coordination method, in request-to-send/clear-to-send (RTS/CTS) scheme, potential forwarders response in a priority order [57]. Once the first CTS is received, the sender starts transmitting the data packet to the related node. Duplicate transmissions can be omitted by using this method. Another coordination strategy is to enable candidates to send back a short acknowledgment (ACK) as long as they received the data packet in a specified order. The first node returning an ACK packet wins the competition and the other forwarders resign. The challenging point of this strategy is that all of the forwarding candidates have to be neighbor so that they can hear one another's ACK transmission. Among existing works, there are even some mechanisms [13][45][50][38] that do not require coordination between candidates. In these mechanisms data packets are simply broadcast to the air. Although these mechanisms can ease the operation of MAC layer and enhance the latency, it is obvious that they can cause tremendous increases in the number of duplicated transmissions.

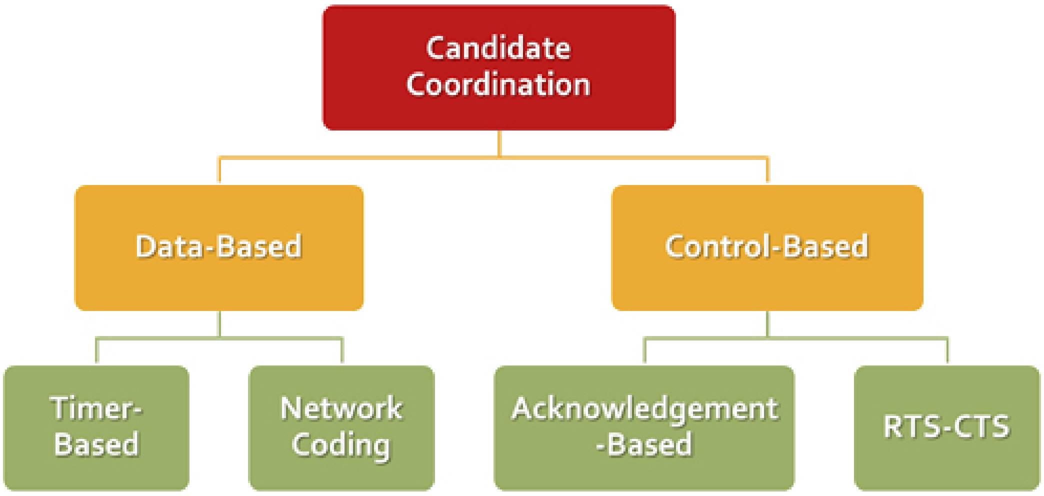


Figure 1.5: Classification of candidate coordination in OR

1.2 Motivation and Objectives

Most of the work on OR has been done for static mesh networks where nodes do not have mobility. When nodes are mobile, candidate set changes frequently with dynamic topology and updating the whole topology information is very costly. Therefore, a fast and efficient candidate selection and coordination algorithm is needed in the mobile scenarios in order to adapt rapidly with the frequent topology changes. Moreover, choosing an efficient metric that considers the mobility properties has a great impact on the performance and complexity of an OR protocol. In this survey, first, we propose an enhanced mobility-based opportunistic routing. Then, we use NS-2 simulation to display the enhanced performance of our proposed protocol in terms of delivery ratio and expected number of transmission in mobile scenarios.

1.3 Thesis Organization

The remainder of this thesis is organized as follows:

- Chapter 2 presents literature review over the existing opportunistic routing parts and protocols; and it discusses their classifications, metrics, features, drawbacks, and challenges.

- Chapter 3 proposes the mechanism of our Enhanced Mobility based Opportunistic Routing (EMOR) protocol.
- Chapter 4 undertakes to investigate the performance evaluation of our proposed protocol in compare comparison to the other famous routing protocols.
- Chapter 5 presents the conclusion and the future works of this survey.

Chapter 2

Literature Review

In this chapter, we talk about the main parts of opportunistic routing that form the OR protocol, such as metric, candidate selection and coordination at first. Then, we indicate the existing classifications and challenges of each part. At last, some major and well-known opportunistic routing protocols are introduced according to the mentioned parts. Moreover, their metrics, features, and drawbacks are discussed.

2.1 Routing Metrics in Opportunistic Routing

One of the main ideas of employing OR is reducing the expected number of data transmissions, which logically represents the number of data transmissions required over the entire network to deliver a data packet from the source to the destination. The total latency, jitter, and energy consumption will be enhanced if the number of transmissions is decreased [2]. Therefore, the role of a good metric that can consider the conditions and select an effective candidate set for forwarding the packet is significantly important over the performance of OR. In addition, prioritization of candidate forwarders is another factor to reduce the total latency and the expected number of data transmission. Generally, OR metrics are classified as local and end-to-end (topology-based) metrics. In local metrics, the selection and prioritization of the next hop forwarders are done by using the information of neighboring nodes, such as their geographical distance or link delivery probability. On the other hand, end-to-end metrics employ the information of the whole network to figure out the best forwarders and form the candidate set. In other words, the calculation of metric in each forwarding node is based on the remaining path from it to the destination. Generally, using the end-to-end approaches can lead one to an optimal solution owing to the fact that the nodes know the entirety of the information of the network. Although the end-to-end metrics seem good, they also can cause significant overhead due to topology changes and computation costs. In the following sections, usual metrics that have been used in the literature will be described and classified.

2.1.1 Local Opportunistic Routing Metrics

The local metrics are only based on the information of neighboring nodes. One of the simplest geographical local metrics is Distance Progress (DP) which is shown in Equation 2.1.

$$DP_{c_i}^{src,dst} = Distance(src, dst) - Distance(c_i, dst) \tag{2.1}$$

Where DP_{c_i} is the distance progress of the candidate c_i, $Distance(src, dst)$ is the geographical distance between the source and the destination, and $Distance(c_i, dst)$ is the geographical distance between the candidate i th and the destination. It simply illustrates geographically closeness to the destination position is considered as a good metric. In other words, it represents how much geographical position of data packet will be enhanced if it is received by the candidate i th. Nodes can inform their neighboring nodes about their geographical position which can be easily provided with GPS equipment by sending hello messages regularly. Moreover, the position of the destination should be known to all other forwarding nodes.

The drawback of DP metric is that it only considers the geographic position of the nodes as the parameters for its metric calculation. It is obvious that this simple metric will work well in the network with lots of forwarding nodes and when there is no limitation and constraint for choosing candidate set. That is, all candidates help together in order to deliver the packet to the destination, regardless of the probability of reaching each others'.

However, most of the time there are some constraints, such as the maximum number of candidates or the number of reachable candidates that forced us to consider other factors besides only geographical position. As an illustration, consider a candidate close to the destination with a very low probability of reaching it from the sender is not a wise candidate selection. To address this issue, another metric called Expected Distance Progress (EDP) has been proposed [16]. It calculates the expected distance progress of sending a packet by using a set of forwarding candidates. Equation 2.2 shows the EDP metric formula where src is the source, dst is the destination, C is the candidate set for reaching the destination, DP_{c_i} is distance progress of candidate i^{th}, $p_{i,j}$ is the link delivery probability between node i and j, and C_{max} is the maximum number of candidates.

$$EDP(src, dst, C^{src,dst}) = \sum_{i=1}^{C_{max}} DP_{c_i}^{sec,dst} \times p_{src,c_i} \prod_{j=1}^{i-1}(1 - p_{src,c_j}) \tag{2.2}$$

In contrast to DP, which merely considers geographical position of the nodes and assumes closeness to the destination as a geed metric, EDP considers both the node geographical positions and link delivery probability to reach them as an enhanced OR metric over candidate set. Note that in OR, the probability that i^{th} candidate forwards the data packet is equal to the probability that the packet is delivered to that node and the other higher priority candidates do not receive the packet successfully. Thus, the probability of sending a data packet by i^{th} forwarder is calculated as illustrated in last part of Equation 2.2.

The drawback of both DP and EDP is that they ignore the cost of the coordination phase, which has significant effects on throughput and duplicated transmission. They simply assume that the packet is forwarded by the higher priority candidate while it is discarded by the other candidates.

The other local OR metric has been proposed in [51] as Expected One-hop Throughput (EOT). It assumes a trade-off between the advantages of distance progress metrics and the cost of medium time delay which is considered for the coordination part. For addressing coordination issues, EOT integrates DP, reliability of candidates, and MAC layer characteristics. Hence, it should be classified as a local OR metric that considers geographical position, link delivery ratio, and coordination phase.

2.1.2 End-To-End Opportunistic Routing Metrics

As mentioned above, end-to-end metrics consider whole network information from the sender node to the destination in order to calculate the metrics. Number of hop up to the destination is one of the simplest and easiest metrics that adapts well in traditional routing. In Hop-count metric [50] [45], the forwarder with less hop count up to the destination is considered as a better forwarder. The defect of this metric is that it does not consider other factors such as link delivery ratio. Consequently, it may choose a forwarder with a good hop count number, but the probability of reaching the forwarder is too low.

Expected number of transmission or ETX [18] is another topological metric that is used in traditional routing as well. It calculates the number of transmissions required to reliably send a packet across a link including re-transmissions. Equation 2.3 represents the expected transmission count of the link where $P_{i,j}$ is the link delivery probability between node i and j.

$$ETX(i,j) = \frac{1}{p_{i,j}} \tag{2.3}$$

ETX is an easy metric to calculate and implement. The ETX of the whole path, from source to the destination, is basically the sum of the ETX of each routed links. However, [20] has shown that ETX has a low performance in multi-rate environments. Therefore, to address this issue, an extension to ETX called Expected Transmission Time (ETT) has been proposed in [20]. ETT considers transmission time of the successfully delivered packet as link cost. Hence, it can calculate the transmission time by multiplying the amount of ETX in to the bandwidth of the link.

Both the ETX and ETT metrics do not adapt well in opportunistic concept because they were initially proposed for traditional routing, and they only consider one path to the destination. Besides, simulation results in [36] show that using ETX may decrease the performance of OR.

For addressing the single path issue of ETX, authors in [33] proposed a new metric called Successful Transmission Rate (STR), which considers multiple links instead of only one. It is more suitable for the opportunistic concept due to the fact that in OR the packet can be received and transmitted by the multiple candidates, and it may traverse along several paths toward the destination. Equation 2.4 displays the recursive formula of STR metric where $P_{i,j}$ is the link delivery probability between node i and j.

$$STR(src, dst, C^{src,dst}) = \sum_{i=1}^{C_{max}} p_{src,c_i} \times STR(c_i, dst, C^{c_i,dst}) \prod_{j=1}^{i-1}(1 - p_{src,c_j}) \tag{2.4}$$

The first idea of the OR metric that adapts well with the definition of OR was proposed in [56] called Expected Any-path Transmission (EAX). Basically, it is an extension to the ETX metric that considers existence of multi paths to the destination.

There are several methods to calculate the amount of EAX. Assume that the candidate set $C_1, ..., C_n$ has been considered as the forwarding candidates to route the packet from the source (S) to the destination (D), and P is the link delivery probability between the nodes as displayed in Figure 2.1.

The probability that at least one of the candidates receives the packet from the source is calculated in Equation 2.5.

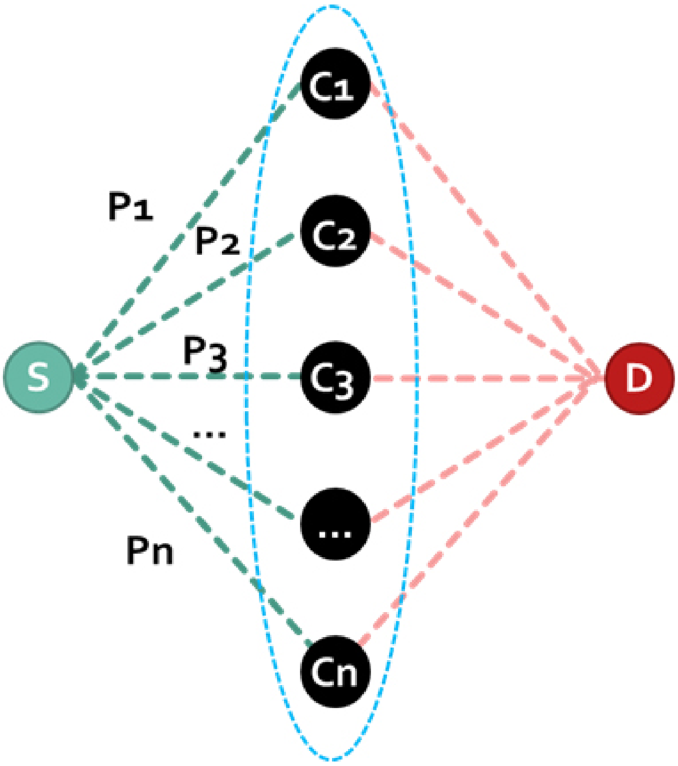


Figure 2.1: Expected Any-path Transmission (EAX) scheme

$$P_1 \vee P_2 \vee P_3 \vee \ldots \vee P_n = 1 - ((1 - P_1) \wedge (1 - P_2) \wedge (1 - P_3) \wedge \ldots \wedge (1 - P_n)) \tag{2.5}$$

According to the ETX metric formula as mentioned in Equation 2.3, the expected number of transmission to receive a packet from the source to at least one of the candidate nodes is calculated in Equation 2.6.

$$First(src, dst, C^{src,dst}) = \frac{1}{1 - \prod_{i=1}^{C_{max}} (1 - p_{src,c_i})} \tag{2.6}$$

After receiving the packet by candidate forwarders, the one with the highest priority has to forward the packet to the destination. Then, the expected number of transmission for the second part (from one of the candidate forwarder to the destination) is calculated as illustrated in Equation 2.7.

$$Second(src, dst, C^{src,dst}) = \frac{\sum_{i=1}^{C_{max}} EAX(c_i, dst, C^{c_i,dst})\, p_{src,c_i} \prod_{j=1}^{i-1}(1 - p_{src,c_j})}{1 - \prod_{i=1}^{C_max}(1 - p_{src,c_i})} \tag{2.7}$$

Consequently, the total EAX from the source to the destination is the sum of both Equations 2.6 and 2.7, as shown in Equation 2.8.

$$EAX(src, dst, C^{src,dst}) = First(src, dst, C^{src,dst}) + Second(src, dst, C^{src,dst}) \tag{2.8}$$

As shown in [15][17], using EAX instead of ETX enhances the expected number of transmissions. However, the main drawback of the EAX metric is high complexity in terms of calculation due to its recursive method. Therefore, it seems that it is not a good choice for the dynamic networks with a high rate of topology changes.

Authors in [24] have introduced a new metric that takes advantage of energy and power consumption. It is useful for wireless mesh networks where nodes' radio are not always on in order to save their energy such as wireless sensor networks. It estimates the number of duty cycles called as Estimated Duty Cycled wake-up (EDC) for end-to-end path.

In [46], authors proposed another new OR metric, which is based on the benefit and cost of transmission. It defines a corresponding utility as the amount of benefit minus the cost of transmission. Equation 2.9 illustrates the recursive formula of Opportunistic Residual Expected Network Utilities (OpRENU) where Opu_i is the Opportunistic Residual Expected Utilities of node i to the destination, $p_{i,j}$ is the link delivery probability, and $COST$ parameter is the transmission cost. Similar to EAX 2.8, OpRENU calculates the corresponding utilities through a set of candidates toward the destination.

$$Opu_{src}^{C^{src,dst}} = \sum_{i=1}^{C_{max}} (Opu_i^{C^{i,dst}} \times p_{src,i} \prod_{j=1}^{i-1}(1 - p_{src,j})) - COST \tag{2.9}$$

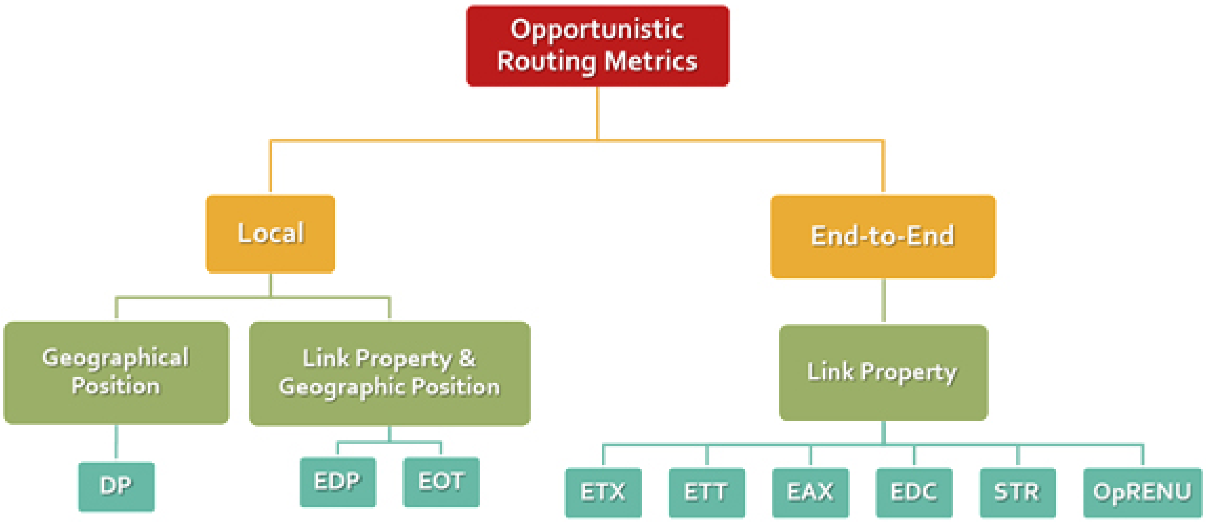


Figure 2.2: Classification of OR metrics

Figure 2.2 classifies all of the mentioned metrics. In section 2.4 we introduce different types of OR protocols that use these different kinds of metrics. As is shown in Figure 2.2, most of the metrics have been proposed for an end-to-end approach with link properties consideration.

2.2 Candidate Selection in Opportunistic Routing

One of the significant parts of OR is known as candidate selection, which is similar to creating a routing table and defining next-hop forwarders in traditional routing.

According to the type of metric that is used in the OR protocol, there are two approaches for candidate selection as well. Some of them only consider the information of the neighboring nodes by using a local metric, while the others have end-to-end and topological views for choosing the candidates. Since using the end-to-end metrics lead us to optimal metrics, the candidate selection algorithms that employ these kinds of metrics, such as EAX [56], considered as optimum selection, and it can be used for comparison purposes. On the other hand, the local-based strategies seem easier to implement and handle due to less signaling and scalability concept.

One of the challenging constraints in candidate selection part is the number of forwarding candidates. Although having a higher number of candidates to forward the packet increases the chance of delivering the packet to the destination, there are some drawbacks to that method. First of all, since all of the candidates have to be mentioned in the header, increasing the number of candidates will increase the size of the packet header as well, which is not desirable. Secondly, a large number of candidates causes complexity in candidate coordination part since a lot of nodes might receive the sent packet. Therefore, it may raise the number of duplicated packets. In addition, if at least one of the candidates receives the packet it can continue the forwarding process in the OR concept. Accordingly, there is a limitation in the number of forwarding candidates in practice.

2.3 Candidate Coordination in Opportunistic Routing

Another important part of OR is candidate coordination. The goal of this phase is to figure out which candidate that successfully received the packet has to forward it among the others. Improper signaling or failure in overhearing between the forwarding candidates causes duplicated transmission, which has an undesirable effects on the performance of the OR protocol. A perfect coordination method should minimize the delay with the lowest overhead, and perform in such a way that all of the candidates can overhear each other to avoid duplicated transmission.

As mentioned in the introduction chapter, coordination methods are divided into two main areas known as control-based and data-based coordination. Generally, in control-based coordination methods, candidates that have successfully received the packet from the sender reply back a control packet explicitly in order according to their priorities to notify the sender and the other candidates. Acknowledgment-based and RTS-CTS are two of the famous control-based approaches. In contrast, data-based methods do not need extra control packet for coordinating. They modify the data packet in order to get coordination information. Timer-based and network coding are two of the useful methods in the data-based approach. We provide an introduction for these coordination mechanisms in the following sections.

2.3.1 Acknowledgment-Based Coordination

The Acknowledgment-based (Ack-based) approach is one of the primary OR coordination methods that has been proposed in [30]. It was used for the first time in the Selection Diversity Forwarding (SDF) [30] protocol. In SDF, four handshaking mechanisms are done in order to perform coordination among the forwarders. Each candidate that received the packet sends back an ACK to the sender to notify it from receiving the packet. Then, the sender decides who is eligible to be the next hop forwarder according to these receiving Ack packets.

The first version of Extremely Opportunistic Routing (ExOR) [4] employs a concept similar to SDF for coordination part. In ExOR the MAC layer is modified in such a way that some time-slots have been reserved for Ack response. Every received Ack represents the successful delivery of the packet. Moreover, Ack carries the ID information of the nodes and the highest priority Ack that has been received by the other candidates. All of the forwarding candidates listen to the medium for all of the Ack slots so that they can decide which one is eligible to forward the packet.

Figure 2.3 illustrates acknowledgment-based coordination using a modified 802.11 MAC. Assume that the sender (S) wants to send a packet to a set of candidates so that candidate A has a higher priority and so on. Then, the time slots for receiving the acknowledgment are considered as A, B ,and C, respectively, as shown in Figure 2.3. Suppose that by any reason candidate B does not overhear the Ack of A, in this case C will notify B about receiving the highest Ack of A by its Ack. Therefore, this Ack packet means that candidate

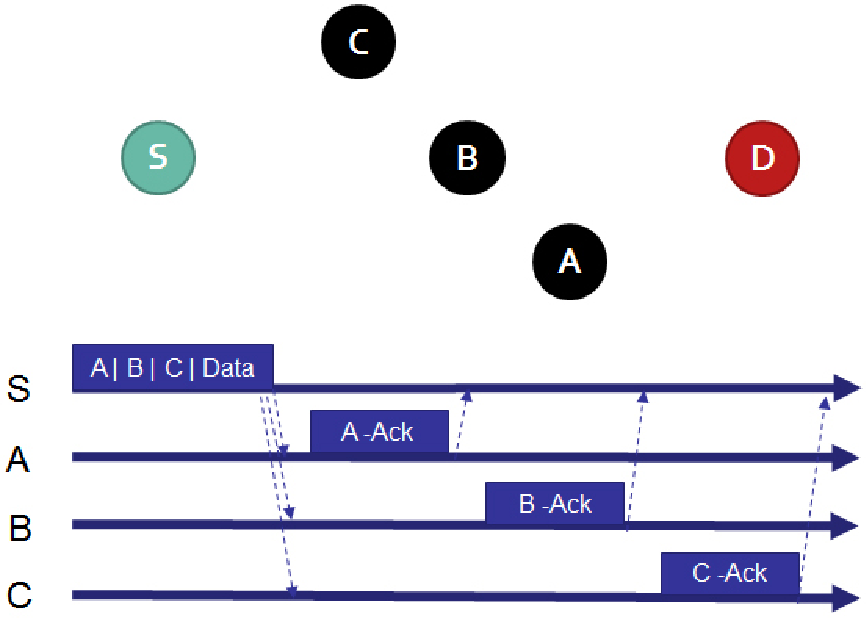


Figure 2.3: Acknowledgment-based coordination mechanism

A received the packet successfully and it is waiting to forward the packet at the end of the Ack time slots.

However, if any lower priority candidate cannot hear other higher priority candidates, it will cause a duplicate transmission. The other defect of this coordination method is the requirement of 802.11 MAC layer modification so that it can channelize for specific Ack time slots.

In [49] it has been shown that if the higher priority candidates do not receive the packet, it will cause a long delay and the channel will be idle for an unnecessarily long period of time. Hence, authors in [49] proposed a new Ack-based coordination mechanism as Fast Slotted Acknowledgment (FSA) to enhance this delay. FSA only needs an Ack from the highest priority candidate that has received the packet successfully. Once the packet is received by the candidates, the highest candidate that grabbed the packet successfully sends back an Ack after a period of time called Short Inter Frame Space (SIFS). Meanwhile, the other candidates listen to the medium to distinguish whether the highest priority candidate forwards the packet or not. In case of not overhearing, the second priority candidate will

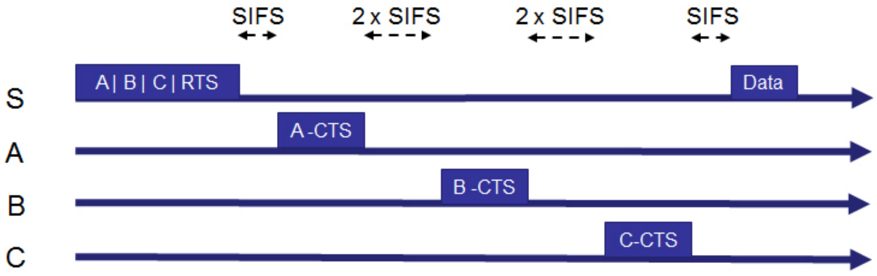


Figure 2.4: RTS-CTS coordination mechanism

be responsible for forwarding and so on. In [49] it has been proved that the performance of the FSA mechanism is mainly dependent on the accuracy of listening to the medium. Otherwise, it will cause failure in forwarding or duplicated transmissions.

2.3.2 RTS-CTS Coordination

In [26] and [57], the authors propose another coordination method called Request-To-Send/Clear-To-Send (RTS-CTS), which employs explicit control packets to figure out the network conditions before transmitting a packet. As displayed in Figure 2.4, first of all, the sender broadcasts a RTS packet that contains the candidate set and the priority of the candidates. Then, the candidates start responding to the RTS by their CTS packets based on their order in the header of the RTS packet. The highest priority candidate waits for a Short Inter Frame Space (SIFS) to send its CTS packet. The time interval for other candidates has to be double that of the SIFS interval to make sure that the sender does not send the data packet to previous higher priority forwarders. After receiving the CTS by the sender, it will wait for a SIFS period of time ,and it will transmit the data packet to the related candidate.

The advantage of the RTS-CTS approach is omitting the duplicated transmission issue. Note that the sender uses the broadcast nature of the wireless medium only at request to send level. After defining the next hop candidate, it uses a unicast mode of transmission to reach the candidate forwarder. Similar to Ack-based method, RTS-CTS needs some modifications in MAC layer level to implement time slot based approach.

2.3.3 Timer-Based Candidate Coordination

One of the most common data-based coordination methods is called timer-based. In this approach candidates are sorted and prioritized in the header of the data packet. In the coordination phase, a time slot for transmission of each of the candidates will be considered and assigned based on their priority. In other words, the i^{th} time slot is assigned to the i^{th} candidate in the header, and the candidate will forward the packet if it does not hear the same packet from other candidates with higher priority. In contrast to control-based coordination approaches, in timer-based we do not need extra control messages. Therefore, it is easy to implement and handle. Most of the time, reaching the higher priority candidates is usually difficult on a count of link delivery probability and geographical distance. Hence, the only drawback to this method can be considered as the high latency due to the waiting time of the low priority candidates. In addition, all of the candidates have to overhear each other to avoid duplicated transmissions, that is, lower priority candidates will re-transmit the packet if they cannot hear the transmission of the higher priority nodes.

2.3.4 Network Coding Candidate Coordination

Integrating of OR with Network Coding (NC) has been proposed in [3], [13], [12], and [47] in order to enhance the duplicated transmission issue of the coordination phase. The basic idea of NC is that a bunch of coded information related to a different packet sent with a single transmission. In other words, packets chunk in small segments of information called batches, and multiple coded batches integrate together to form a transmission payload.

In order to decode the packets at the destination, the sender broadcasts the logical combination of the original packets. Then, candidates cooperate together to forward the coded packets to the destination. Once the destination grabs sufficient information about the packets, it can decode them into original packets. As an illustration for displaying

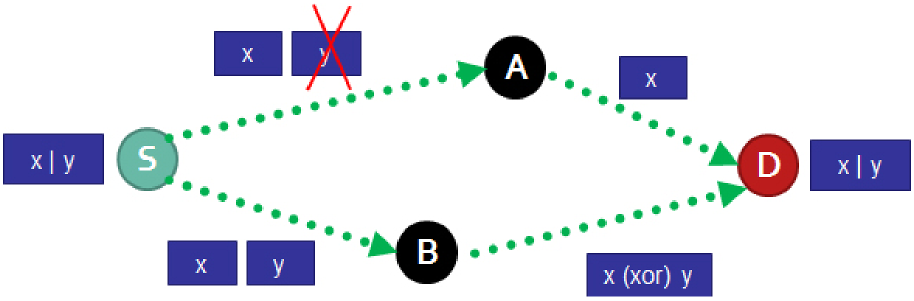


Figure 2.5: Network coding example

the combination of OR and NC, consider the network topology of Figure 2.5. The sender (S) wants to send packet x and y to the destination node (D) via the candidate A and B. Hence, the sender selects A and B as its candidate set and broadcasts both of the packets. Assume that candidate A, which has a higher priority, only receives one of the packets and misses packet y, then it will forward the packet x toward the destination. Meanwhile, candidate B will be noticed that A sent only one packet. Therefore, B will transmit the XOR combination of packet x and y. Finally, D can perform the XOR operation to figure out the native information of the coded packets.

The defect of using NC with OR is that it may lead us to a high number of unnecessary coded packet forwarding. Although there is no duplicated transmission over the network, we may have a bunch of coded packets that do not have any extra information. In fact, as authors show in [12], there is a trade off between the number of coded packet transmissions and the delivery ratio of original data. In addition, complex coding and decoding algorithms may cause considerable calculation overhead as well.

2.4 Opportunistic Routing Protocols

In the following section, some major and well-known opportunistic routing protocols will be introduced. Moreover, their metrics, features, and drawbacks will be discussed.

2.4.1 Extreme Opportunistic Routing (ExOR)

The first idea of the OR scheme has been proposed in Extreme Opportunistic Routing (ExOR) [18], which considers the overhearing of the nodes in wireless multihop networks. ExOR improves routing performance by integrating layer two and three of network layers together. The source node indicates a list of available candidates which are closer to the destination than the source to forward the packet and prioritizes the forwarders based on expected number of transmission (ETX) to the destination. ETX has a direct relation with the distance of the nodes. Then, the shorter distance between forwarder and the destination means higher ETX and higher priority to forward the packet.

All the packets are transmitted in form of broadcasting, and a forwarder will transmit the packet as long as its ID exists in the forwarder list of the packet and it has not heard the same packet from higher priority forwarders.

ExOR has a good performance, especially in static multihop wireless networks, but the main drawback of that is related to the coordination part where low quality link connections or mobility of the nodes lead them to decide incorrectly for transmitting, and that causes duplicated re-transmissions in coordination part.

2.4.2 Simple Opportunistic Adaptive Routing (SOAR)

After proposing the ExOR, the other protocol called Simple Opportunistic Adaptive Routing (SOAR) [43] has been proposed in order to reduce the number of duplicated transmissions. SOAR employs the close candidates along the shortest path to make sure that they can overhear each other. Therefore, all of the selected candidates can hear their transmissions with high probability, and there is no more duplication issues. Similar to ExOR [18], SOAR uses the ETX metric as a cost of the wireless links and runs the shortest path algorithm to find out the shortest path from the source to the destination. For the candidate selection part, it considers the nodes close to the shortest path as forwarding candidate. In addition, SOAR pays attention to the link delivery probabilities among the candidates to make sure that it is not too low for overhearing. The timer-based mechanism is used for the coordination part. Also, the sender sends cumulative Ack to notify the sender about the receiving of the packets and adjust the transfer rate. Then, based on this Ack, the sender can vary its transmission rate. In the simulation part of [43] the authors have compared SOAR, ExOR, and a traditional routing in terms of performance. Their results show that SOAR significantly outperforms traditional routing and ExOR in most of the scenarios.

2.4.3 Opportunistic Any-Path Forwarding (OAPF)

In ExOR, only ETX is taken into consideration for choosing the forwarders, which may cause selection of the nodes with low delivery ratio to them. To address this issue, Opportunistic Any-Path Forwarding (OAPF) [55] proposes a new metric which is called expected any-path count (EAX). It uses a recursive formula to calculate the optimal forwarder set at every hop toward the destination.

The disadvantage of this protocol is the time complexity of its calculation at each of the nodes. Although it leads us near to an optimal path in opportunistic approach, it needs to have the information of the whole path up to the destination. Therefore, it is not suitable for the hop by hop approach such as mobile scenarios.

2.4.4 Least-Cost Opportunistic Routing (LCOR)

The first idea of optimum candidate selection was proposed in [21] and [22]. Optimum term refers to the candidate selection that minimizes the expected number of transmissions via perfect candidate forwarding selection. In [21] [22], the authors have proposed a new distributed algorithms called Least-Cost Opportunistic Routing (LCOR) to minimize the cost of routing from the source to the destination in terms of opportunistic scheme.

Basically, LCOR is the extension of the well-known Bellman-Ford algorithm. In general, the goal of this optimum protocol is to find an optimal candidate set to minimize the total number of transmissions. To address this goal, LCOR employs the EAX metric as an end-to-end metric [21]. The algorithm performs recursively at each hop to figure out all of the possible cases of the candidate set. Hence, it is going to be time consuming at the network with lots of nodes. High computational costs of dense networks can be considered as the other drawback of this protocol.

2.4.5 Minimum Transmission Selection (MTS)

The authors in [32] have proposed another candidate selection algorithm similar to LCOR protocol [21] [22] in terms of optimum approach. MTS uses the EAX metric in recursive manner to find out the eligible candidate forwarders and form the candidate set. The key idea of MTS is as follows: if the expected number of transmissions from a forwarding node (e.g. node A) to the destination is less than its neighbor (e.g. node B). Then, using the node A and its candidate set for the CS of the node B is a wise choice owing to the fact it reduces the expected number of transmissions [15]. In other words, MTS tries to find a node with the least EAX value (MinNode) at each iteration. After that, the MinNode and its candidate set will be added to the neighbors of MinNode. This optimal algorithm uses a common timer-based coordination method. It is shown in [17] that MTS outperforms the other optimum candidate selection (LCOR) in terms of the execution time, whereas the expected number of transmissions in both algorithms are approximately the same. However, MST only considers the candidate selection part, and it suffers the complexity of the EAX metric similar to LCOR.

2.4.6 MAC-independent Opportunistic Routing and Encoding protocol (MORE)

MAC-Independent Opportunistic Routing and Encoding Protocol (MORE) [13] employs the advantages of network coding in order to solve the duplication issue of ExOR protocol. Source node generates coded packet by random linear combination of the K native packets in the one batch. It starts to send a coded packet until it gets an acknowledgement for all of the packets in the batch. Then, it proceeds to the next batch. Packets are always sent in coded format, which contain a list of capable forwarders and code vector to indicate the combination process. Once the coded packet is received by the forwarder node, it will check whether it is a new or old one. In case of a new packet, it will transmit the packet after decreasing the counter, which is calculated based on the expected number of transmissions. If the counter reaches zero, the packet will be dropped. After receiving the packet by the destination, it decodes the whole batch and sends back an ACK to notify the sender to shift to the next batch. Simulation results illustrate that using concept of network coding significantly reduces the number of duplicated packets when compared to ExOR protocol.

2.4.7 Coding-aware Opportunistic Routing mechanism and Encoding (CORE)

Similar to MORE protocol, Coding-Aware Opportunistic Routing Mechanism (CORE) [47] employs a combination of opportunistic routing and localized inter-flow network coding concept. In CORE, the neighboring node with the highest coding gain is chosen to forward the packet. It tries to maximize the payload that can be sent over one batch. A packet holder forwards a packet to the next hop by changing the code among its forwarder set in such a recursive way that improves coding throughput. Geographical distance progress is used to calculate the metric. Timer-based coordination method with help of networking coding are responsible for coordination phase. This mechanism leads us to significant improvement in packet delivery with little extra overhead.

2.4.8 Practical Network Coding for Wireless Mesh Environments (COPE)

Practical Network Coding for Wireless Environments (COPE) is another network coding approach for OR that proposed in [28]. In fact, it is an opportunistic method that is used for network coding. Each node listens to the medium and gets the local information about its neighbors to detect the coded packets and decode it as needed. The advantage of this protocol is that it can support all kinds of traffic such as on-demand and bursty. Also, it is capable of carrying multiple unicast flows. The authors in [28] show that COPE has considerably higher throughput compared to usual 802.11 MAC method when there are multiple flows over the wireless network.

2.4.9 NC(XC) Opportunistic Routing (XCOR)

The other type of OR protocol that employs the inter flow network coding concept is called XCOR [29]. It tries to find the forwarding candidates by using the ETX metric along the shortest path from source to the destination. At each hop, the node with the best ETX to the destination forwards the packet immediately. Meanwhile, the other candidates use a timer-based method based on their priority for waiting. Then, every forwarding node periodically sends a reception report to alert its neighbors about the packets it has already received. According to these reports and flows that cross from each node, XCOR calculates the utility of each possible combination of the packets of the different flows to distinguish one with the best effective utility. Authors in [29] proposed a heuristic algorithm for computing the utility of the flows in order to reduce the the computation cost.

2.4.10 OPportunistic Routing in dynamic Ad Hoc networks (OPRAH)

In OPRAH [45], relay nodes create a set of downstream paths toward the destination and also upstream paths toward the source by sending on-demand route requests and route replies. OPRAH is more suitable for dynamic wireless networks among all of the mentioned OR protocols up to now. However, it selects only one forwarding node as the transmitting node, and it does not take advantage of transmissions that reach nodes other than the previously selected forwarder. Simulation results show that end-to-end forwarding reliability of packets have been increased by using this protocol. Nevertheless, it seems that receiving duplicated packets at the destination is unavoidable. Owing to the fact that the route set may contain spatially or partially disjointed paths from the intermediate nodes toward the destination.

2.4.11 Contention-Based Forwarding (CBF)

Contention-Based Forwarding (CBF) [23] is one of the simplest geographically proposed Opportunistic Routing methods. CBF is a receiver-based protocol that does not need hello or beacon messages to distinguish the neighboring nodes and candidate sets. In CBF, the sender knows the position of itself and the destination via GPS and location server. Once the sender wants to transmit a packet to the destination, it inserts the information of its location and destination's location into the packet and broadcasts the packet. After receiving the packet by neighboring nodes, each neighbor schedules a timer to forward the packet according to the Equation 2.10 and 2.11 where DP_{c_i} is the distance progress of ith candidate, τ is the constant amount, and R is the radio transmission range. Thus, the closer to the destination, the shorter the expiration timer. Meanwhile, if the node overhears the same packet from other candidates, it will cancel the timer and drop the packet. Accordingly, CBF can be classified as a timer-based approach. Note that only candidates closer than the source to the destination participate in CBF algorithm. This protocol has been used for the comparison part of this survey to show the performance of one receiver-based routing protocol in comparison to the suggested one.

$$DP_{c_i} = Distance(src, dst) - Distance(c_i, dst) \tag{2.10}$$

$$TimeToSend = \tau \times (1 - \frac{DP_{c_i}}{R}) \qquad 0 < DP_{c_i} < R \tag{2.11}$$

There are some defeats for this protocol. First, we will face duplicated transmission as long as the candidates cannot overhear each other. Second, There is no limitation for the maximum number of forwarding candidates, which causes complexity in the coordination phase. Third, we will show in the simulation result chapter that throughput of CBF is considerably dependent on the number of the forwarding nodes over the network. In other words, CBF does not perform well on a sparse network.

2.4.12 Geographic Opportunistic Routing (GOR)

Geographic Opportunistic Routing (GOR) [53] is a geographic Opportunistic Routing protocol, as its name shows. In [51], the authors illustrated that giving the priority to the closer neighbors to the destination does not always lead us to the optimal solution and best throughput. They have proposed the EOT metric to evaluate the local behavior of the nodes in GOR. As mentioned in OR metric section, the EOT is responsible for considering the coordination overhead among the candidates. GOR inserts a new node into the candidate set in a greedy manner without changing the priorities of already selected forwarding candidates. It tries to maximize the EOT at each hop. Ack-based method is employed for coordination part of GOR. The low number of comparing protocols in experiment part can be considered as a drawback of this proposed protocol.

2.4.13 Geographic Random Forwarding (GeRaF)

Geographic Random Forwarding (GeRaF) [57] is a geographical opportunistic forwarding protocol. It uses location information to prioritize the eligible forwarders in the candidate forwarding set. Each packet contains the position of the source and the destination. Only those reachable nodes closer to the destination than the source node are eligible to forward the packet. Then, these neighboring nodes are prioritized based on their distances to the destination. It employs RTS-CTS mechanism in hop-by-hop manner to forward the packet. Although GeRaF is simple to implement and use, it is more suitable for dense networks. Time to access location server for getting the position information of the nodes is considered as another drawback of this protocol.

2.4.14 Position-based Opportunistic Routing (POR)

Position-based Opportunistic Routing (POR) has proposed in [48]. In POR, Nodes get their location by using their GPS and broadcast it to their neighbors via hello messages periodically. Once the sender needs to send a data packet to the destination, it puts its candidate set in the packet header and broadcasts it. Note that, the candidate set can be calculated in active or passive mode by using the DP metric, as mentioned in Equation 2.1. It means that the closer the node gets to the destination, the higher priority in the candidate set selection and prioritization phase. After receiving the packet by the other nodes, if their ID does not exist in the header, they simply drop the packet. Otherwise, a timer-based candidate coordination method is employed to determine which one has to forward the packet. These hello messages for updating neighboring nodes information can be considered as the only overhead of this protocol.

In addition, some modifications need to be done at MAC layer to use the broadcast concept of OR in better way. The authors in [48] have shown that POR has better performance compared to Ad Hoc On-Demand Distance Vector (AODV) [40] as a traditional routing and Greedy Perimeter Stateless Routing (GPSR) [27] as a well-known wireless routing. The main drawback of this simple protocol is that it only considers the closeness to the destination as a good metric and does not observe other factors such as link delivery probability.

2.4.15 Distance Progress Opportunistic Routing (DPOR)

In contrast to POR protocol [48], which only considers the closeness of the nodes to the destination, in Distance Progress Opportunistic Routing (DPOR) [16] both the geographical position of the nodes and link delivery probability to them are taken into consideration. In other words, the sender tries to find candidates that are closer to the destination than itself, and their link delivery ratio is not too low. DPOR uses the EDP metric to calculate the set of candidates that maximize the EDP amount. Each sender does the same metric calculation until there is no more desirable candidates or the number of candidates reaches the maximum number. The authors in [16] evaluate the performance of DPOR with three other algorithms (ExOR, OAPF, and MTS), and they show that the performance of DPOR is close to the optimal algorithm and is much better than the other ones in terms of the expected number of transmissions. However, DPOR only focuses on the candidate selection part and it assumes perfect coordination in the candidate coordination phase. Hence, this can be a defect to evaluate the performance of the whole protocol.

2.4.16 Location-Aided Opportunistic Routing protocol (LAOR)

As the name of Location-Aided Opportunistic Routing protocol (LAOR) [54] shows, this is a geographical protocol that uses the location information of nodes in order to select and prioritize the candidate set. Furthermore, LAOR minimizes resource consumption and duplicated transmissions by reasonably choosing forwarding nodes to prevent routes from diverging. In addition, it employs a re-transmission mechanism in order to localize recovery issues if the sender does not get the Ack of the sent packet in the specified time slot. The base idea of this protocol is similar to the other routing protocols such as POR [48] and GeRaF [57]. The only difference is related to the coordination part that LOAR uses acknowledgment-based method. The main drawback of this proposed protocol is the poor results of the experiments in terms of comparing it with other good routing protocols. Moreover, a high rate of control messages is another concern of this routing protocol.

2.4.17 Geographical Opportunistic Source Routing (GOSR)

Geographical Opportunistic Source Routing (GOSR) [35] is initially proposed for VANET environment. It is a combination of the opportunistic concept and Geographical Source Routing protocol (GSR) [34]. First, GSR algorithm is done in an on-demand way. It generates a graph from the topology, and the length of the road segments are considered as the cost of the graph's links. Then, the Dijkstra algorithm is run to find the shortest path to the destination. After that protocol is entered into the opportunistic phase, and it tries to find a scope near the best path that contains the eligible forwarders. In the coordination phase, after receiving the packet by the forwarders, they transmit the packet on specific time according to their distance to the destination if they do not receive the same packet from other forwarders (timer-based coordination approach). Simulation results show that employing GOSR can reduce the number of hop count and end-to-end latency.

2.4.18 TOpology-assisted Geo-Opportunistic routing (TO-GO)

TOpology-assisted Geo-Opportunistic routing (TO-GO) [31] is a geographic OR protocol that is introduced for Vehicular ad hoc networks. TO-GO tries to find an effective candidate set between the sender and the farthest node in the same road, and it also takes the road information topology in order to calculate effective candidates. First of all, the sender predicts whether the target node is at the junction point or not. It is assumed that the nodes at junction can transmit the packet in any direction. Then, the sender chooses a set of candidates that can hear the target and other candidates. After receiving the packet, each candidate waits for a period of time based on its distance to the target node and forwards the packet if it does not hear the packet from other forwarders. The results of this protocol illustrate that using prediction in an environment with loss and fading model can be lead to a good throughput. However, TO-GO only considered grid mobility model in its experiments and did not examine other mobility models.

Table 2.1: Summary of opportunistic routing protocols

Protocol	**Metric**	**Candidate Selection**	**Candidate Coordination**
ExOR v.1	ETX	End-to-end	Ack-based
ExOR v.2	ETX	End-to-end	Timer-based
SOAR	ETX	End-to-end	Timer-based
OAPF	EAX	End-to-end	Timer-based
LCOR	EAX	End-to-end	Timer-based
MTS	EAX	End-to-end	Timer-based
MORE	ETX	End-to-end	Network coding
CORE	DP	Hop-by-hop	Timer and network coding
COPE	ETX	Hop-by-hop	Network coding
XCOR	ETX	End-to-end	Timer and network coding
OPRAH	Hop count	End-to-end	N/A
CBF	DP	Hop-by-hop	Timer-based
GOR	EOT	Hop-by-hop	Ack-based
GeRaF	DP	Hop-by-hop	RTS-CTS
POR	DP	Hop-by-hop	Timer-based
DPOR	EDP	Hop-by-hop	N/A
LAOR	DP	Hop-by-hop	Ack-based
GOSR	ETX	Hop-by-hop	Timer-based
TO-GO	DP	Hop-by-hop	Timer-based

Chapter 3

Enhanced Mobility-based Opportunistic Routing (EMOR)

3.1 Overview

In this section, we explain our Enhanced Mobility-based Opportunistic Routing method (EMOR). As mentioned in the introduction part, employing the idea of OR can increase the performance of the routing in terms of the delivery ratio and reliability in wireless multihop environments. Most of the work on OR has been done for static mesh networks so far where nodes do not have mobility. When nodes are mobile, candidate set changes frequently and updating the whole topology information is very costly. Therefore, a fast and efficient candidate selection and coordination algorithm is needed in the mobile scenarios in order to adapt rapidly with the frequent topology changes. EMOR is a routing approach that combines the concept of the opportunistic routing and mobility properties of the nodes to address this issue and adapt well for mobile scenarios. In the simulation results chapter, we will illustrate that the packet delivery ratio and the expected number of the transmissions will be enhanced by using the EMOR method for the mobile ad-hoc scenarios when compared with other routing protocols.

As mentioned before, geographical hop-by-hop protocols are used mostly in mobile scenarios because handling the states of the end-to-end paths are so difficult and costly, while the topology is always changing. EMOR is a hop-by-hop OR approach which consists of two phases: candidate selection and candidate coordination. To obtain the local information in EMOR, each node broadcasts its ID, current position, velocity, and the number of neighbors, which we refer to as density of node, by broadcasting hello messages every $T_{interval}$. Once the hello messages are received by a node, it has the local information of its neighbors and can select some of its neighboring nodes as its candidate set.

In the candidate selection phase, EMOR uses the predicted position of nodes and the link delivery probability between the sender and neighboring nodes to select the candidate set. It also considers the number of neighbors of its neighbor which we refer to as density as another parameter to select the candidates. Figure 3.1, 3.2, and 3.3 display the consideration factors for the selection phase. Once the candidates are selected, the sender puts them in the header of the data packet and broadcasts it. In the second phase of EMOR (candidate coordination), each candidate that has received the packet coordinate with each other to decide which one has to forward the packet. In the following section we explain both the candidate selection and coordination phases in more details.

To maintain the neighboring list information up to date, each node keeps the list of its neighbor during a specific period of time called dead time (represented by T_{dead}). If it does not receive any more hello messages from the neighbor during this period, it will consider it as dead one and will remove it from its neighbor list.

3.2 Candidate Selection

In the candidate selection phase, EMOR considers different parameters to select some of the neighboring nodes as the candidates set. In OR, the nodes that are closer to the destination than the current node are considered as the potential candidates. Using the hello message information, including the current position, velocity, and direction of nodes,

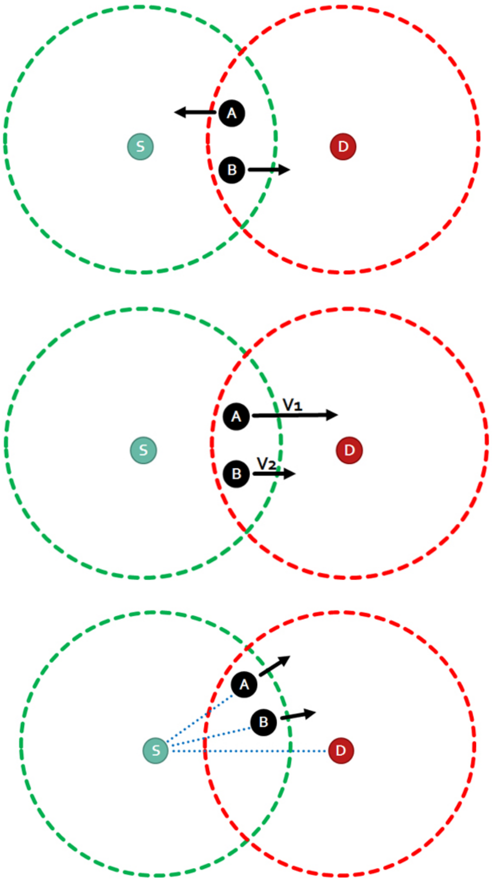


Figure 3.1: Consideration of direction, velocity, and angel of movement

each node can estimate the future position of its neighbors. Based on Equations 3.1 and 3.2 the position of neighboring nodes will be predicted for a short period of time based on the hello interval so that the curved pattern of the nodes can be covered. Note that in Equations 3.1 and 3.2, θ is the angel of the motion vector of the adjacent node with horizon line, $V_{current}$ is the current velocity of the neighboring node, and X and Y are the predicted and current coordinate position.

$$X_{next} = V_{current} \times \cos\theta + X_{current} \tag{3.1}$$

$$Y_{next} = V_{current} \times \sin\theta + Y_{current} \tag{3.2}$$

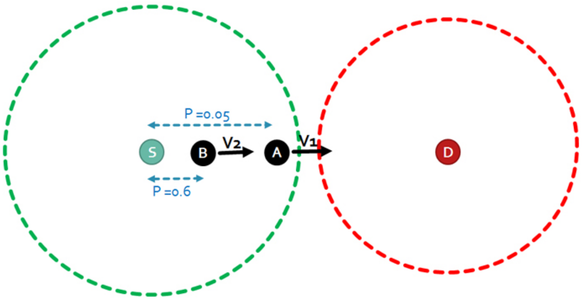


Figure 3.2: Consideration of link delivery probability

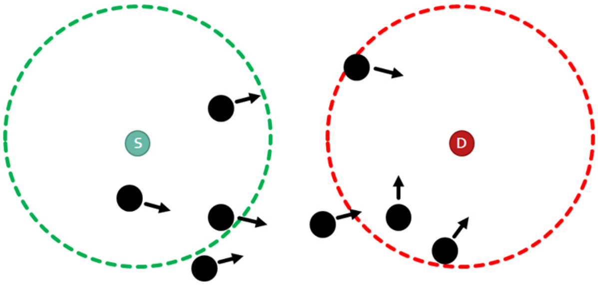


Figure 3.3: Consideration of density of neighboring nodes

Based on the predicted position of neighbors and also the known position of the destination, estimated distance of each neighbor to the destination will be calculated by using Pythagorean equation, and it is represented by PD_i where i is the ID of node.

Although selecting the nodes which are close to the destination that the sender could increase the packet progress toward the destination, reaching the nodes far from the sender is more difficult. Therefore, the other factor that has to be considered in the candidate selection process is link delivery probability between the current node and its neighbors. The link delivery probability between a node and its neighbors can be calculated using the number of received hello messages in a predefined period of time. This probability can be calculated by dividing the number of received hello messages from the neighbor in a specific period of time by the number of hello messages that were supposed to be received. We illustrate it by $P_{(i,j)}$ where i and j are the ID of two neighboring nodes.

Despite the fact of having more candidates to forward the packet increases the chance of delivering the packet to the destination, there are some drawbacks for that. First of all, since all of the candidates have to be mentioned in the header, increasing the number of candidates will increase the size of packet header. Secondly, a large number of candidates causes the candidate coordination to be more complex since a lot of nodes might receive the sent packet. Therefore, it may raise the number of duplicated packets. In addition, in OR if at least one of the candidate receives the packet it can continue the forwarding the process. Therefore, there is a probability that when a source broadcasts a packet at least one of its candidates received the packet which can be calculated as shown in Equation 3.3:

$$P_1 \vee P_2 \vee P_3 \vee \ldots \vee P_n = 1 - ((1 - P_1) \wedge (1 - P_2) \wedge (1 - P_3) \wedge \ldots \wedge (1 - P_n)) \tag{3.3}$$

Where P_i is the link delivery probability between the sender and node i. As we can see in Equation 3.3 increasing the number of candidates does not improve the probability of reaching the candidates set that much. Therefore, as shown in [14] having a small number of candidates can provide a large enough probability to reach at least one of the candidates and we have considered constraint on the maximum number of candidates in each node which is set to C_{max}. In simulation part, five candidates are considered according to [14] as a good enough number of relaying candidates.

Another parameter that has to be taken into consideration for calculating metric in the candidate selection part is geographical progress of a transmitted packet, which is defined as difference between distance of the source to the destination and predicted distance of the neighboring candidate to the destination (PD_i). We show it as Geographical Progress (GP). According to this definition, we define potential candidates as the nodes which their predicted position is closer to the destination that the current node. We have normalized the GP value by dividing it to the transmission range in order to use it in metric calculation. Equation 3.4 illustrates our Expectation Predicted Progress metric (EPP) over candidate set $C_1, C_2, \ldots, C_{max}$ which is sorted based on their priorities and used in candidate selection algorithm. Note that the candidate nodes are prioritized based on this distance progress in candidate selection algorithm.

$$EPP = \sum_{i=1}^{C_{max}} Normalized(GP_{c_i}) \times P_{src,c_i} \times \prod_{j=1}^{i-1}(1 - P_{src,c_j}) \tag{3.4}$$

Note that in OR, the probability that i^{th} candidate forwards the data packet is equal to the probability that the packet is delivered to that node and the other higher priority candidates do not receive the packet successfully. Thus, the probability of sending a data packet by i^{th} forwarder is calculated in Equation 3.5.

$$P_{forwarding} = P_{src,c_i} \times (1 - P_{src,c_1}) \times (1 - P_{src,c_2}) \times (1 - P_{src,c_3}) \times \ldots \times (1 - P_{src,c_{i-1}}) \tag{3.5}$$

Algorithm 1 represents the pseudo code of candidate selection algorithm that EMOR uses. First of all, the neighbor list will be checked and updated, which means that all of the farther, dead, out of coordination range, and under density threshold neighbors will be removed from the neighboring list. Note that the density of each neighbor is used in order to make sure that the selected candidate has enough neighbors so that it will not face the local maximum issue. That is, the packet will not reach to a candidate that it cannot forward it anymore toward the destination because of lack of any neighboring node. The Expectation Predicted Progress (EPP) will then be calculated so that it maximizes the progress of the sent packet. As a result, candidates set for forwarding the packet toward the destination will be provided. (For more information about the source code see Appendix A)

Algorithm 1 Candidate selection pseudo code

```
forall the $node_i \in neighborList[i]$ do
    if ($dead\ neighbor,\ out\ of\ coordination\ zone,\ GP_i < 0,\ Dens_i < Dens_{threshold}$) then
        remove $node_i$ from $neighborList[i]$
    end
end
for $i = 1 \rightarrow C_{max}$ do
    sort the $candidateSet[i]$ based on $PD_i$
    forall the $node_j \in neighborList[j]$ do
        calculate $EPP_j$ over $candidateSet[i]$
    end
    $candidateSet[i] \leftarrow node_j$ that maximize the $EPP$
    ignore $node_j$ for next iteration
end
```

3.3 Candidate Coordination

When a node selects its candidates, it puts them in the header of the data packet and broadcasts it. Each node that receives the packet will check if its ID exists in the header or not. In case it does not exist, the node will simply drop the packet. Otherwise, the candidate will wait for a period of time according to its priority, which is mentioned in the candidates set in order to transmit the packet. During this period of time the candidate will listen to the medium to see whether any other higher priority candidate is forwarding the packet or not. A candidate will forward the packet if it does not hear the transmissions of the same packet from other candidates during its waiting time. Note that the highest priority candidate will not wait and, as long as receives the packet, it will immediately forward it. This process continues until the data packet will be reached to the destination.

One of the challenging issues in OR is the amount of duplicated packets generated due to failure in the coordination phase. In other words, more than one candidate will forward the same packet if they cannot overhear each other. To address this issue, the feature of Reuleaux triangle is used to make sure that sender and candidates can overhear each other. The worst positions of nodes are displayed in Figure 3.4 so that they can coordinate together around the triangle area. R represents the radius of transmission range.

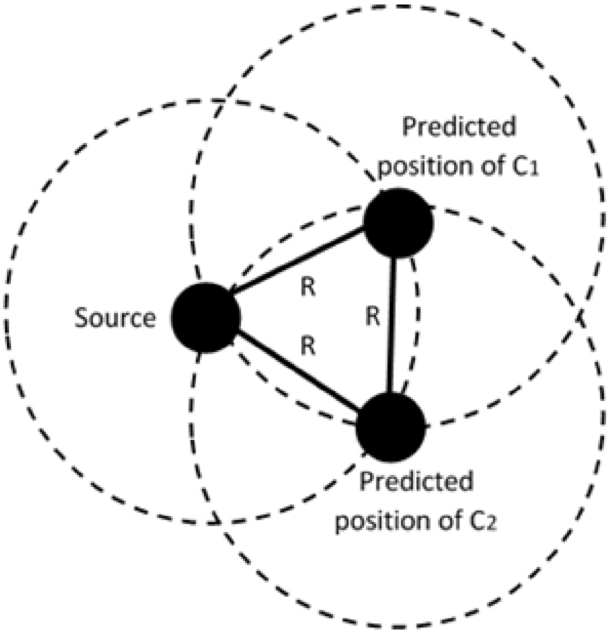


Figure 3.4: Coordination zone

In EMOR, when a node sends a packet it waits for a period of time to see if any of its candidates forwards the packet or not. In other words, it acknowledges receiving the packet in EMOR, which is done hop-by-hop by overhearing the broadcast signal of one of the candidates. In case the sender does not hear any transmission of the same packet from one of its candidates, it will re-transmit it for at most N_{retry} times.

Chapter 4

Simulation Results

In this chapter, we will first introduce the conditions of the simulation environment, such as primary assumptions, propagation models, and mobility models. Then, we mention the metrics that are measured during these simulations in order to compare the performance of the protocols. Finally, we present the graphs and charts related to the simulations and discuss them.

4.1 Simulation Environment

In our NS-2 simulation, we assume that the position of the destination is known to all of the nodes via location server. Moreover, some random traffic is generated by the nodes to emulate a more realistic scenario in case of congestion. Also, five candidates are chosen in the simulations in practical manner and based on [14].

An environment with a high rate of packet loss is used for simulating in order to illustrate the performance of OR concept rather than other routing protocols of these environments. The two-ray model predicts the received power as a deterministic function of distance, and it represents the communication range as a circle. But in reality, the received power at a certain distance is a random variable due to multipath propagation

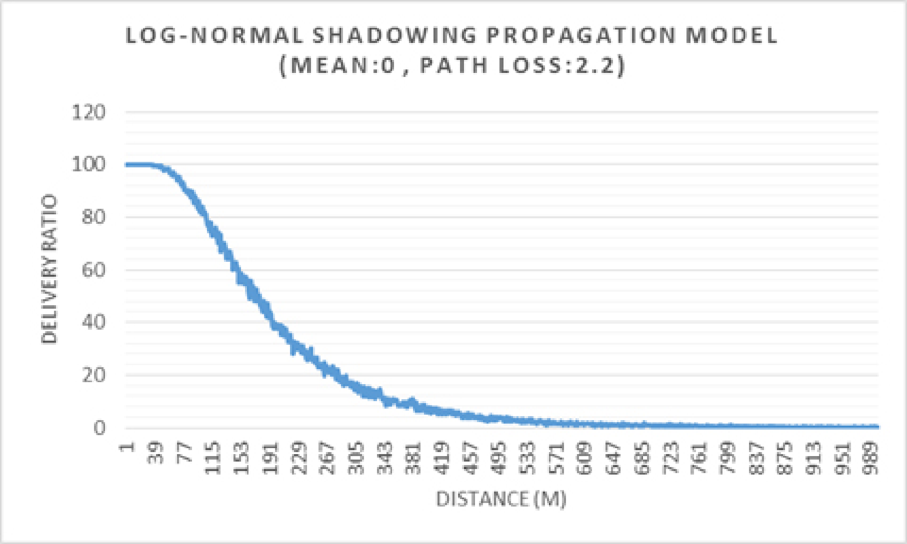


Figure 4.1: Shadowing propagation model

effects, also known as fading effects. Shadowing propagation model is employed to address these conditions.

Path loss model and variation of the received power at a certain distance are two main parts of the shadowing model. According to the Equation 4.1 and 4.2, the mean received power at distance d (represented by $\overline{P_r(d)}$) can be predicted where P_r is the received signal power at the destination, P_t is the transmitted signal power. G_t and G_r are the antenna gains of the transmitter and the receiver, respectively. L is the system loss, and λ is the wavelength. Finally, $\overline{P_r(d)}$ is calculated relative to $P_r(d_0)$ as a reference in Equation 4.2 where β is the path loss [41].

$$P_r(d) = \frac{P_t G_t G_r \lambda^2}{(4\pi)^2 d^2 L} \tag{4.1}$$

$$\frac{P_r(d_0)}{\overline{P_r(d)}} = \left(\frac{d}{d_0}\right)^{\beta} \tag{4.2}$$

The behavior of the delivery ratio of the used propagation model based on the geographical distance between nodes is represented in Figure 4.1.

Random Direction, Random Waypoint, and Manhattan Grid mobility models are used to examine the behavior of the routing protocols in different mobile scenarios.

In the Random Waypoint mobility model each node chooses a target position randomly and starts moving toward the target with randomly selected speed between defined minimum and maximum speed. Then, it will choose another random point after reaching the specified target. This model is used widely as a reference for mobile scenarios.

On the other hand, the Random Direction mobility model [44] is basically similar to the Random Waypoint model, but it does not suffer from the density waves in the center of the simulation area that the Random Waypoint model does. In this model each node keeps going up to the edge of the simulation area border with random direction. Once the simulation boundary is reached, the node pauses for a specified time, and it chooses another angular direction between 0 and 180 degrees for its path.

Finally, in the Manhattan Grid mobility model [1], the simulation area divided into arbitrary grid segments. Nodes have to go straight forward; only at the junction points can they choose whether they turn right, left or keep going straight. The grid of 5x5 with a probability of 0.5 for going straight and 0.25 for turning right and left has been considered to evaluate the performance of mentioned protocols in an environment such as VANET. We will show in following sections the behavior of mentioned protocols in these different mobility models.

In the comparison part, we compare AODV [40] [39] as a traditional routing protocol, POR [48] and DPOR [16] as geographical and hop-by-hop OR protocols, CBF [23] as a receiver-based OR protocol that does not use hello packet to discover the network, and finally GPSR [27] which is a well-known geographical protocol for wireless mesh networks.

End-to-end packet delivery ratio, latency, expected number of data transmissions, and expected number of effective data transmissions are measured for the evaluation part. Delivery ratio shows the general throughput of a protocol, which is the number of sent packets at the source node divided by the number of successfully received packets at the destination. Latency or End-to-End delay displays the time of transmission of the packet from the source to the destination. The expected number of data transmissions is defined as the number of all data transmissions over the network divided by successfully received data packets at the destination, and it logically shows how many data transmissions are required over the entire network to deliver a data packet from the source to the destination. On the other hand, the expected number of effective data transmissions only illustrates the number of transmissions of the received data packets over the network.

4.2 Simulation-1

The purpose of the first simulation is to show the behavior of each of the mentioned routing protocols in different speeds. The parameters of simulation-1 are mentioned in Table 4.1. Number of nodes is considered so that they can cover the playground of the simulation. Figures 4.2, 4.3, 4.4, and 4.5 display the performance evaluation and comparison of the mentioned protocols in terms of packet delivery ratio, latency, expected number of data packet transmissions, and candidate forwarding status.

Table 4.1: Simulation-1 parameters

Playground size	500m x 500m
Propagation model	Log-normal Shadowing (mean=0, stdDev=6, path loss=2.2)
MAC layer mode	802.11 g (2.412e+9Hz)
Mobility model	Random Direction
Speed	5, 10, 15, 20, 25, 30 (m/s)
Hello interval	Every 1 second
Data traffic	Constant Bit Rate
Number of nodes	40

As expected, AODV as a traditional routing has a low and unstable performance (large

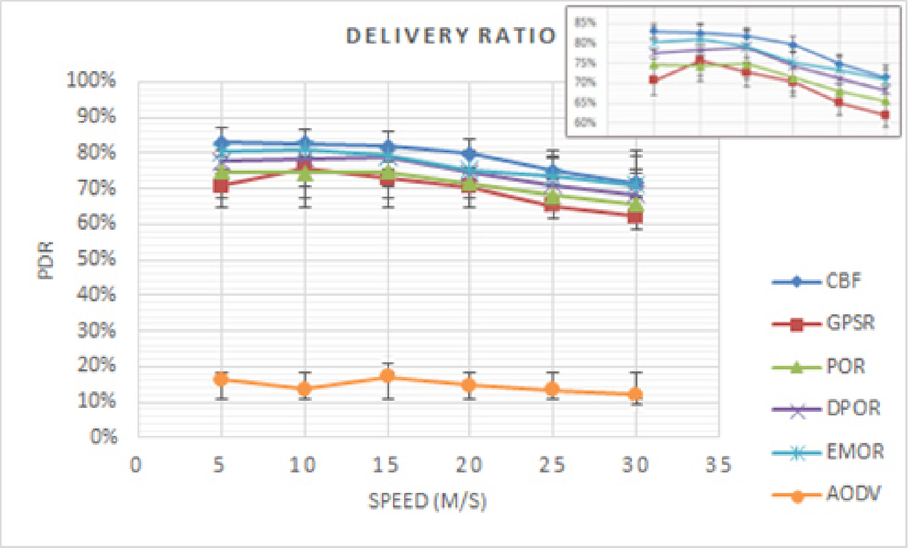


Figure 4.2: Delivery ratio

confidential interval) in mobile scenarios with a high percentage of packet loss. Hence, another small graph is considered at Figure 4.2 to illustrate the delivery ratio of the other protocol in a better way.

Generally, as indicated in Figure 4.2, packet delivery ratio decreases as the speed raises. Although CBF has the best delivery ratio, it suffers from high latency and high number of data transmission over the network (see Figures 4.3 and 4.4). On the other hand, all of the opportunistic routing protocols have better packet delivery ratio than GPSR in simulation 1 environment.

As illustrated in Figure 4.2 and 4.3, Enhanced Mobility-based Opportunistic Routing protocol (EMOR) has a higher delivery ratio when compared to the other OR protocols and even GPSR with having reasonable latency. Note that logarithm scale is used to display the different times of these protocols in Figure 4.3. In addition, the low rate of expected transmission of EMOR in Figure 4.4 indicates the performance of proposed coordination algorithm, which reduces the number of duplicated data packets.

Although GPSR has lower a delivery ratio than OR protocols, according to Figure 4.3 it has less delays when compared to the other OR protocols that use timer-based coordination

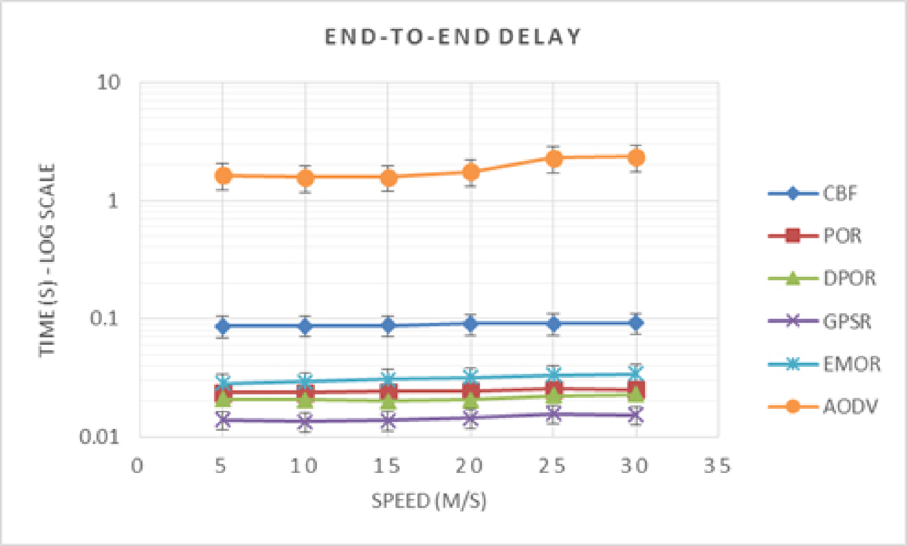


Figure 4.3: End-to-end delay

approach. In fact, the only drawback of the timer-based method can be considered as the high latency due to waiting time of the low priority candidates for forwarding. Figure 4.5 displays the role of each candidate in forwarding the data packets among all of the OR protocols. As indicated, almost half of the transmissions are done by middle candidates. As such, a proposal for future works these portions can be considered in such a way that transmission priority will be given to middle candidates in order to enhance the total end-to-end latency.

4.3 Simulation-2

The aim of second simulation is to show the effect of the number of nodes and the role of different mobility models over performance of the mentioned protocols. The parameters of simulation 2 are stated in Table 4.2.

Figures 4.6, 4.7, 4.8, and 4.9 display the behavior of the considered protocols in different mobility models. As illustrated in charts of Figure 4.6, in terms of delivery ratio, CBF has a linear increasing behavior which is totally dependent on the number of nodes. On the other hand, AODV has a considerable exponential decreasing trend. Besides, OR protocols and

Figure 4.4: Expected number of data transmission

GPSR have almost stable and good trends regarding the number of the relay nodes. Note that among all of these protocols, EMOR has a higher packet delivery ratio and steadier performance, even in sparse networks. Furthermore, as displayed in Figure 4.6, there are little differences between the delivery ratio of charts in terms of various mobility models. However, packet delivery ratio of Manhattan Grid mobility model is generally lower than the two others.

In Figure 4.7, the total latency of these routing protocols are shown. All routing protocols except AODV have the same behavior in different mobility models. The proposed protocol has higher latency when compared to the GPSR, POR, and DPOR due to its complexity in the candidate selection phase. Nonetheless, the total end-to-end delay of all mentioned routing protocols can be considered stable in the various number of forwarding nodes.

Figure 4.8 displays the expected number of total data transmissions in different mobility models. As mentioned before, this value logically shows how many data transmissions are required over the network so that a data packet is received successfully from the source to the destination. The main point of these charts is the performance of CBF protocol as a receiver-based algorithm. Although CBF does not employ the hello messages to discover

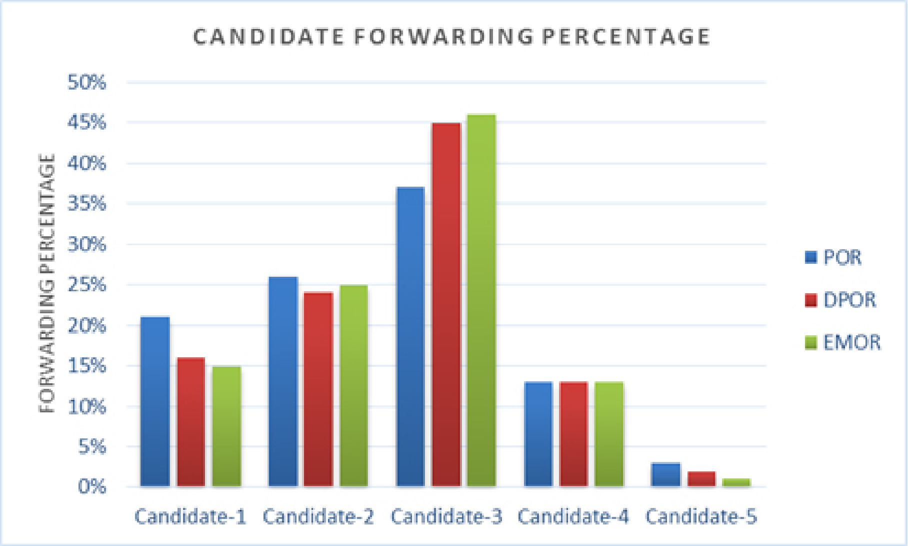


Figure 4.5: Candidate forwarding percentage

the network and saves the resources, the high amount of data transmissions specially in sparse network causes extra overhead. On the other hand, POR, which uses a simple geographical metric, needs more data transmissions compared to EMOR, DPOR, and GPSR.

The expected number of effective data transmissions in Figure 4.9 give us a better comparing scheme. Effective data transmission is only focused on the number of transmissions of the successfully received data packets. In this view, AODV has much better behavior than CBF. In addition, all of the OR protocols have lower expected transmissions, which proves the advantage of employing opportunistic concept in wireless routing. Among all of them, EMOR has the lowest expected number of data transmissions.

Finally, Figure 4.10 displays the role of each candidate in the data packet forwarding in OR protocols (POR, DPOR, and EMOR). As shown in all three charts, the third candidate has the highest forwarding percentage among the other relay nodes. Behavior of the forth and fifth candidate are approximately the same in all of the mobility models. However, the contribution of the first and second forwarding candidates are different from others in Manhattan Grid mobility model due to the characteristic of Grid model. The general purpose of Figure 4.10 is to note that, although forwarding candidates sort based on their

Table 4.2: Simulation-2 parameters

Playground size	500m x 500m
Propagation model	Log-normal Shadowing (mean=0, stdDev=6, path loss=2.2)
MAC layer mode	802.11 g (2.412e+9Hz)
Mobility model	Random Direction, Random Waypoint, Manhattan Grid
Speed	min:5 max:30 (m/s)
Hello interval	Every 1 second
Data traffic	Constant Bit Rate
Number of nodes	20, 30, 40, 50

metrics and have priority to transmit the packet, reaching the best or higher priority candidates is another concerning issue.

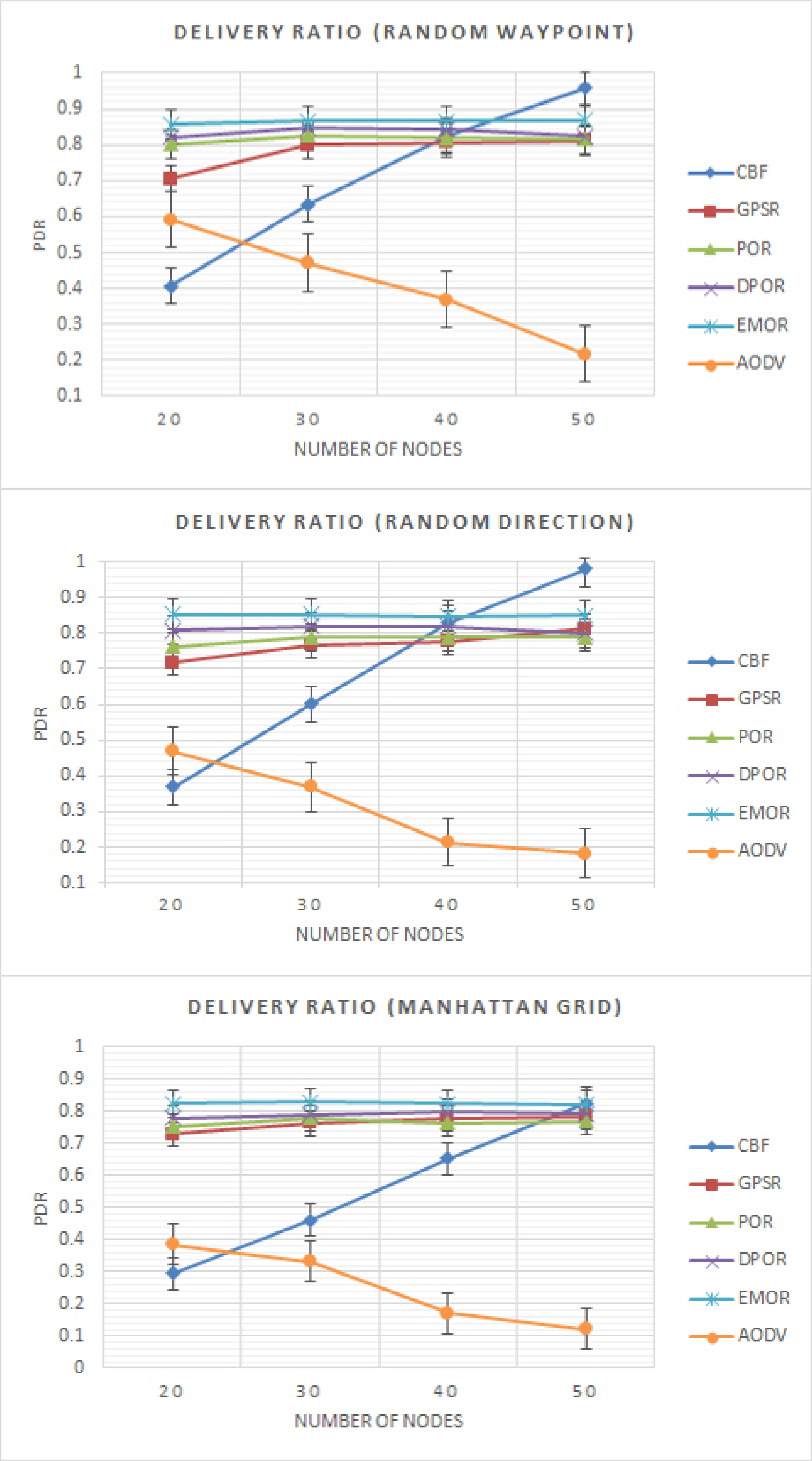


Figure 4.6: Delivery ratio in different mobility models

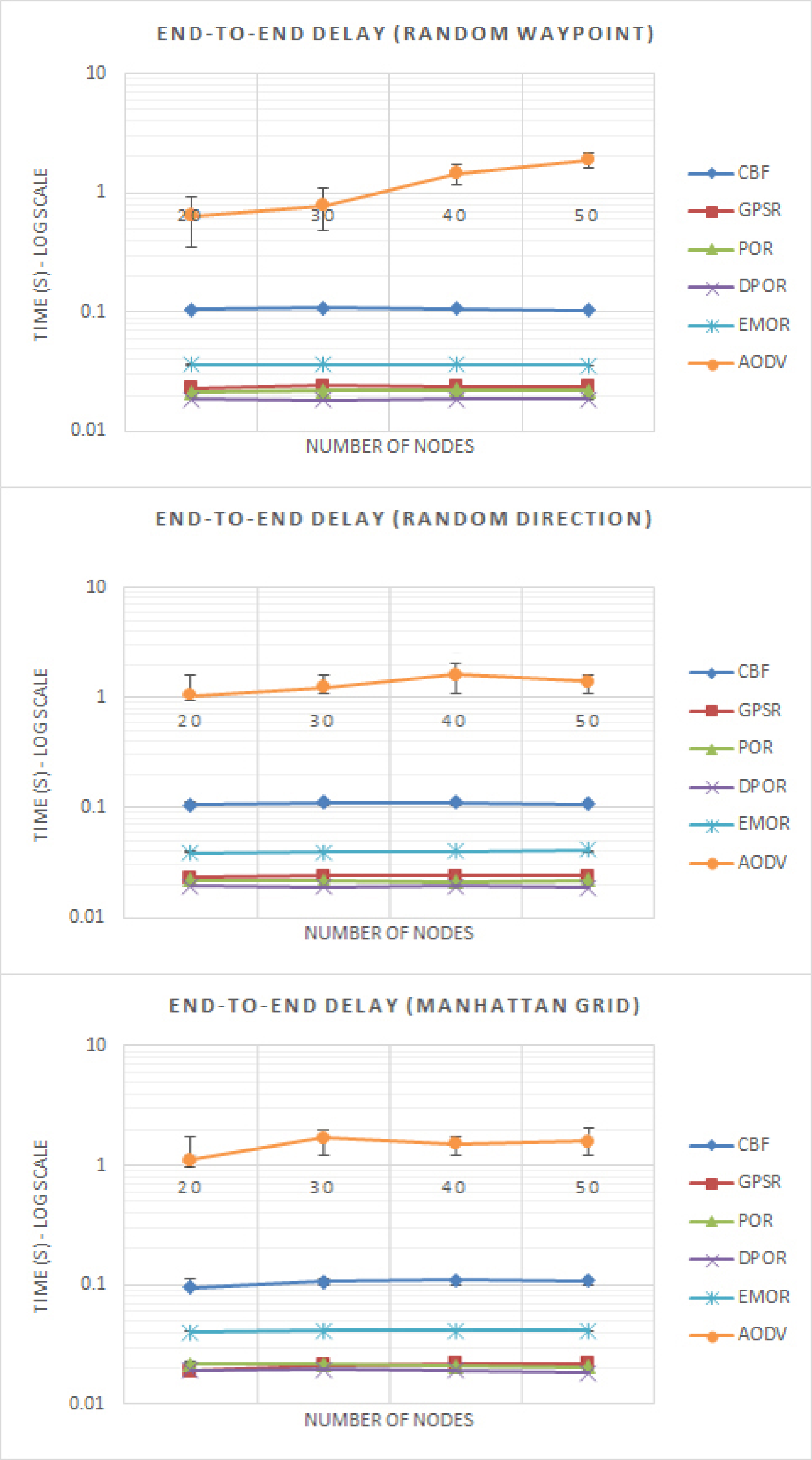
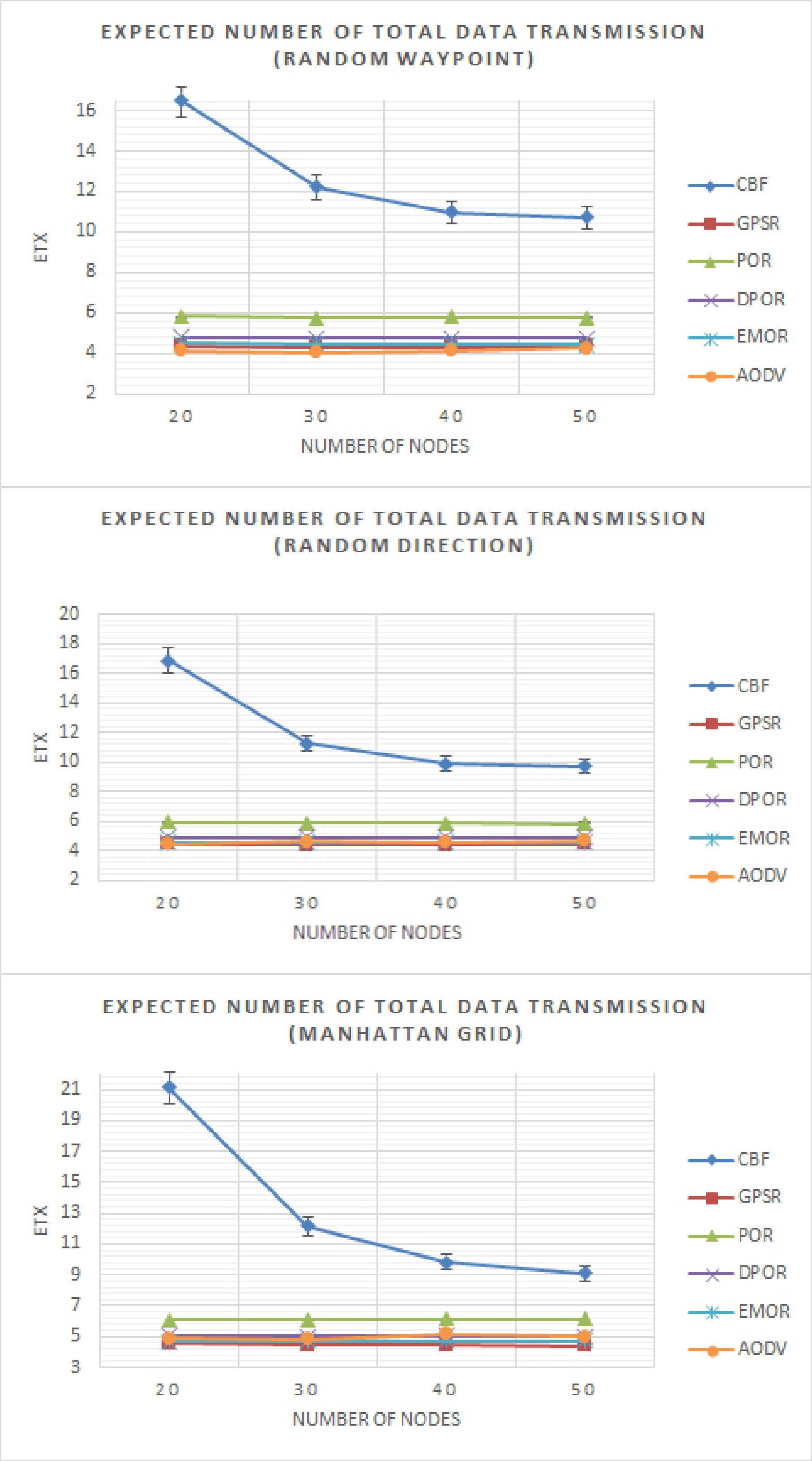


Figure 4.7: End-to-end delay in different mobility models


EXPECTED NUMBER OF TOTAL DATA TRANSMISSION (RANDOM WAYPOINT)

EXPECTED NUMBER OF TOTAL DATA TRANSMISSION (RANDOM DIRECTION)

EXPECTED NUMBER OF TOTAL DATA TRANSMISSION (MANHATTAN GRID)

Figure 4.8: Expected number of total data transmission in different mobility models

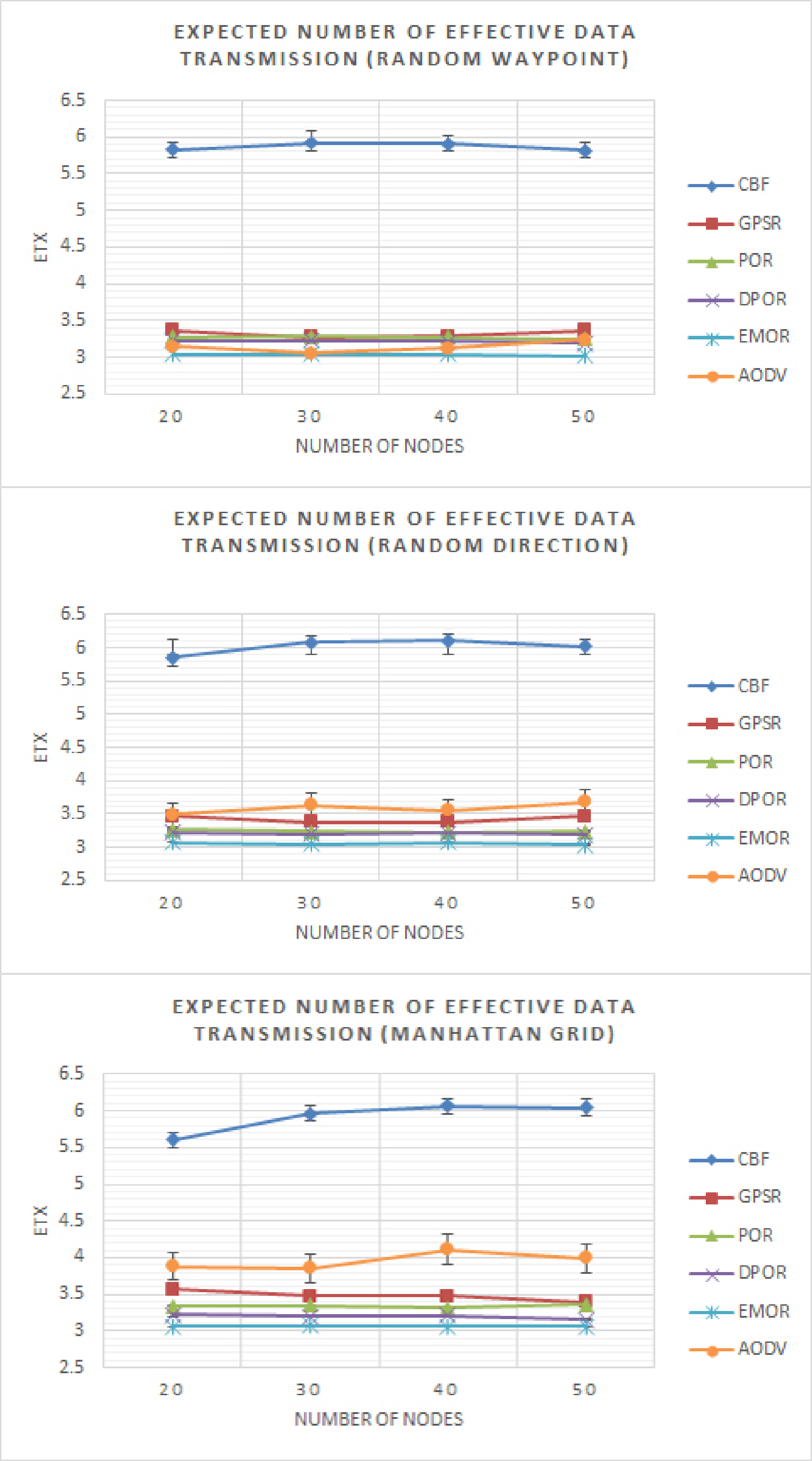


Figure 4.9: Expected number of effective data transmission in different mobility models

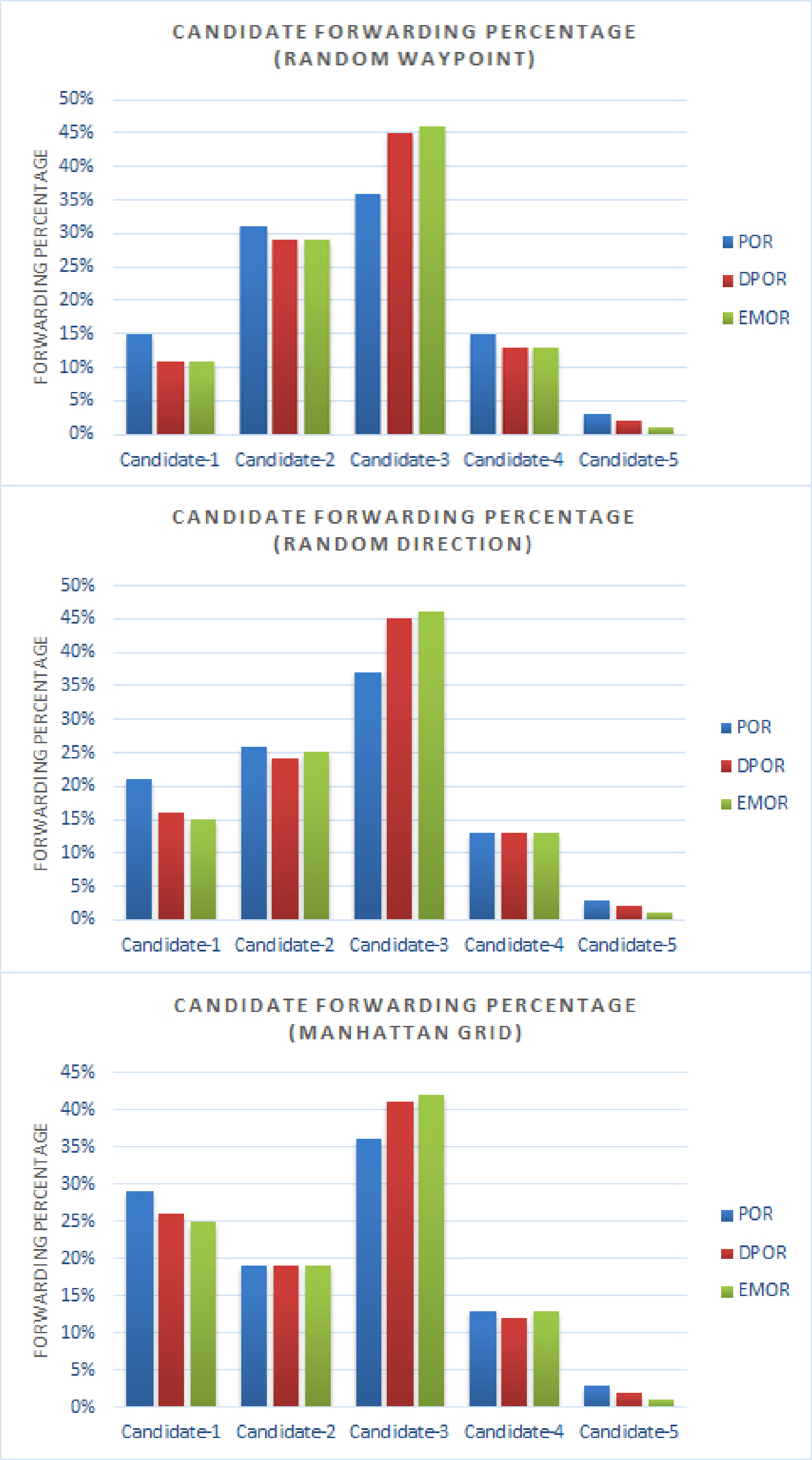


Figure 4.10: Candidate forwarding percentage in different mobility models

Chapter 5

Conclusion and Future Works

In this chapter, we present the conclusion of the proposed opportunistic routing protocol, and mention some future works to extend this survey for further studies.

5.1 Conclusion

Opportunistic routing has considerable potential to be employed over wireless environments with a high rate of packet loss and noise. It uses broadcast nature of transmission to avoid extra re-transmissions; and as a result, it enhances latency and delivery ratio. The proposed EMOR protocol defines a new metric that is adapted for mobile scenarios. It covers not only geographical position of forwarders but also link delivery probability to reach them. In addition, it considers the density of the relay nodes, and position prediction. Moreover, a timer-based coordination algorithm is employed so that source and the forwarders can overhear each other to avoid duplicated transmissions.

Results of the simulations show that the proposed protocol has more stable performances in high speed, and it improves delivery ratio and number of expected transmissions in terms of different type of mobility models.

5.2 Future Works

Although using OR concept can increase the total throughput of the wireless routing protocol and has potential advantages, there are some challenges which need further studies. In this section, we mention some of these challenges faced by the proposed OR.

- In EMOR, hello messages are the main part of unwanted traffic. To decrease the effect of these messages and save the resources, we can consider the procedure to discover the rate of changes over the network. Then, hello timer interval can be set based on these obtained statistics.

- The computational complexity of the proposed candidate selection algorithm is in the order of $O(n^2)$. It seems it may cause delays or inaccuracies in the network with lots of high speed nodes such as urban environment. To address this issue we can minimize the number of forwarding candidates or find an alternative candidate selection algorithm with the same output and lower complexity.

- Using timer-based coordination method will cause long waiting time and maybe starvation for lower priority forwarding candidates. To avoid this latency and according to the diagrams of Figure 4.10, we can give weighted priority to the candidates so that the middle priority nodes that transfer most of the times get higher priority to transmit the packets. This approach will reduce the total end-to-end delay.

- Network coding is a technique which can be used to improve a network's throughput, efficiency and scalability, as well as resilience to attacks and eavesdropping. Instead of simply relaying the packets of information they receive, the nodes of a network take several packets and combine them together for transmission. Then, it is possible to merge some packets or flows together and save the resources by using the advantages of NC.

- For better implementing of OR concept in network layer in terms of coordination, we need to take into consideration some modifications in MAC layer as well. Then, the MAC layer IEEE 802.11 should be modified in form of desirable time slot based.

- Power consumption is another challenging concern which has to be taken into account in a wireless mesh network such as wireless sensor network. In this case, nodes have to turn off their radio most of the times to save their energy. Dealing with throughput and energy saving is a challenging issue that has to be considered in dynamic wireless network as well.

- Last but not least, providing a secure and trusted routing protocol is always a major concern in Multi-hop Wireless Networks. In case there will be a fake or untrusted node over the network, it leads candidate selection and coordination algorithms into a wrong or unsafe decision. There is a progressing trend to figure out the trusted and untrusted nodes, but there is a lot to do yet.

APPENDICES

Appendix A

Candidate Selection and Coordination Source Code

A.1 C++ Source Code implemented in NS-2

The algorithm of candidate selection and coordination of EMOR is as follow:

```
for (int i=0; i<1000; i++){      //start forming neighborList for EMOR
    if (neighborList[i].src_index_ == 0){//not exist in neighborList
        neighborList[i].src_index_ = ph->hello_src_index_;
        neighborList[i].distanceToDestination_ = distance_cd;
        neighborList[i].distanceDifference_ = distance_sd - distance_cd;
        neighborList[i].p_ = ph->p_; //density
        neighborList[i].keepAlive_ = CURRENT_TIME;
        neighborList[i].angle_ = beta_;
        break;
    }
    else if (neighborList[i].src_index_ == ph->hello_src_index_){
```

```
        //already exist in neighborList
        neighborList[i].distanceToDestination_ = distance_cd;
        neighborList[i].distanceDifference_ = distance_sd - distance_cd;
        neighborList[i].p_ = ph->p_;
        neighborList[i].keepAlive_ = CURRENT_TIME;
        neighborList[i].angle_ = beta_;
        break;
    }
}    //end of for

for (int e=0; e<candidateNumber_; e++){
    //clearing candidateList for new EPP calculation
    candidateList[e].src_index_ = -1;
    candidateList[e].distanceToDestination_ = 0;
    candidateList[e].distanceDifference_ = 0;
    candidateList[e].edp_ = 0;
    candidateList[e].etx_ = 0;
    candidateList[e].p_ = 0;
    candidateList[e].keepAlive_ = 0;
    candidateList[e].flag_ = 0;
}    //end of for

for (int i=0; i<1000; i++){     //updating neighborList for all contents
    if (neighborList[i].src_index_ != 0 &&
    (neighborList[i].keepAlive_ + 5 < CURRENT_TIME ||
    neighborList[i].distanceDifference_ < 0 ||
    neighborList[i].p_ < 2 ||
    neighborList[i].angle_ < (alpha_ - 30) ||
    neighborList[i].angle_ > (alpha_ + 30) )){
```

```
    //dead time = 4 x HelloTime  or  neighbor is far than source
    or less that density threshold or out of triangle
        for (int j=i; j<999; j++){      //remove and shift left
            neighborList[j].src_index_ = neighborList[j+1].src_index_;
            neighborList[j].distanceToDestination_ =
            neighborList[j+1].distanceToDestination_;
            neighborList[j].distanceDifference_ =
            neighborList[j+1].distanceDifference_;
            neighborList[j].etx_ = neighborList[j+1].etx_;
            neighborList[j].edp_ = neighborList[j+1].edp_;
            neighborList[j].p_ = neighborList[j+1].p_;
            neighborList[j].keepAlive_ = neighborList[j+1].keepAlive_;
            neighborList[j].flag_ = neighborList[j+1].flag_;
            neighborList[j].angle_ = neighborList[j+1].angle_;
        }
        neighborList[999].src_index_ = 0;
        neighborList[999].distanceToDestination_ = 0;
        neighborList[999].distanceDifference_ = 0;
        neighborList[999].etx_ = 0;
        neighborList[999].edp_ = 0;
        neighborList[999].p_ = 0;
        neighborList[999].keepAlive_ = 0;
        neighborList[999].flag_ = 0;
        neighborList[999].angle_ = 0;
    }
}    //end of for

//start calculating ETX
... (Omitted-refer to the original code for more information)
```

```
//end of calculating ETX

//start calculating EPP

double transmissionRange_ = 100;      //for normalizing data
//double zarib_ = 0.5;     // (a)---x(1-a)---
double etx_notReceivedByOthers_[1000];
double neighbor_edp_max;
double neighbor_edp_last = -1;
double neighbor_edp_temp = 0;
int neighborIndex_ = 0;
int neighborIndexArray_[7] = {1001, 1001, 1001, 1001, 1001, 1001, 1001};

for (int i=0; i<candidateNumber_ ; i++){        //number of candidateList
    //sorting candidateList based on distanceToDestination - bubble sort()
    int flagBit = 1;    // set flag to 1 to start first pass
    for(int m = 1; ( m <= i) && flagBit; m++){      //4
        flagBit = 0;
        for (int n=0; n < i-1; n++){      //3
            if (candidateList[n+1].distanceToDestination_ <
            candidateList[n].distanceToDestination_){
                // swap elements
                candidateListhelp_.src_index_ = candidateList[n].src_index_;
                candidateListhelp_.distanceToDestination_ =
                candidateList[n].distanceToDestination_;
                candidateListhelp_.distanceDifference_ =
                candidateList[n].distanceDifference_;
                candidateListhelp_.etx_ = candidateList[n].etx_;
                candidateListhelp_.edp_ = candidateList[n].edp_;
```

```
                candidateListhelp_.p_ = candidateList[n].p_;
                candidateListhelp_.keepAlive_ = candidateList[n].keepAlive_;
                candidateListhelp_.flag_ = candidateList[n].flag_;

                candidateList[n].src_index_ = candidateList[n+1].src_index_;
                candidateList[n].distanceToDestination_ =
                candidateList[n+1].distanceToDestination_;
                candidateList[n].distanceDifference_ =
                candidateList[n+1].distanceDifference_;
                candidateList[n].etx_ = candidateList[n+1].etx_;
                candidateList[n].edp_ = candidateList[n+1].edp_;
                candidateList[n].p_ = candidateList[n+1].p_;
                candidateList[n].keepAlive_ = candidateList[n+1].keepAlive_;
                candidateList[n].flag_ = candidateList[n+1].flag_;

                candidateList[n+1].src_index_ = candidateListhelp_.src_index_;
                candidateList[n+1].distanceToDestination_ =
                candidateListhelp_.distanceToDestination_;
                candidateList[n+1].distanceDifference_ =
                candidateListhelp_.distanceDifference_;
                candidateList[n+1].etx_ = candidateListhelp_.etx_;
                candidateList[n+1].edp_ = candidateListhelp_.edp_;
                candidateList[n+1].p_ = candidateListhelp_.p_;
                candidateList[n+1].keepAlive_ = candidateListhelp_.keepAlive_;
                candidateList[n+1].flag_ = candidateListhelp_.flag_;
                flagBit = 1;               // indicates that a swap occurred
            }
        }
    }
```

```
for (int q=0; q<1000; q++)       //reseting last edp
    neighborList[q].edp_ = neighbor_edp_temp;
neighbor_edp_max = -1;
for (int j=0; j<1000; j++){      //for all of neighbors
    if (neighborList[j].src_index_ != 0 &&
    neighborIndexArray_[0] != j &&
    neighborIndexArray_[1] != j &&
    neighborIndexArray_[2] != j &&
    neighborIndexArray_[3] != j &&
    neighborIndexArray_[4] != j &&
    neighborIndexArray_[5] != j &&
    neighborIndexArray_[6] != j){
    //for not empty neighborList and already selected
        if (i==0){
            etx_notReceivedByOthers_[j] = 1;
        }
        else {
            etx_notReceivedByOthers_[j] =
            etx_notReceivedByOthers_[j] * ( 1 - candidateList[i-1].etx_ );
        }
        neighborList[j].edp_ =
        neighborList[j].edp_ +
        ((double)(neighborList[j].distanceDifference_ /
        transmissionRange_)) * neighborList[j].etx_ *
        etx_notReceivedByOthers_[j];

        if(neighborList[j].edp_ > neighbor_edp_max){     //taking max edp
            neighbor_edp_max = neighborList[j].edp_;
```

```
                neighborIndex_ = j;
                neighbor_edp_temp = neighbor_edp_max;
            }
        }       //end of not empty neighnorList
    }       //end of for neighborList

    if (neighbor_edp_last < neighbor_edp_max){
        //checking current edp with last edp
        neighbor_edp_last = neighbor_edp_max;       //update neighbor_edp_last
        candidateList[i].src_index_ = neighborList[neighborIndex_].src_index_;
        candidateList[i].distanceToDestination_ =
        neighborList[neighborIndex_].distanceToDestination_;
        candidateList[i].distanceDifference_ =
        neighborList[neighborIndex_].distanceDifference_;
        candidateList[i].etx_ = neighborList[neighborIndex_].etx_;
        candidateList[i].edp_ = neighborList[neighborIndex_].edp_;
        candidateList[i].p_ = neighborList[neighborIndex_].p_;
        candidateList[i].keepAlive_ = neighborList[neighborIndex_].keepAlive_;
        candidateList[i].flag_ = neighborList[neighborIndex_].flag_;
        neighborIndexArray_[i] = neighborIndex_;
        //do not consider previous selection
    }       //end of if checking edp
    else
        break;
}       //end of for candidateList
//end of calculating EPP
```

References

[1] Universal mobile telecommunicatios system (umts) - selection procedures for the choice of radio transmission technologies of the umts. Technical report, European Telecommunications Standards Institute (ETSI), 1998.

[2] A. Boukerche A. Darehshoorzadeh. Survey paper: Opportunistic routing protocols in wireless networks: A performance comparison. *WCNC*, 2013.

[3] R. Ahlswede, N. Cai, S. Y. R. Li, and R. W. Yeung. Network information flow. *Information Theory, IEEE Transactions on*, 46(4):1204–1216, July 2000.

[4] S. Biswas and R. Morris. ExOR: opportunistic multi-hop routing for wireless networks. *ACM SIGCOMM Computer Communication Review*, 35(4):133–144, 2005.

[5] A. Boukerche. Performance evaluation of routing protocols for ad hoc wireless networks. *Mobile Networks and Applications*, 9(4):333–342, 2004.

[6] A. Boukerche and L. Bononi. *Simulation and Modeling of Wireless, Mobile, and AD HOC Networks*, pages 373–409. John Wiley Sons, Inc., 2005.

[7] A. Boukerche, S. K. Das, and A. Fabbri. Swimnet: A scalable parallel simulation testbed for wireless and mobile networks. *Wireless Networks*, 7:2001, 2001.

[8] A. Boukerche and X. Fei. A coverage-preserving scheme for wireless sensor network with irregular sensing range. *Ad Hoc Networks*, 5(8):1303 – 1316, 2007. Recent Research Directions in Wireless Ad Hoc Networking.

[9] A. Boukerche and X. Li. An agent-based trust and reputation management scheme for wireless sensor networks. In *Global Telecommunications Conference, 2005. GLOBECOM '05. IEEE*, volume 3, pages 5 pp.–, Nov 2005.

[10] A. Boukerche, R. W. N. Pazzi, and R. B. Araujo. Hpeq a hierarchical periodic, event-driven and query-based wireless sensor network protocol. In *Local Computer Networks, 2005. 30th Anniversary. The IEEE Conference on*, pages 560–567, Nov 2005.

[11] A. Boukerche and Y. Ren. A secure mobile healthcare system using trust-based multicast scheme. *Selected Areas in Communications, IEEE Journal on*, 27(4):387–399, May 2009.

[12] R. Bruno and M. Nurchis. Survey on Diversity Based Routing in Wireless Mesh Networks: Challenges and Solutions. *Computer Communications*, 2009.

[13] S. Chachulski, M. Jennings, S. Katti, and D. Katabi. Trading structure for randomness in wireless opportunistic routing. In *SIGCOMM '07: Proceedings of the 2007 conference on Applications, technologies, architectures, and protocols for computer communications*, pages 169–180, New York, NY, USA, 2007. ACM.

[14] A. Darehshoorzadeh, M. Almulla, A. Boukerche, and S. Chaiwala. On the number of candidates in opportunistic routing for multi-hop wireless networks. In *Proceedings of the 11th ACM International Symposium on Mobility Management and Wireless Access*, MobiWac '13, pages 9–16, New York, NY, USA, 2013. ACM.

[15] A. Darehshoorzadeh and L. Cerdà-Alabern. Candidate selection algorithms in opportunistic routing. In *PM2HW2N '10: Proceedings of the 5th ACM workshop on Performance monitoring and measurement of heterogeneous wireless and wired networks*, pages 48–54, New York, NY, USA, 2010. ACM.

[16] A. Darehshoorzadeh and L. Cerdà-Alabern. Distance progress based opportunistic routing for wireless mesh networks. In *Wireless Communications and Mobile Computing Conference (IWCMC), 2012 8th International*, pages 179–184, 2012.

[17] A. Darehshoorzadeh, L. Cerdà-Alabern, and V. Pla. Modeling and comparison of candidate selection algorithms in opportunistic routing. *Computer Networks*, 55(13):2886 – 2898, 2011.

[18] D. S. J. De Couto, D. Aguayo, J. Bicket, and R. Morris. A high-throughput path metric for multi-hop wireless routing. *Wireless Networks*, 11(4):419–434, 2005.

[19] H. A. de Oliveira, E.F. Nakamura, A. A. F. Loureiro, and A. Boukerche. Directed position estimation: a recursive localization approach for wireless sensor networks. In *Computer Communications and Networks, 2005. ICCCN 2005. Proceedings. 14th International Conference on*, pages 557–562, Oct 2005.

[20] R. Draves, J. Padhye, and B. Zill. Routing in multi-radio, multi-hop wireless mesh networks. In *MobiCom '04: Proceedings of the 10th annual international conference on Mobile computing and networking*, pages 114–128, New York, NY, USA, 2004. ACM.

[21] H. Dubois-Ferriere, M. Grossglauser, and M. Vetterli. Least-cost opportunistic routing. In *Proceedings of 2007 Allerton Conference on Communication, Control, and Computing*, 2007.

[22] H. Dubois-Ferriére, M. Grossglauser, and M. Vetterli. Valuable detours: Least-cost anypath routing. *Networking, IEEE/ACM Transactions on*, 19(2):333 –346, Apr 2011.

[23] H. Füßler, J. Widmer, M. Käsemann, M. Mauve, and H. Hartenstein. Contention-based forwarding for mobile ad hoc networks. *Ad Hoc Networks*, 1(4):351 – 369, 2003.

[24] E. Ghadimi, O. Landsiedel, P. Soldati, and M. Johansson. A metric for opportunistic routing in duty cycled wireless sensor networks. In *Sensor, Mesh and Ad Hoc Communications and Networks (SECON), 2012 9th Annual IEEE Communications Society Conference on*, pages 335 –343, Jun 2012.

[25] Ch. Hsu, H. Liu, and W. K. G. Seah. Survey paper: Opportunistic routing - a review and the challenges ahead. *Comput. Netw.*, 55(15):3592–3603, October 2011.

[26] Sh. Jain and S. R. Das. Exploiting path diversity in the link layer in wireless ad hoc networks. *Ad Hoc Netw.*, 6(5):805–825, 2008.

[27] B. Karp and H. T. Kung. Gpsr: greedy perimeter stateless routing for wireless networks. In *Proceedings of the 6th annual international conference on Mobile computing and networking*, MobiCom '00, pages 243–254, New York, NY, USA, 2000. ACM.

[28] S. Katti, H. Rahul, W. Hu, D. Katabi, M. Médard, and J. Crowcroft. XORs in the air: practical wireless network coding. *IEEE/ACM Trans. Netw.*, 16(3):497–510, 2008.

[29] D. Koutsonikolas, Y. Charlie Hu, and Ch. Wang. Xcor: Synergistic interflow network coding and opportunistic routing. In *ACM MobiCom'08 SRC*, 2008.

[30] P. Larsson. Selection diversity forwarding in a multihop packet radio network with fading channel and capture. *SIGMOBILE Mob. Comput. Commun. Rev.*, 5(4):47–54, 2001.

[31] K. C. Lee, U. Lee, and M. Gerla. To-go: Topology-assist geo-opportunistic routing in urban vehicular grids. In *Wireless On-Demand Network Systems and Services, 2009. WONS 2009. Sixth International Conference on*, pages 11 –18, Feb. 2009.

[32] Y. Li, W. Chen, and Z. Zhang. Optimal forwarder list selection in opportunistic routing. In *Mobile Adhoc and Sensor Systems. MASS '09. IEEE 6th International Conference on*, pages 670 –675, Oct. 2009.

[33] Y. Li, Y. Liu, and P. Luo. Link probability based opportunistic routing metric in wireless network. In *Communications and Mobile Computing, 2009. CMC '09. WRI International Conference on*, volume 2, pages 308–312, January 2009.

[34] L. Liu, Z. Wang, and W. Jehng. A geographic source routing protocol for traffic sensing in urban environment. In *Automation Science and Engineering, 2008. CASE 2008. IEEE International Conference on*, pages 347–352, Aug 2008.

[35] Z. Liu, T. Zhao, Y. Wei, and X. Li. Gosr: geographical opportunistic source routing for vanets. *Mobile Computing and Communications Review*, 13(1):48–51, 2009.

[36] M. Lu and J. Wu. Opportunistic routing algebra and its applications. In *IEEE INFOCOM*, pages 2374–2382, 2009.

[37] Ch. Luk, W. Lau, and O. Yue. An analysis of opportunistic routing in wireless mesh network. In *Communications, 2008. ICC '08. IEEE International Conference on*, pages 2877–2883, May 2008.

[38] M.S. Nassr, J. Jun, S. J. Eidenbenz, A. A. Hansson, and A.M. Mielke. Scalable and reliable sensor network routing: Performance study from field deployment. pages 670 –678, May. 2007.

[39] C. Perkins, E. Belding-Royer, and S. Das. RFC 3561: Ad hoc On-Demand Distance Vector (AODV) Routing, Jul 2003.

[40] Ch. Perkins and E. Royer. Ad-hoc on-demand distance vector routing. In *Proceedings of the 2nd IEEE Workshop on Mobile Computing Systems and Applications*, pages 90–100, 1997.

[41] T. S. Rappaport. Wireless communications, principles and practice, prentice hall, 1996.

[42] Y. Ren and A. Boukerche. Modeling and managing the trust for wireless and mobile ad hoc networks. In *Communications, 2008. ICC '08. IEEE International Conference on*, pages 2129–2133, May 2008.

[43] E. Rozner, J. Seshadri, Y. Ashok Mehta, and L. Qiu. Soar: Simple opportunistic adaptive routing protocol for wireless mesh networks. *IEEE Transactions on Mobile Computing*, 8(12):1622–1635, 2009.

[44] J. Boleng T. Camp and V. Davies. Survey of mobility models for ad hoc network research. *Wireless Communication and Mobile Computing (WCMC)*, 2(5):483–502, 2002.

[45] C. Westphal. Opportunistic routing in dynamic ad hoc networks: the oprah protocol. In *Mobile Adhoc and Sensor Systems (MASS), 2006 IEEE International Conference on*, pages 570–573, October 2006.

[46] J. Wu, M. Lu, and F. Li. Utility-based opportunistic routing in multi-hop wireless networks. In *Distributed Computing Systems, 2008. ICDCS'08. The 28th International Conference on*, pages 470–477, 2008.

[47] Y. Yan, B. Zhang, H. T. Mouftah, and J. Ma. Practical coding-aware mechanism for opportunistic routing in wireless mesh networks. In *Communications, 2008. ICC '08. IEEE International Conference on*, pages 2871–2876, May 2008.

[48] Sh. Yang, F. Zhong, Ch. Yeo, B. Lee, and J. Boleng. Position based opportunistic routing for robust data delivery in manets. In *Proceedings of the 28th IEEE conference on Global telecommunications*, GLOBECOM'09, pages 1325–1330, Piscataway, NJ, USA, 2009. IEEE Press.

[49] Z. Yang, K. Zeng, and W. Lou. FSA: A Fast Coordination Scheme for Opportunistic Routing. In *2009 IEEE International Conference on Communications*, pages 1–5. IEEE, June 2009.

[50] Y. Yuan, H. Yang, S. H. Y. Wong, S. Lu, and W. Arbaugh. ROMER: Resilient opportunistic mesh routing for wireless mesh networks. In *IEEE Workshop on Wireless Mesh Networks (WiMesh)*, 2005.

[51] K. Zeng, W. Lou, J. Yang, and D. R. Brown. On Throughput Efficiency of Geographic Opportunistic Routing in Multihop Wireless Networks. *Mobile Networks and Applications*, 12(5-6):347–357, April 2008.

[52] K. Zeng, W. Lou, and H. Zhai. Capacity of opportunistic routing in multi-rate and multi-hop wireless networks. *Wireless Communications, IEEE Transactions on*, 7(12):5118–5128, Dec 2008.

[53] K. Zeng, W. Lou, and Y. Zhang. Multi-rate geographic opportunistic routing in wireless ad hoc networks. In *Military Communications Conference, 2007. MILCOM 2007. IEEE*, pages 1 –7, Oct. 2007.

[54] K. Zeng, Z. Yang, and W. Lou. Location-aided opportunistic forwarding in multi-rate and multihop wireless networks. *Vehicular Technology, IEEE Transactions on*, 58(6):3032 –3040, Jul 2009.

[55] Z. Zhong, J. Wang, and S. Nelakuditi. Opportunistic any-path forwarding in multi-hop wireless mesh networks.

[56] Z. Zhong, J. Wang, S. Nelakuditi, and G. Lu. On selection of candidates for opportunistic anypath forwarding. *SIGMOBILE Mob. Comput. Commun. Rev.*, 10(4):1–2, 2006.

[57] M. Zorzi and R. R. Rao. Geographic random forwarding (geraf) for ad hoc and sensor networks: multihop performance. *Mobile Computing, IEEE Transactions on*, 2(4):337–348, Oct.-Dec. 2003.