BRNO UNIVERSITY OF TECHNOLOGY
VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION
FAKULTA ELEKTROTECHNIKYA KOMUNIKAČNÍCH TECHNOLOGIÍ

DEPARTMENT OF TELECOMMUNICATIONS
ÚSTAV TELEKOMUNIKACÍ

MOVEMENT PREDICTION OF WIRELESS NODES IN MOBILE AD HOC NETWORKS (MANETS)
PREDIKCE POHYBU BEZDRÁTOVÝCH UZLŮ V MOBILNÍCH AD HOC SÍTÍCH (MANET)

DOCTORAL THESIS
DIZERTAČNÍ PRÁCE

AUTHOR
AUTOR PRÁCE
Eng. Nermin Makhlouf

SUPERVISOR
ŠKOLITEL
doc. Ing. Jaroslav Koton, Ph.D.

BRNO 2018

ACKNOWLEDGMENTS

Firstly, I would like to express my sincere gratitude to my supervisor, doc. Ing. Jaroslav Koton, Ph.D. for the professional support of my Ph.D study and related research.

My sincere thanks also go to all my friends and colleagues in Department of Telecommunications, Brno University of Technology.

I would like to thank my family, my parents and siblings; you were always supporting and encouraging me with your best wishes.

To my cute little daughter, Celena, thank you for being good kid with your mother when I was studying in the school.

And finally, I do not know how to begin with saying thank you to my soul mate, my dearest husband Samer. You have been a constant source of support and encouragement. Great appreciation and enormous thanks are due to you, for without your understanding, this thesis would never have been completed.

I thank you all.

Abstract

The rapid evolution in the field of mobile computing has led to a new alternative way for mobile communication, in which mobile nodes form a self-organising wireless network, called a Mobile Ad hoc Network (MANET). The specific characteristics of MANETs impose many challenges to network protocol designs on all layers of the protocol stack because of unpredictable topology changes and mobile nature. Mobility prediction is a tool to deal with the problems emerging from the nodes' mobility by predicting future changes in the network topology. This is crucial for different tasks such as routing.

In this doctoral thesis, two mobility prediction methods for MANET networks are developed. The first method supposes that each node can build its virtual map depending on its location over the time. This method is called mobility prediction using virtual map. In order to evaluate the developed prediction algorithm, it has been implemented in the network simulator NS-2. I have investigated existing mobility models, and how the prediction method can be applied to them. Simulations respectively realize performance improvement in terms of average end to end delay, packet delivery ratio and network throughput under different mobility model. The proposed prediction concept is implemented over AODV (Ad Hoc On-Demand Distance Vector) routing protocol.

In the second method, I have developed an artificial neural network for movement prediction in MANETs. The prediction model for mobility has been done by the data collected from location patterns. The Bayesian technique was used for learning or training ANNs. It has been implemented in software for training Bayesian neural networks called Model Manager. The best way to evaluate the final model is done by making predictions and comparing predictions with target data. The predictions are made by using 50 patterns as input variables.

The reached and in the thesis discussed results show that improvement in the most significant network parameters, i.e. delay, throughput and packet delivery ratio, are reached even by 30% compared to AODV routing protocol, where the proposed prediction model is not utilized.

Keywords

Artificial neural networks, Directional antenna, Global Positioning System (GPS) coordinates, MANETs, Mobility model, Mobility prediction, Routing protocol.

Abstrakt

Rychlý vývoj v oblasti mobilní informatiky vyústil v nový, alternativní způsob mobilní komunikace, v němž mobilní uzly tvoří samoorganizující se bezdrátovou síť, jíž se říká mobilní síť ad hoc (Mobile Ad hoc Network, MANET). Specifické vlastnosti sítí MANET stavějí návrh síťového protokolu před řadu problémů na všech vrstvách protokolové sady . Příčinou jsou nepředvídatelné změny topologie a mobilní povaha těchto sítí. Nástrojem, který řeší problémy plynoucí z mobility uzlů, je predikce budoucích změn v topologii sítě. To má zásadní význam pro různé úlohy jako přesměrování.

Tato disertační práce se zabývá dvěma metodami predikce mobility pro sítě MANET. První metoda se nazývá „predikce mobility s využitím virtuální mapy“ (mobility prediction using virtual map) a předpokládá, že každý uzel si dokáže vybudovat svou virtuální mapu v závislosti na svém umístění v průběhu času. Vyvinutý predikční algoritmus byl implementován do síťového simulátoru NS-2, aby jej bylo možné vyhodnotit. V této práci zkoumám stávající modely mobility a způsob, jakým v nich lze aplikovat tuto metodu predikce. Simulace sledují zlepšení výkonnosti, co se týče průměrného zpoždění na bázi end-to-end, poměru doručených paketů a propustnosti sítě. Navržený koncept predikce byl implementován pomocí směrovacího protokolu AODV(Ad Hoc On-Demand Distance Vector).

Pro druhou metodu jsem vyvinula umělou neuronovou síť pro predikci pohybů v sítích MANET. Model pro predikci mobility vznikl na základě dat shromážděných ze vzorců umístění. K učení či trénování ANN byl využit bayesovský přístup. Ten byl implementován v softwaru pro trénování bayesovských neuronových sítí s názvem Model Manager. Nejlepším způsobem hodnocení závěrečného modelu je provedení predikcí a jejich srovnání s cílovými daty. Predikce vznikají na základě 50 vzorců jako vstupních proměnných.

Dosažené výsledky prezentované s diskutované v práci se vyznačují zlepšením zásadních parametrů komunikační sítě, jako jsou propustnost, zpoždění, Poměr doručených paketů, až o 30% v porovnání s klasickým směrovacím protokolem AODV, kde není implementován predikční model.

Klíčová slova

Umělé neuronové sítě, směrová anténa, souřadnice systému GPS (Global Positioning System), sítě MANET, model mobility, predikce mobility, směrovací protokol.

Bibliographic citation

Makhlouf, N. Movement prediction of wireless nodes in mobile ad hoc networks (MANETS). Brno: Brno University of Technology, Faculty of Electrical Engineering and Communication, 2018. 123 p. Ph.D. Thesis. Supervisor doc. Ing. Jaroslav Koton, Ph.D.

DECLARATION

I hereby confirm that I have elaborated my doctoral thesis on the theme of "Movement prediction of wireless nodes in mobile ad hoc networks (MANETS)" independently, under the supervision of doc. Ing. Jaroslav Koton, Ph.D. and with the use of technical literature and other sources of information which are all quoted in the thesis and detailed in the list of literature at the end of the thesis.

Brno 12.12.2018 …………………………

(Author’s signature)

Faculty of Electrical Engineering and Communication
Brno University of Technology
Technicka 12, CZ-61600 Brno,
Czech Republic

http://www.six.feec.vutbr.cz

ACKNOWLEDGEMENT

Research described in this doctoral thesis has been implemented in the laboratories supported by the SIX project; reg. no. CZ.1.05/2.1.00/03.0072, operational program Výzkum a vývoj pro inovace.

Brno 12.12.2018

…………………………

(Author’s signature)

MINISTERSTVO ŠKOLSTVÍ, MLÁDEŽE A TĚLOVÝCHOVY

EVROPSKÁ UNIE
EVROPSKÝ FOND PRO REGIONÁLNÍ ROZVOJ
INVESTICE DO VAŠÍ BUDOUCNOSTI

OP Výzkum a vývoj pro inovace

List of contents

LIST OF FIGURES

LIST OF TABLES

1 INTRODUCTION

Mobile Ad hoc NETworks (MANETs) have attracted the attention of the scientific community for more than three decades. MANET is a wireless network of Mobile Nodes (MNs) connected by a wireless link without central control [1]-[3]. The mobile node can be carried by people or can be on autonomous system, e.g. vehicle. Each node in a MANET can move independently in any direction, therefore links to other devices may change frequently. Furthermore, each node makes its decision based on the network situation, without any reference infrastructure and thus nodes can behave as routers or hosts. MANETs are Multi-Hop wireless networks since a node may not be able to connect directly with other nodes which are out of its range. In such cases, the data packets from the source need to travel through a number of nodes (hops) to reach the destination. The intermediate nodes between the source and destination behave as routers [4], [5]. Since the nodes in MANET move continually, there are weak and untrustworthy links between them. The design of the network layer protocols has been extensively studied [6]-[12].

Dynamic topology is a special feature of a mobile ad hoc network. Links between nodes are created and broken, as the nodes move within the network. These frequent changes in topology affect the performance of ad hoc networks. A mobility model is designed to describe the movement of MNs and how their location, speed, and acceleration change over time. Many mobility models are studied in [13]-[15]. The mobility model should be able to mimic the real movement of MNs.

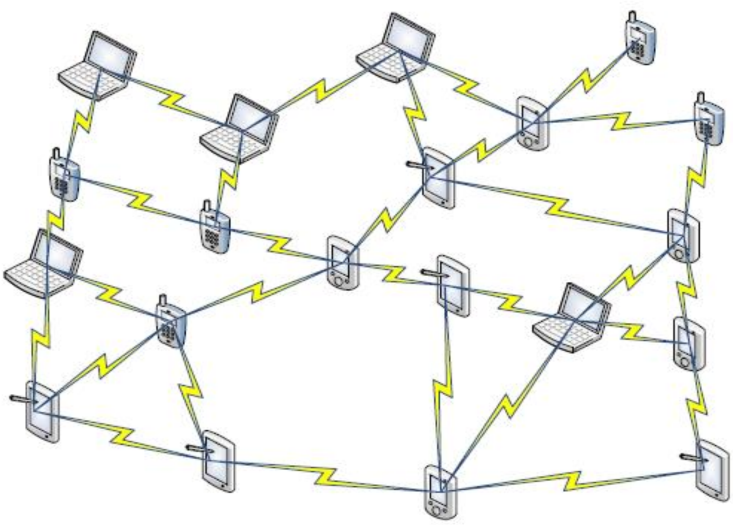
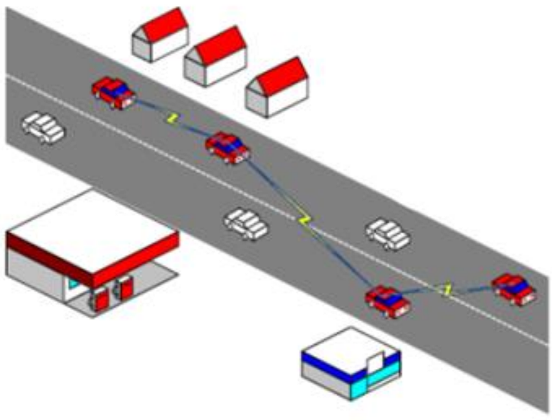


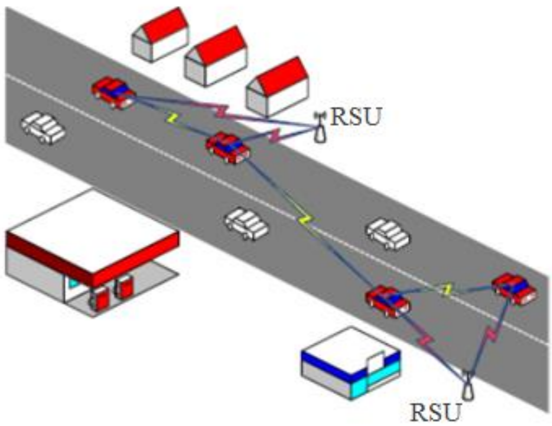
Fig. 1.1 An example of an ad hoc network

Generally, ad hoc networks can be applied in any situation where deployment of a fixed communication infrastructure is not possible or a temporary communication is required. MANETs are used for commercial environment, education, entertainment, disaster scenarios and military applications, where the fixed communication infrastructure is not expected to be used in a long time period [16]. In such situations, the MANETs can be advantageously used where other technologies either fail or cannot be effectively deployed. Since the nodes are mobile; they are allowed to move freely. This produces frequent connectivity changes. In such dynamic topology, some pairs of nodes may not be able to connect directly with each other, so they use some intermediate nodes to connect to their destinations. These networks are called multi-hops networks. Fig. 1.1 shows an example of a MANET which has different devices (nodes). A MANET's node may be a Personal Digital Assistant (PDA), laptop, mobile phone, and other wireless device carried by high-speed vehicles.

A Vehicular Ad hoc NETwork (VANET) is special kind mobile ad hoc networks where wireless transceiver in vehicles form a network with the RoadSide Unit (RSU) without any additional infrastructure [17], [18]. The vehicles can communicate with each other, these types of communication are called Vehicle to Vehicle (V2V) communication and they represent the main communications in VANETs, Fig. 1.2.a. The communication between the vehicle and the roadside unit (road infrastructure) are called Vehicle to Infrastructure (V2I) communications, Fig. 1.2.b.



(a)



(b)

Fig. 1.2 VANET networks: a)Vehicle to Vehicle communications, b) Vehicle to Infrastructure communications

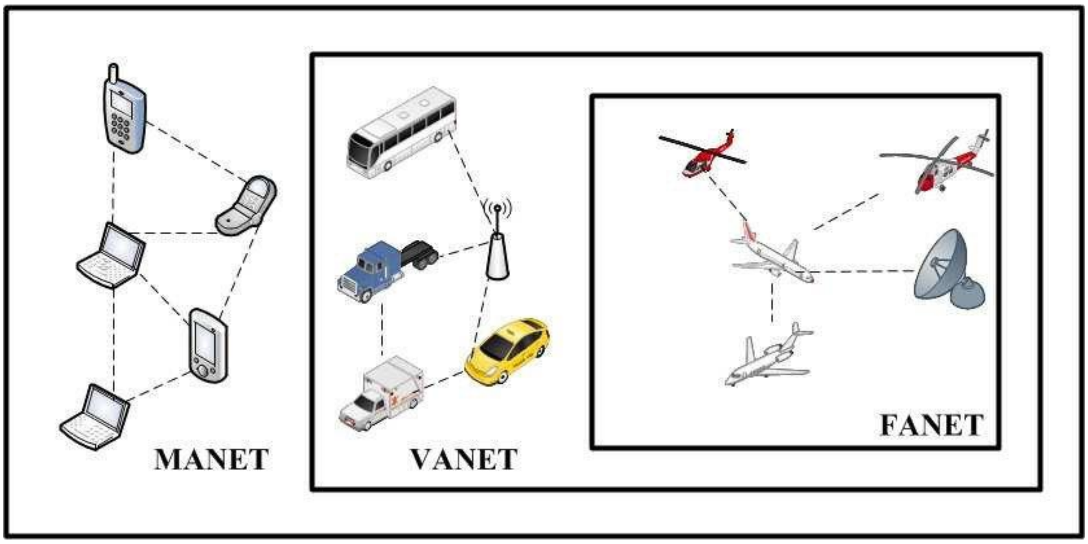


Fig. 1.3 . MANET, VANET and FANET [153]

Flying Ad hoc Networks (FANETs) are the ad hoc nodes operating in air comprising of Unmanned Air Vehicles (UAVs) which can fly autonomously or can be operated remotely without carrying any human personnel [153]. These UAVs, if used as network nodes have a crucial advantage of formation of an aerial mesh. FANET is used in both civilian and military applications, such as traffic monitoring, wind estimation, disaster monitoring and border surveillance. FANET is a subset of VANET. The relationship among MANET, VANET and FANET is illustrated in Fig. 1.3.

Generally speaking about MANETs, in the case that a node wants to send data, it needs to find the location of the destination. While data is being transmitted, the node may move, and hence the transmission fails, and the transmitter has to retransmit the data using the transport layer. Because data transmission and reception require some energy to perform this action and the MNs are powered by a battery, it is very important to use energy as efficiently as possible in all layers of the protocol stack [19]. The energy efficiency of the Medium Access Protocol (MAC) for the MANET networks is completely different from what it is in the MAC for fixed network and cellular network as e.g. discussed in [6], [20], [21]. The features of MANET may be summarized as follows:

- The dynamic nature of network where nodes move arbitrarily and hence the topology changes.
- Infrastructure-less in contrast to with the cellular network.
- Nodes in MANET share the wireless medium and this medium is not protected from outside signals.

- Each node can forward data packets to other nodes, so it acts as a router.

One way to address the effects of node mobility is estimating the movement of the MNs. Mobile prediction is a method for estimating the trajectory of the future position of the nodes. This topic has been studied in various fields, such as cellular networks and routing for wireless mobile ad hoc networks [22]-[25]. It is clear that the application of cellular networks operates with more different prerequisites for mobility prediction than for ad hoc networks, the hardware of the networks and the behaviour of the nodes are radically different. However, the problem of mobility prediction is the same, whether used in wireless networks with fixed infrastructure or in wireless mobile ad hoc networks. If topology change can be predicted, then path reestablishment can be completed prior to a topology change. A mobility prediction scheme for mobile ad hoc networks should present accurate prediction with minimal control overhead. The study and analysis of MANET have been carried out by simulation analysis. Network simulators are widely used to evaluate the performance of MANETs like NS-2, NS-3, and OPNET, among others [26]. Because the real testbeds need a high investment in terms of hardware, and, more importantly, the replication of real mobile conditions is very difficult in a controlled environment like a laboratory.

This thesis deals with defining a new prediction model and its utilization in AODV routing protocol to provide the improvement of the MANET network parameters. After this introduction, the thesis is further organized in 6 main chapters where Chapter 2 includes state of art. Chapter 3 introduces the mobility models that have been proposed for MANETs. Chapter 4 presents some mobility prediction methods that have been proposed for an ad hoc network The Chapter 4 also reviews the challenges at medium access control caused by integration of smart antennas system in ad hoc networks. The objectives of this research are presented in Chapter 5. Chapter 6 includes the research methodology of the dissertation and describes the proposed mobility prediction based on the information of current network status. Chapter 7 provides the analysis of the simulation results, and finally, Chapter 8 concludes this thesis.

2 STATE OF ART

Due to rapid topology changing and frequent disconnection, the mobility prediction helps to develop a snapshot of the future network topology. Therefore, it minimizes location updating, thereby reducing communication delay and improving the Quality of Service (QoS). Most prediction methods depend on building a mobility profile of the MN. A complete profile development of an MN is not feasible and hence 100% accuracy is impossible. The prediction accuracy depends on the regularity of an MN and hence MNs that show regularity in movement can be predicted with better accuracy. Nevertheless, regular MNs can sometimes behave unpredictably. Recently, various mobility prediction methods were proposed for MANET. These methods differ in the parameters they predict, the information they use for prediction, and the purpose for which the prediction results are used.

The authors of [27] presented and tested a sequential learning algorithm for the short-term prediction of human mobility. Constant order Markov model technique is used for the prediction. This algorithm predicted the human mobility that used large data sets of sequences. Accuracy of prediction is high; however, mobility prediction cannot be computed if the mobile history data is not available.

An Enhanced Localization Solution (ELS) was proposed in [28]. ELS is an innovative self-adaptive solution that combines standard location tracking techniques as well as human mobility modelling and machine learning techniques. The results showed that ELS worked well for different nodes' behaviours, and the location prediction could be used in more than 50% of the cases, with low error and with an effective advantage in terms of power consumption.

Suraj *et al.* in [29] proposed a new approach to mobility prediction. It is completely based on lightweight genetic algorithms to improve the MANET routing algorithms. The architecture of this genetic predictor did not include all genetic operations and was modified to reach a termination condition without a large number of iterations. This technique opened up new possibilities in the field of mobility prediction for MNs in an ad hoc network and can lead to better QoS than probability based techniques. However, heavy computational power is needed for prediction and more memory is needed for storage in an MN.

In [30], a new method is proposed to dynamically predict the future position of a pedestrian based on the real trajectory data. The proposed method performed well on different types of trajectories. This method depends on pedestrian tracked data. The performance is better for high mobility scenarios, where the error vary from three meters to less than one meter.

Authors of [31] proposed a Bayesian model to predict the mobility of a node in MANETs to help a routing protocol to avoid broadcasting request messages from a high mobility node/region based on the prediction result. This model did not relay on the information from GPS. The packet delivery ratio of this method improved up to 46.32% at the maximum speed of 30 m/s in the density of 200 nodes/km^2.

A mobility prediction method and modelling technique based on the Markov model were proposed in [32], the area in which the mobiles move is geographically partitioned into cells to form the Markov chain. To enhance the accuracy rate (i.e. to determine the mobility behaviour in a shorter time), the number of states must be increased. That is, the geographical area must be divided into a larger number of sub-regions of a smaller size.

In [33], a new solution for the prediction of the future node locations in a MANET is proposed using a neural learning machine-based model. This solution is based on architectures of the standard MultiLayer Perceptron (MLP) and the Extreme Learning Machine (ELM). This model outperforms existing mobility prediction algorithms and achieves accuracy scores higher by an order of magnitude. This accuracy allows the proposed mobility predictor to improve the overall quality of service in MANETs. The proposed model can predict routing tables which would reduce the data exchange in MANETs. Hence the life of the node battery is extended.

Proposed prediction method in [34] predicted future location of user by considering online posts which have been tagged with geological coordinates collected through the GPS interface of smart phones. However, the prediction accuracy is low because of limited amount of information.

An ad hoc on-demand distance vector routing algorithm, which was proposed in [35], took into account node mobility. This algorithm estimated life time of the link based on the measured mobility to choose the best route according to the link durations. The proposed algorithm also implemented the mobility estimates for route maintenance while the original ad hoc on-demand distance vector routing algorithm uses a fixed value as duration for route maintenance. This algorithm significantly reduced the number of overhead messages for route discovery and route maintenance. Thus the performance (such as packet delivery rate, end to end delay) is improved.

3 MOBILITY MODELS FOR MANETS

A Mobile ad hoc network is self-organized i.e., there is no pre-existing infrastructure. Hence it independently determines its own configuration parameters such as position identification, power control, routing and addressing. The free movement of nodes raises several new issues that did not have to be solved in fixed network infrastructures. This mobility is a very significant attribute in MANETs. The MN may follow different mobility models that affect the network performance. Therefore, the mobility model should be able to mimic the real movement of MNs. A mobility model is mainly designed to describe the actual movement pattern of MNs, their geographic position, speed, etc. It determines the location of nodes in the topology at any given instant, which in turn directly impacts the network connectivity. The mobility models are mathematical algorithms that try to fit the behaviour of real movement patterns. However, the drawback of these algorithms is that the mobility model reflects the real behaviour of MNs only to a certain degree. A good overview of mobility models can be found in [13] or in [36].

As shown in Fig. 3.1, there are two categories of mobility models: entity mobility models and group mobility models [13], [37]-[39]. The entity mobility models represent MNs whose movements are independent of each other. While the group mobility models represent MNs whose movements are dependent on each other.

In this part, some mobility models that have been proposed for an ad hoc network are presented:

- Random Walk Mobility model.
- Random Waypoint Mobility model.
- Random Direction Mobility model.
- Modified Random Direction Mobility model.
- Map-based Mobility model.
- Reference Point Group Mobility model

The Random Walk Mobility model and the Random Waypoint Mobility model are the two most used mobility models in ad hoc networks. Therefore, these mobility models will be used in the simulation and their improvements proposed within this thesis.

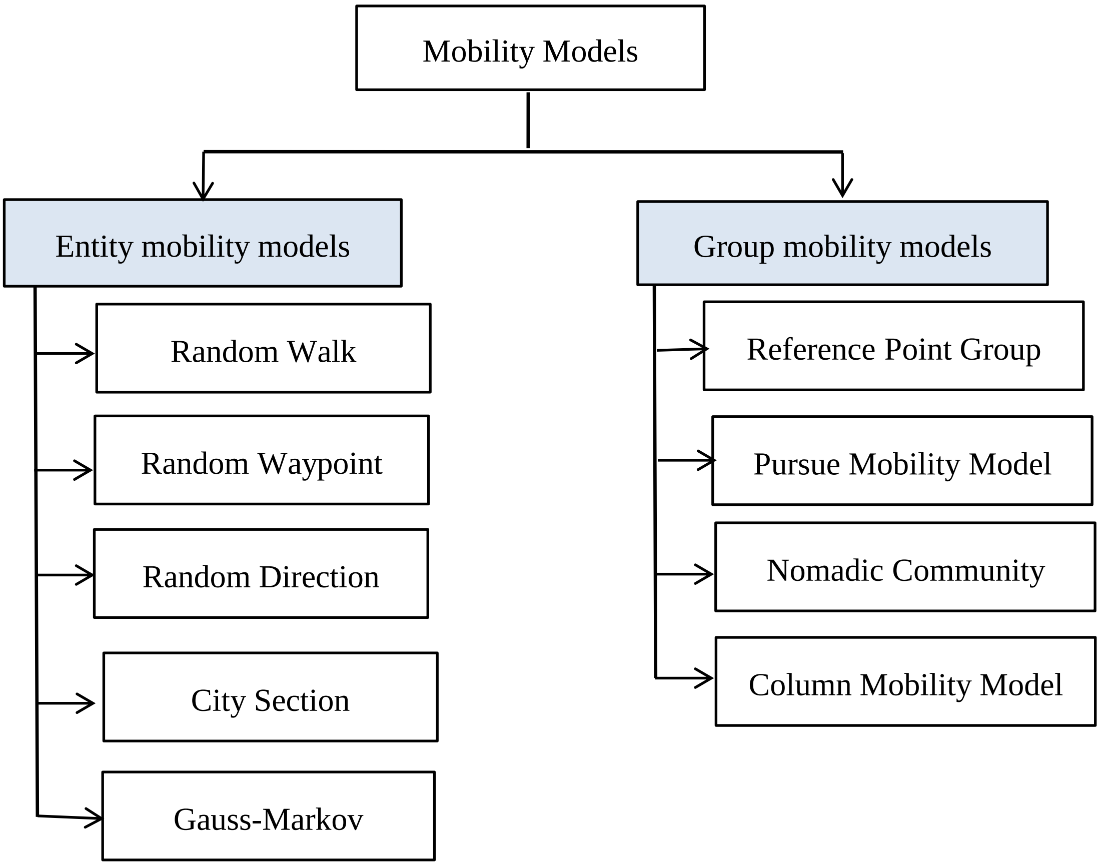


Fig. 3.1 Summary of mobility models for MANETs

3.1 Random Walk Mobility Model

The Random Walk Mobility model (RWM) was first represented mathematically by Einstein in 1926 [40]. In this mobility model, an MN can move randomly, which means that the direction and speed of moving are selected randomly with uniformly distributed. Both the speed and the direction are limited by a pre-defined range, $[speed_{\min}, speed_{\max}]$ and $[0; 2\pi]$ respectively. Each movement in the RWM model occurs in either a constant time period t or at a constant distance travelled d, at the end of which a new direction and speed are calculated. If a node reaches the border of the area, it selects a new direction [40]. If the node moves according to these rules and reaches the boundary of simulation field, the node moves away from the boundary by an angle$(\pi - \theta(t))$, where $\theta(t)$ is the angle during the time interval t. Many derivatives of the RWM model have been studied including the one-dimensional, two-dimensional, three-dimensional, and d-dimensional walk [41]. Fig. 3.2 shows an example of the movement pattern of an MN based on the Random Walk Mobility model after 30 steps.

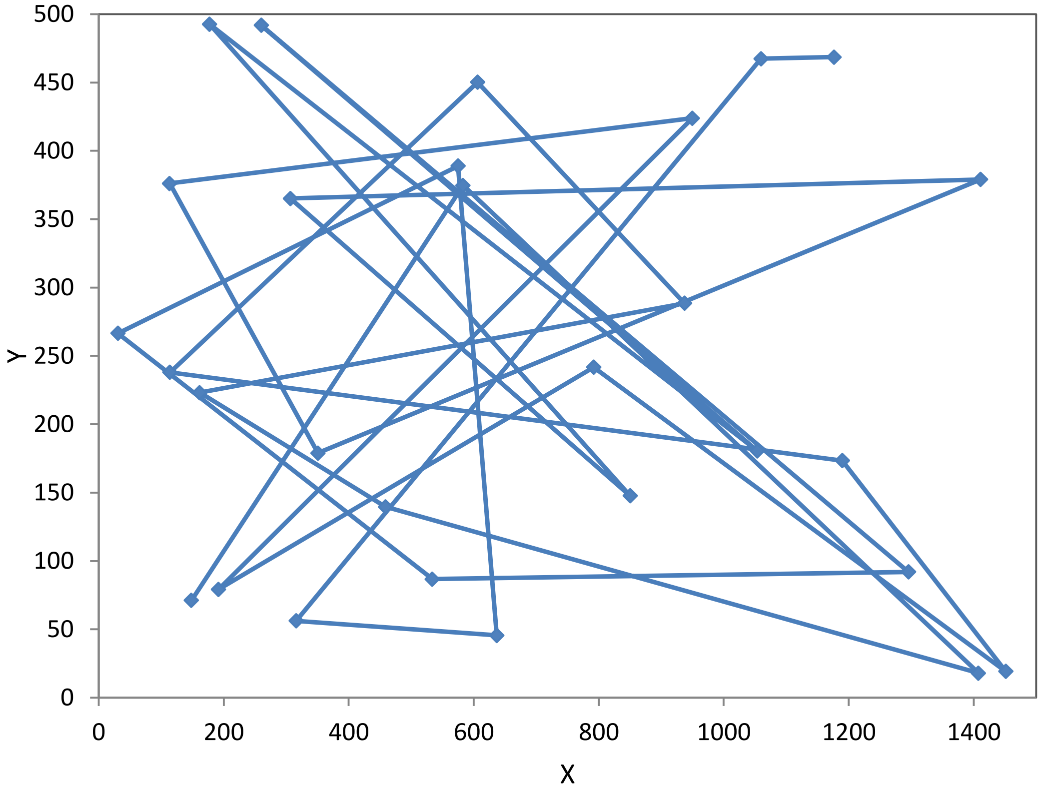


Fig. 3.2 Movement pattern of an MN using the 2-D Random Walk Mobility model

3.2 Random Waypoint Mobility Model

The Random Waypoint Mobility model (RWP) is one of the most popular mobility models used to evaluate the mobility prediction in MANET. With this model, the node moves from its current position to a new one by randomly selecting destination and speed. The distribution of speed is uniform within a range $[speed_{min}, speed_{max}]$. When the node reaches its destination, it waits for a certain time (pause time) and then it again selects a random destination and a new speed. The path between the current position and the destination is considered to be straight line.

Note that the Random Waypoint Mobility model is equivalent to the Random Walk Mobility model if the pause time is zero and $[speed_{min}, speed_{max}]$ of Random Waypoint equal to $[speed_{min}, speed_{max}]$ of Random Walk Mobility model [13], [42], [43]. An example of the Random Waypoint movement of an MN after 30 steps is shown in Fig. 3.3. This model and its derivatives are widely used. Some research efforts described inherent deficiencies such as non-stationarity. Stationarity means that statistical properties (mean, variance and the probability distributions) do not change in time during simulation [44]. Where in RWP model, the average speed of the nodes is found to consistently decrease over a large interval of time, speed actually decays will cause node speed to become zero [47]. With increasing simulation time, the speed of the nodes will have an exponential distribution. Hence speed distribution is not uniform [45], [46]. Another deficiency is nonuniform spatial distribution (border effect) which means that MNs travel through the centre of the simulation area with a greater probability than any other area. At the end of the simulation, the density of nodes is much higher at the centre of the simulation area and almost non-existent at the boundaries. However, it is sometimes still widely used since the decaying effects are only observed during long simulations. Long simulation using RWP model under the above-described conditions was performed to show the decrease in average instantaneous speed. The model is initialized with a speed distribution [0, 40]. The average speed of the nodes at the end of the simulation is shown in Fig. 3.4.

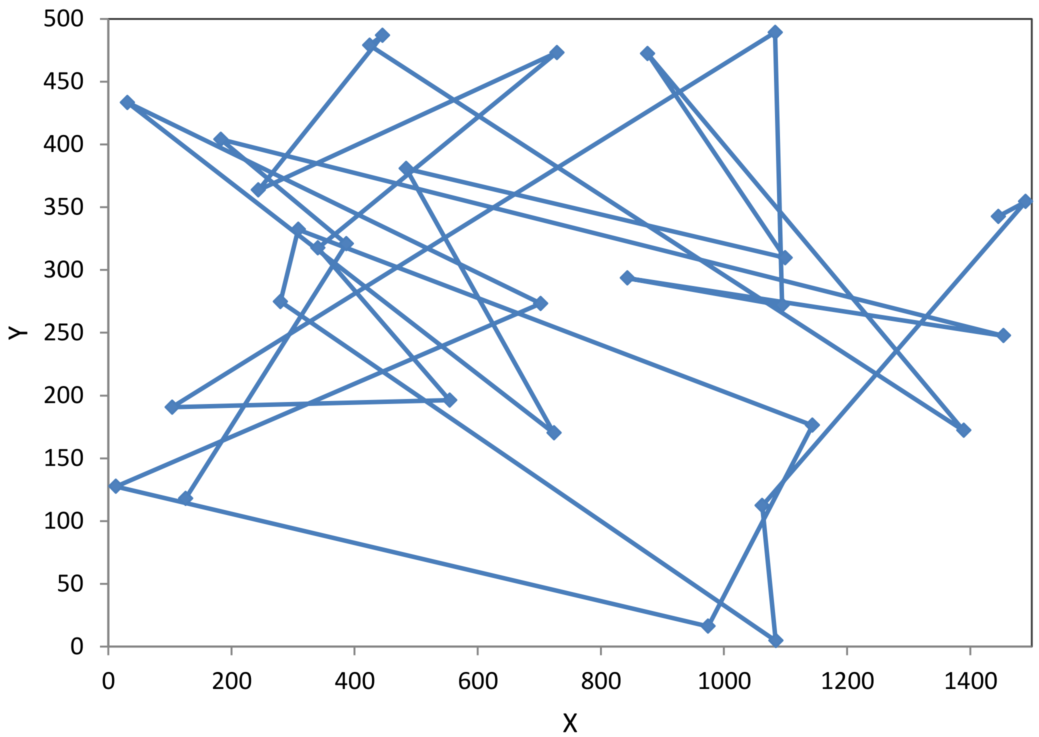


Fig. 3.3 Movement pattern of an MN using the RWP Mobility model

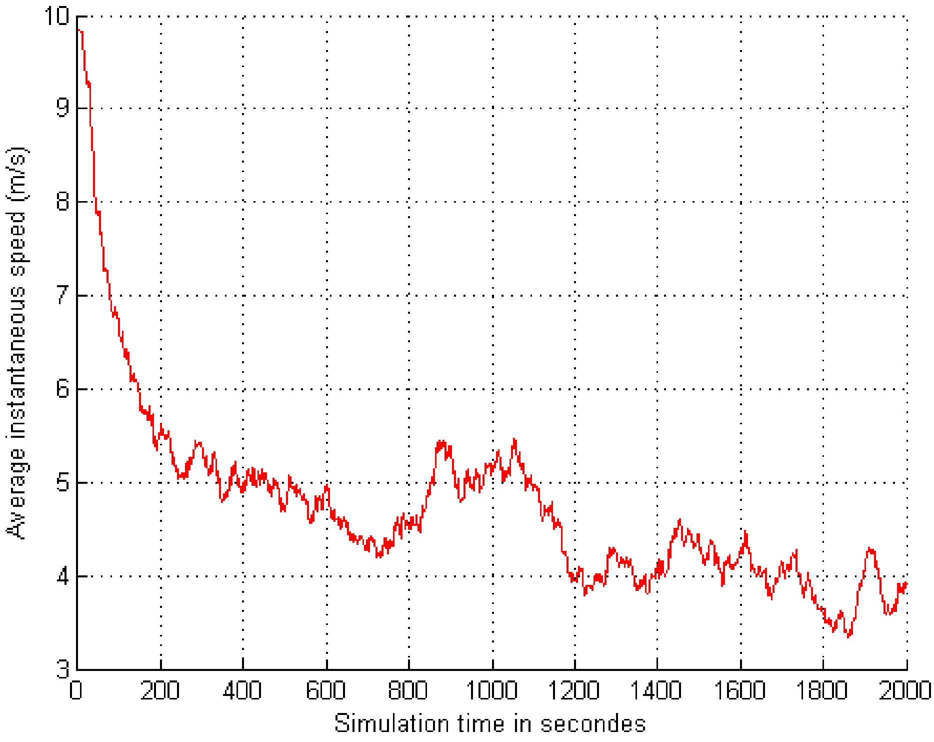


Fig. 3.4 Average instantaneous speed in meters/second for RWP model [47]

3.3 Random Direction Mobility Model

Similar to RWP model, the Random Direction Mobility (RDM) model also considers user movement along straight line paths, constant speeds and pauses in between motion positions. This model was set to overcome the non-uniform spatial distribution problem in RWP model [48]-[50]. Instead of selecting a random destination within the simulation area, in the random direction model, a node randomly and uniformly selects a direction by which to move along until it reaches the boundary. Once the boundary is reached, the node pauses for a certain period of time, selects a new direction in the interval $[0, 2.\pi]$, and continues the process. The nodes are uniformly distributed within the simulation area by this model. This model is used for simulation studies of mobile cellular networks [48]. Fig. 3.5 shows an example path of an MN, which begins at the centre of the simulation area using the RDM Model.

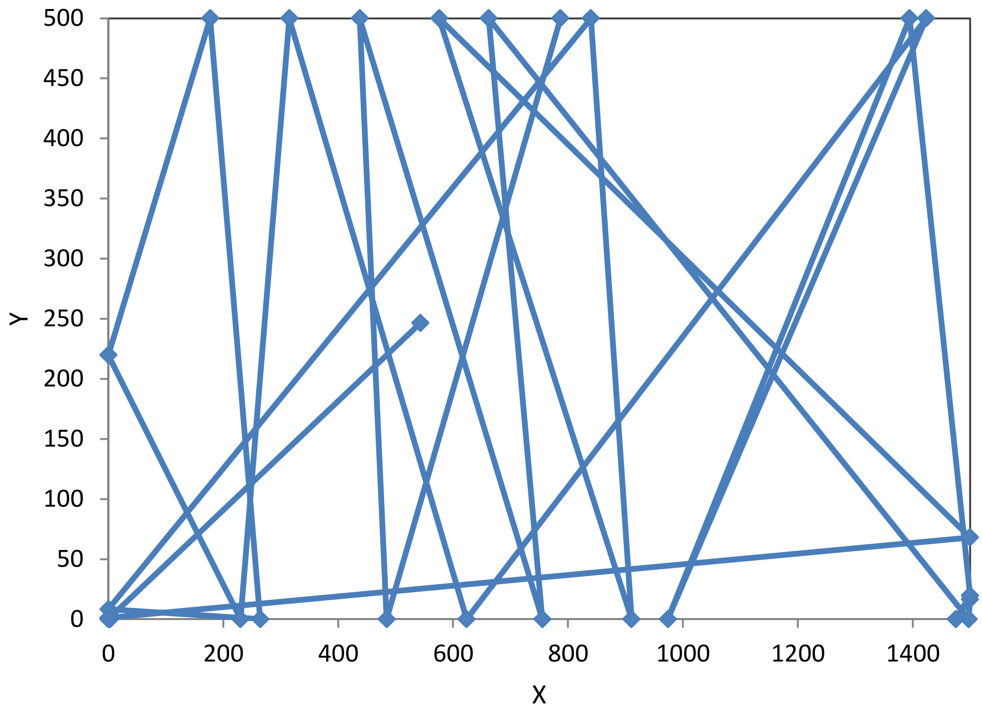


Fig. 3.5 Movement pattern of an MN using the RDM model

3.4 Modified Random Direction Mobility

Another variation of the Random Direction Mobility model is the Modified Random Direction Mobility model (MRD). Similar to RDM, an MN randomly and uniformly selects a direction by which to move. However, it chooses a random destination point on the projected line between current point and boundary.

In comparison to the RWP, the difference is that the distance an MN travels is uniformly distributed in [0, distance-to-border] while each movement in the RWP model occurs in either a constant time period t or at a constant distance travelled d.

3.5 Map-Based Mobility Modelling

Map-Based Movement (MBM) allows the nodes to use random paths based on a map which is defined in dependence on the network environment (e.g., the environment may be a map of a city) [14].

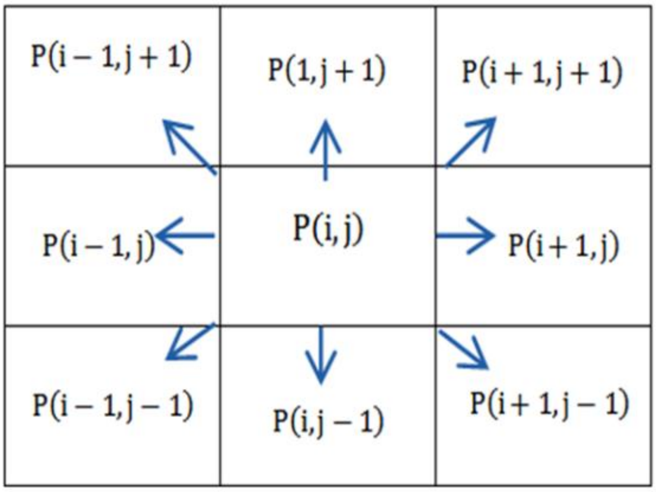


Fig. 3.6 Matrix of movement probability

In other words, nodes move randomly but always follow the paths defined by map data. In MBM each node is represented by a matrix of movement probability in eight main neighbouring directions based on the determinants of the given environment, Fig. 3.6, and the centre of matrix $P(i,j)$ represents the probability of the node staying in the same position [14].

In order to get a softer path and reduce the meanders, it’s possible to use an additional matrix of 3x3 in size to determine additional weights based on the direction of the previous movement. This matrix is called the direction matrix. For example, Fig. 3.7a prefers a horizontal movement with the possibility of turning up or down. If the node moves in the upright direction, this direction in the next step has the highest probability. It means that the direction matrix will rotate according to the previous step as shown in Fig. 3.7b.

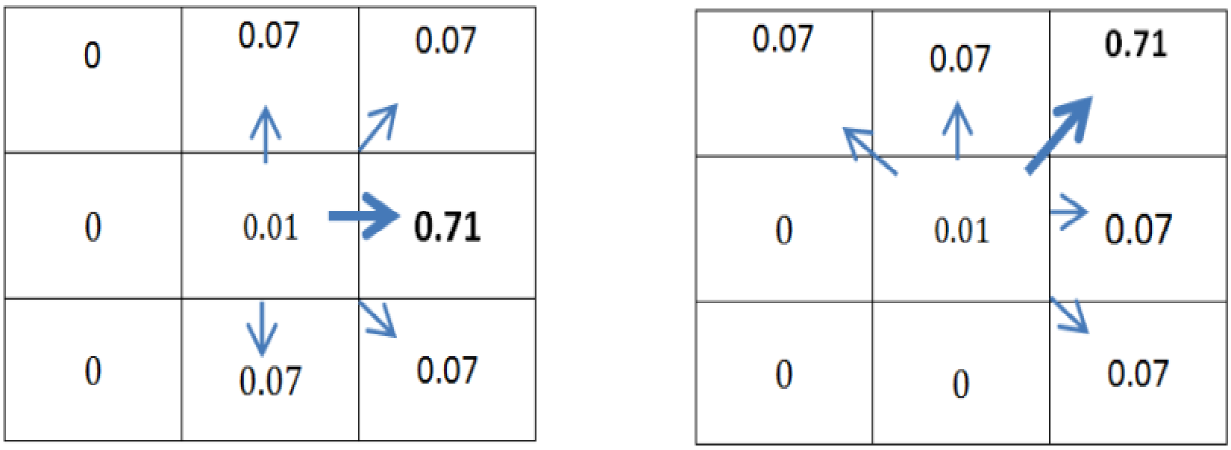


Fig. 3.7 Direction matrix: a) initial state b) rotated matrix

When the MN has to move towards a hotspot with known coordinates, it is possible to use a global direction matrix, which prefers a global direction from an entire virtual environment. This matrix is only rotated when the current hotspot is achieved and the node will continue moving toward the next hotspot.

3.6 Reference Point Group Mobility Model

Group mobility models are proposed for many situations where a group of members work together in a cooperative way to accomplish a common purpose such as an avalanche rescue. Reference Point Group Mobility Model (RPGM) represents the group mobility model which was first introduced in [51]. In this mobility model, each group has a logical centre. The centre's motion defines the entire group's motion behaviour, including location, speed, direction, acceleration, etc. Thus, each group is composed of one reference point and a number of members. The movement of reference point defines the motion of reference point itself, and the general motion trend of the whole group. For each node, mobility is assigned with a reference point that follows the group movement. In each group, nodes are uniformly distributed within a radius r from the reference point. The reference point scheme allows independent random motion behaviour for each node, in addition to the group motion. Thus this mobility model makes two vectors: group mobility vector, which is shared by all members of the same group and internal mobility vector, which represents the relative mobility of a node inside the group. The vector sum of the two mobility vectors decides the mobility of the node [52], [53].

3.7 Summary

There are several mobility models that have been proposed to fit certain specific mobility scenarios [13]. This chapter presented some mobility models that have been proposed for MANETs: Random Walk Mobility model, Random Waypoint Mobility model, Random Direction Mobility model, Modified Random Direction Mobility model, Map-based Mobility model and Reference Point Group Mobility model. The Random Walk Mobility model and the Random Waypoint Mobility model are the two most used mobility models in ad hoc networks. Therefore, these mobility models are later assumed, discussed and analysed by simulations as will be shown in the Chapter 7, whereas the mobility prediction is utilized.

4 MOBILITY PREDICTION

As already mentioned above, the free movement of nodes raises several new issues that did not have to be solved in fixed network infrastructures. One way to address the effects of node mobility is estimating the movement of the MNs before the actual movement (mobility prediction). Mobility prediction methods are commonly based on historical movement patterns of the MN. Predictions can be for the next step of MN movement or for a whole sequence of movements. Almost all prediction methods depend on the fact that node movements are not completely random [25].

4.1 Existing Methods of Mobility Prediction

Mobility prediction has received a lot of attention from the research community. A lot of research has been conducted to improve the accuracy of the mobility prediction methods or to enhance the performance of the mobility based networking protocols [54]-[62]. This section presents some mobility prediction methods that have been proposed for an ad hoc network protocol. Fig. 4.1 shows the most frequently reported applications of the mobility prediction in MANET which can be summarized as estimation of the link availability time, path reliability, route duration, network partitioning prediction, and routing enhancement [23], [63], [64].

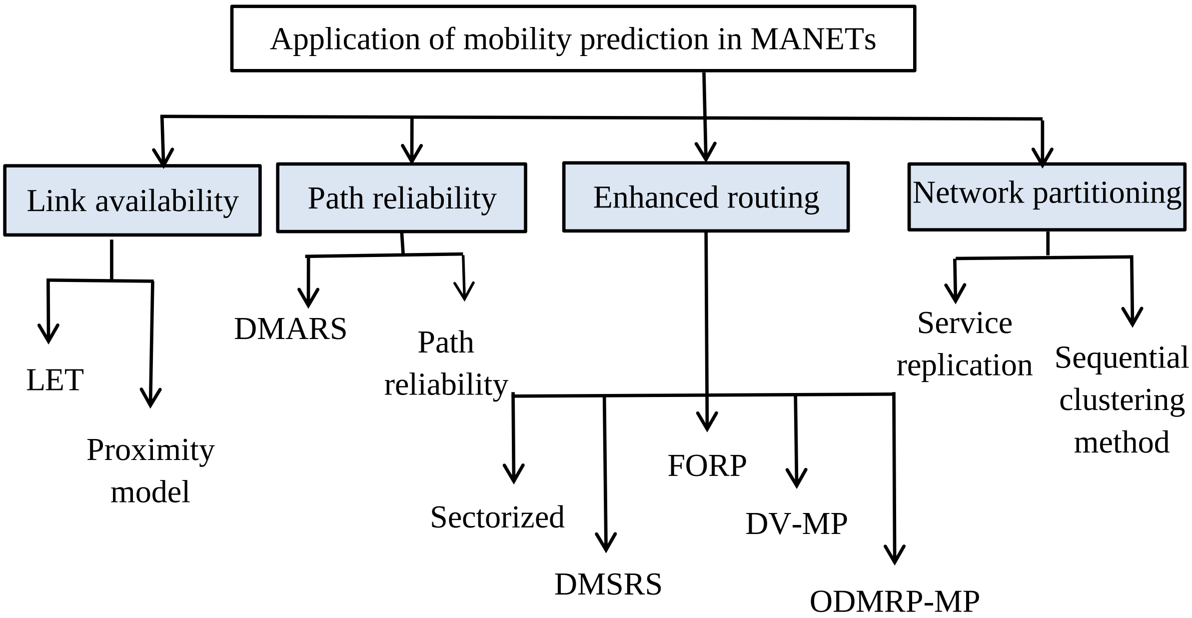


Fig. 4.1 Applications of mobility prediction

The mobility prediction is widely used to estimate the link availability time in a wireless mobile network. A link is available between every two nodes as long as they are within the transmission range of one another. A prediction mechanism for Link Expiration Time (LET) between two MNs has been studied in [65]-[69] to improve routing protocols for MANETs. LET is a period of time that two neighbouring nodes will remain connected by using mobility parameters of these two nodes (e.g. speed, direction and radio propagation range).

By exploiting the fact that an MN moves by a non-random movement pattern in real situations, it is possible to estimate the future topology of the network. The Global Positioning System (GPS) and the signal strength methods both use physically measured parameters to predict the link availability time. A prediction algorithm using GPS was proposed in [68] to determine the LET. It is assumed that all MNs in the network have their clock synchronized (e.g. by using the GPS clock itself). Assume two nodes i and j are within the transmission range of each other, Fig. 4.2. If the nodes i and j at locations (x_i, y_i) and (x_j, y_j) are moving at speed v_i and v_j respectively, then the link expiration time is calculated as follows:

$$LET = \frac{-(ab + cd) + \sqrt{(a^2 + c^2)r^2 - (ad - bc)^2}}{a^2 + c^2}, \qquad (1)$$

where,

$$a = v_i cos\theta_i - v_j cos\theta_j, \qquad (2)$$

$$b = x_i - x_j, \qquad (3)$$

$$c = v_i sin\theta_i - v_j sin\theta_j, \qquad (4)$$

$$d = y_i - y_j, \qquad (5)$$

$$r^2 = b^2 + d^2, \qquad (6)$$

whereas θ_i and θ_j are the moving directions of nodes i and j, respectively.

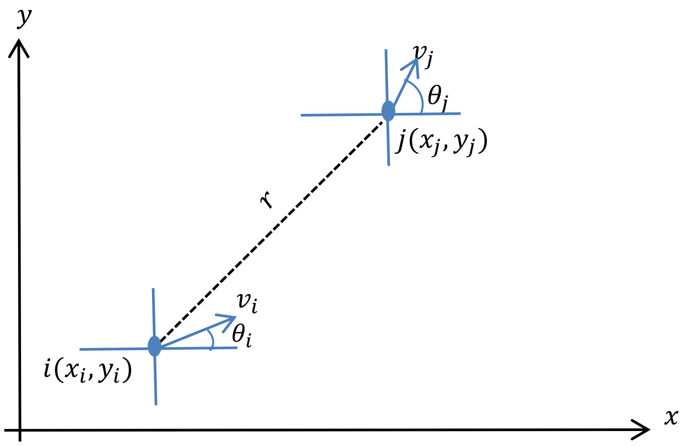


Fig. 4.2 Link expiration time between pair of nodes

The mobility information of each MN can be observed from GPS. If the nodes i and j are traveling at the same speed and the same direction ($v_i = v_j, \theta_i = \theta_j$), the link expiration time is infinite. After predicting the LETs of all links of a route, the minimum value of LETs is the Route Expiration Time (RET). This allows rebuilding route before route failure. However, this method is not suitable in the case of sudden changes in the direction and speed. It is only used for simple mobility model. A prediction algorithm using signal strength was proposed to determine the time when two nodes are moving out of the radio transmission range. This algorithm only uses the signal power of receiving the packet and does not depend on any add-on device.

A path reliability is another metric for path selection [70] because LET alone cannot exactly judge link availability. If D_t is the expiration time of an active link between two nodes i and j at time t_0, the availability of this link $L(D_t)$ is devided into two parts:

- availability of the link when the speeds and moving direction of tow nodes i and j are constant over the time period of $(t_0 + D_t)$,
- availability of the link for the other cases [70].

A prediction mechanism for a proximity has been presented in [71], [72]. The proximity model has been presented to quantify the future proximity of adjacent nodes and provides a quantitative metric that reflects the future stability of a given link. This model is used in MANETs to minimize the requirement for precise mobility information.

Network partition occurs because of the group mobility behavior of the MNs in MANETs, where the MNs belonging to the same mobility group show similar mobility pattern, while the nodes of different groups show varied mobility patterns [73]. The system proposed in [74] is for the detection and recovery of network partitioning using additional nodes. This system has a reactive behavior, which means that the system only becomes active when a communication problem occurs, for example, when a link breaks down.

The Markov model has been used in algorithms for prediction [75]. Several mobility prediction methods and modeling techniques based on the Markov model were proposed in [76]-[80]. A Mixed Markov chain Model (MMM) has been proposed for next location prediction in [81]. This approach considers that standard Markov Models (MM) and Hidden Markov Models (HMM) are not generic enough to encompass all types of mobility. Therefore, the concept of MMM was proposed as an intermediate model between individual and generic models. The prediction of the next location is based on a Markov model belonging to a group of individuals with similar mobility behaviour. This approach clusters individuals into groups based on their mobility traces and then generates a specific Markov model for each group. The prediction accuracy of MMM is better than those of a simple Markov model and a hidden Markov model. A mobility prediction-based clustering (MPBC) scheme was proposed by Ni *et al.* [82] for wireless mobile ad hoc networks. A node may change the associated cluster head (CH) several times during the lifetime of its connection. The proposed clustering algorithm includes an initial clustering stage and a cluster maintaining stage. The Doppler shifts associated with periodically exchanged Hello packets between neighbouring nodes are used to predict their relative speeds, and the estimation speeds are used to predict the remaining time that a cluster member may stay in the transmission range of its cluster head.

In general, the GPS-based prediction method is one of the good solutions to estimate the network mobility [31], [84]. The authors of [84] modified Q-Routing algorithm by Reinforcement Learning (RL) techniques. The aim of this method is to increase network nodes' lifetime by proper energy-efficient policies and achieving fault-tolerance, through a rapid path-handover and mobility prediction.

The authors of [85] proposed a new approach to predict user mobility in the absence of mobility history by using Short Message Service (SMS) and instantaneous Geological coordinates. This work was applied to predict the mobility of medical rescue vehicles and social security systems.

In [83], I implemented a neural network to predict the future movement of MNs in ad hoc networks. This method consists of three-layer feed-forward network. The backpropagation algorithm was used to learn the neural network. The mobility prediction allows finding the more stable links in a mobile ad hoc network. The training and testing neural network were done by the input and output patterns representing locations of mobile ad hoc node which moves according to RWM Mobility model. I evaluated the neural network by presenting the new input patterns to this network to produce the predictions. It was found out that feed-forward neural network works well for prediction the next position of a MN in MANET.

In [154], I proposed a prediction algorithm of the future movement of nodes based on the information of current network status as will be seen in section 6.1. It determines the behavior of the mobility prediction method using data collected from the node which moves according to random walk mobility model. This article studied the impact of the mobility prediction method on mobile nodes' parameters such as delay, throughput and packet delivery ratio as will be seen in section 7.1.

4.2 Quality of Service

The QoS (Quality of Service) is an integral part of MANETs. The topology of MANET is changing due to dynamic joining and leaving of the wireless nodes. Requirements on QoS rise from the application layer in the form of ensuring the required values of some network parameters such as throughput, delay or jitter. Most of the modern multimedia applications strictly require QoS methods [86], [87]. The goal of using QoS is deterministic behaviour of the network, which is achieved when the information transmitted through the network is delivered on time. In this case, the required quality and utilization of network sources are optimized [88].

The main issues associated with providing QoS in MANET are:

- **Unreliable communication channel** - the bit errors are the main problem of unreliable wireless channel. This is due to high interference, thermal noise or multipath fading effect. Using wireless environment for MANETs may lead to leakage of information.
- **Maintenance of the route** - the dynamic nature of the network topology and also the changing behaviour of the communication medium makes the maintenance of network very difficult. The established routing path may be broken during the process of data transfer. Hence it is necessary to maintain and

update the routing paths with minimal overhead and delay. That requires the reservation of resources at the intermediate nodes.

- **Mobility of the nodes** - the nodes in MANET are considered as MNs that move independently and randomly. Therefore, the topology information has to be updated frequently [89].
- **Limited power supply** - providing the QoS consumes more power due to overhead from MNs. This is reflected in a faster discharging of the battery.
- **Lack of centralized control** - the MNs in ad hoc networks can join or leave the network. The network is then set up spontaneously. There is no centralized control. That leads to increased algorithm's overhead and complexity.
- **Collisions** - the nodes must communicate between themselves on common channel. This generates the problems of interference and channel contention (for example the hidden node problem). For peer-to-peer data communication it is possible to use TDMA (Time Division Multiple Access) systems where each MN may transmit at a predefined time [90]. Other options are to use a different frequency band or spreading code CDMA (Code Division Multiple Access) for each transmitter.
- **Security** - it is an important part of MANETs. The physical medium of communication is insecure. There is a need to design security-aware routing algorithms for these networks [91].

Today, most user equipment (mobile phones, tablets, net-books, etc) supports multiple Radio Access Technologies (RATs), such as Long Term Evolution (LTE) and WiFi. One of the most engaging challenges for mobile operators is the question how to manage the data traffic in mobile networks which is increasing exponentially; mainly due to the growing popularity of applications for mobile devices. Mobile data offloading represents the idea of cost-efficient technology to release the overloaded parts of RAN in cellular networks. This emerging solution introduces the concept of offloading the mobile data from primary cellular communication technology to the IEEE 802.11 infrastructure with aim to gain extra capacity (higher throughput) and improve the overall network performance and user experience. In [127], we addressed two the most discussed solutions for offloading between the LTE cellular network and WiFi when the performance needs/requirements exceed the threshold for providing the services via LTE network under an agreed-upon QoS. In detail, the implementation of SNR (Signal-to-Noise Ratio) threshold based handover and network throughput based handover solutions following the 3GPP standard for Access Network Discovery and Selection Function framework for WiFi offloading in simulation environment NS-3 is proposed. We have shown that user equipment is connected to WiFi access point (AP) at the distance of 117 meters at 1.68 Mbit/s throughput, beyond this range the user equipment is switched to the LTE network at 8.5 Mbit/s throughput for 3 MHz channel bandwidth and 16.08 Mbit/s throughput for 5 MHz channel.

4.3 Routing Protocols for MANETs

MANET routing algorithm inherited from traditional algorithms fails to consider ad hoc network characteristics like mobility and resource constraints. Therefore, it is subject to inefficiency. Since a mobile ad hoc network consists of wireless hosts that may move often and hence may cause links to be broken frequently. The protocol must adapt to frequent changes of network topology to set reliable paths, which remain valid as long as possible. The increased mobility of ad hoc nodes presents a challenging issue for protocol design. Various routing schemes have been proposed for ad hoc networks, they are classified on the basis of the way the network information is obtained in these routing protocols. These protocols are proactive (table driven), reactive (on demand) and hybrid routing protocols [8], [92]-[95]. Proactive routing protocols in wireless networks have a minimum end to end delay but they incur additional overhead due to maintaining up-to-date information [96]. While reactive routing protocols have lower overhead because routes are determined only on demand. End to end delay increases due to route discovery procedures. Hybrid routing protocols are using the best features of both the proactive and reactive routing protocols. The most popular protocols from these categories are:

- Optimized Link State Routing protocol (OLSR) [97].
- Ad hoc On-Demand Distance Vector (AODV) [98]-[100].
- Destination-Sequenced Distance-Vector routing (DSDV) [101].
- Dynamic Source Routing (DSR) [10], [102].
- Temporally Ordered Routing Algorithm (TORA) [103].
- Zone Routing Protocol (ZRP) [11].

A simplified view of the routing protocols in MANET is given in Fig. 4.3.

In [97], I described the quality of services support in MANET networks based on the modification of the OLSR routing protocol. The selected messages of this protocol were extended by new fields that were used for the QoS assurance. The experimental model of QoS support in MANET was implemented into the OPNET Modeler simulation environment.

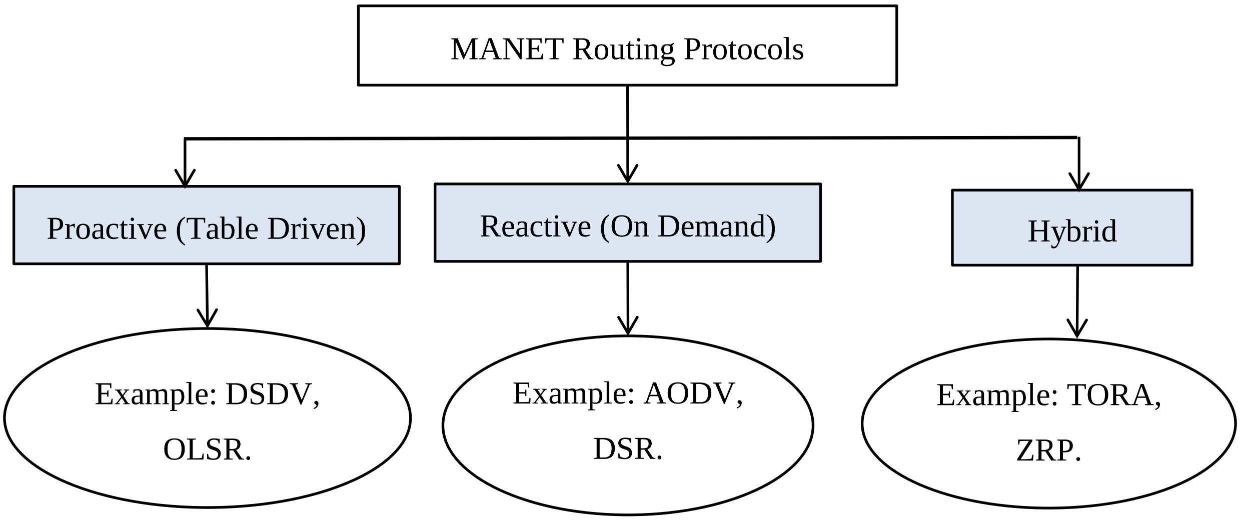


Fig. 4.3 Classification of MANET routing protocols

In [105], a new routing protocol is proposed that rank the available routes according to their path stability. Therefore, a signal strength based link prediction technique is proposed for demonstration. The proposed routing concept is implemented over AODV routing protocol.

In order to improve routing protocol performance, there are two schemes that utilize location information (for instance, obtained using the global positioning system). The first scheme is Location-Aided Routing (LAR) that uses location information obtained from the GPS to limit the search for a new path to a smaller request region of the ad hoc network [106]. Another location-based routing protocol is Distance Routing Effect Algorithm for Mobility (DREAM) [11], [107]. DREAM updates routing table periodically. Some routing protocols that have been proposed for an ad hoc network protocol are presented in this part.

Me with other authors in [104] implemented DSDV routing protocol into the simulation model of MANET, which was designed in the NS-3 environment. Further, the simulation focused on the analysis of the effect of QoS support on the most significant network parameters. For the purpose of performed analysis, two network services with different user priorities were defined: File Transfer Protocol (FTP) and Voice over IP (VoIP). The obtained results show that the deployment of QoS into the MANET brings the decrease of the end to end delay and jitter for VoIP service.

Table 4-1 End to end delay for VoIP service

User priority level	End to end delay [ms]
QoS=0	210
QoS = 6 (default)	160
QoS = 6 (improved)	143

Table 4-2 Jitter for VoIP service

User priority level	Jitter [ms]
QoS=0	5.2
QoS = 6 (default)	3.7
QoS = 6 (improved)	1.4

The end to end delay for VoIP was decreased from 210 ms to 160 ms in case of default QoS settings and to 143 ms in case of our improved algorithm as shown in Table 4-1.This change corresponds to the percentage decrease of 31.90% for improved QoS. The jitter for VoIP service decreases about 73.07% in the scenario with the modified QoS model, Table 4-2.

Due to the lower priority, the values of network parameters for the FTP service were increased. The end to end delay for the FTP application was increased from the value 262 ms to 319 ms. Another increase occurs when the improved QoS algorithm was implemented. This result verifies the theoretical presumptions when the FTP traffic has the lowest priority.

By the simple modification of the default QoS algorithm we were able to achieve better performance for the delay-sensitive network application [104]. The proprietary implementation of QoS support into the routing protocol DSDV was successful and brings expected results. The created model of MANET can be used as a basis for the related research.

4.3.1 Ad Hoc On-demand Distance Vector (AODV)

AODV routing protocol is reactive routing protocol and one of the most popular routing protocols designed to be used in mobile ad hoc networks [108]-[110]. AODV is developed based on the DSDV routing algorithm [99]. This protocol does not maintain permanent route table. Instead, routes are built by the source on demand. It has two phases: route-discovery and route-maintenance. The network remains inactive unless a connection is required by a node in the network. The source node that wants to know a route to a given destination broadcasts a Route REQuest (RREQ). The route-discovery phase is initiated as shown in Fig. 4.4, where node A wants to connect to node G. The adjacent node receives RREQ with the addresses of the source node and a destination node if it is same with the destination node's address. It sends a Route REPly (RREP) to the source node, otherwise, checking the routings in the route table that could reach the destination node, then sends RREP to the source node, or continues flooding the network by sending RREQ [98]. Fig. 4.5 presents the route determination from source A to the destination G. AODV protocol can maintain neighborhood information through broadcasting hello message periodically. A route is maintained only when it is used. If a source node moves, a new route discovery process is initiated. If intermediate nodes or the destination move, the next hop links break resulting in link failures. Therefore, routing tables are updated for the link failures and all active neighbours are informed by a Rout ERRor (RERR) message.

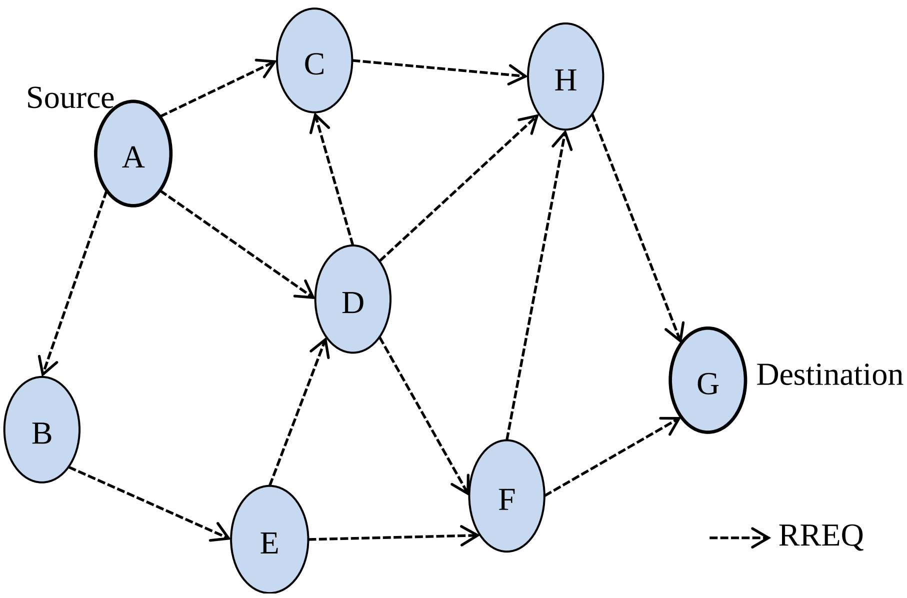


Fig. 4.4 Propagation of RREQ

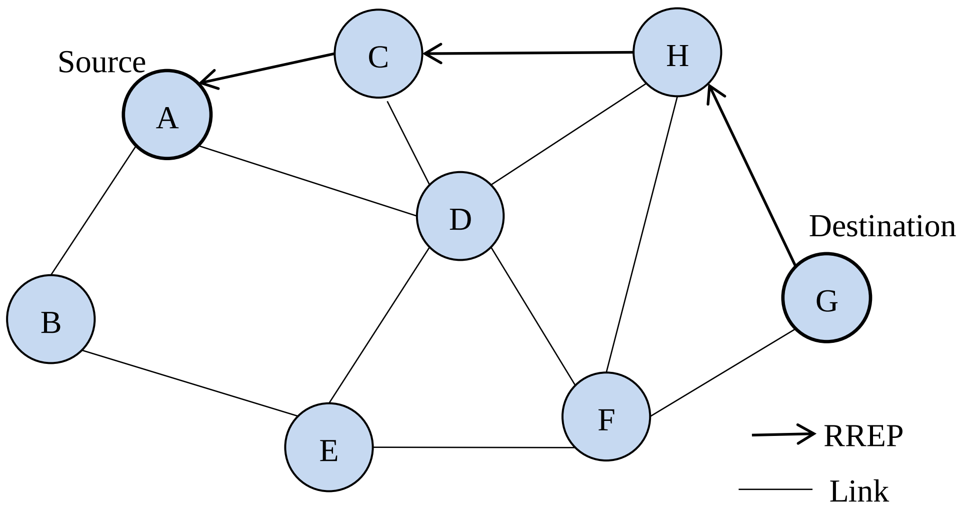


Fig. 4.5 Path of the RREP from the destination to the source

The problem of the AODV is that it selects a route with fixed lifetime instead of any reliable lifetime parameters and uses the route until a link failure occurs. The duration for route maintenance has fixed value while each link has different link duration as well as active route duration due to node mobility. However, AODV is a very useful and desired protocol in MANETs for its performance, because It was observed that AODV is very good in term of packet delivery ratio, throughput and end to end delay [108], [111].

4.3.2 On-Demand Multicast Routing Protocol for MANETs

The On-Demand Multicast Routing Protocol (ODMRP) has been proposed in [112], the source node creates and updates group membership and multicast routes on demand (non-periodically) to reduce the channel overhead. ODMRP depends on the concept of forwarding group. Forwarding group is a set of nodes that is in charge of forwarding multicast data on shortest paths between any member pairs. It has two cycles: a request cycle and a reply cycle. When a source wants to send packets, it periodically broadcasts the join request, which includes a member advertising packet and data, to the entire network to update the membership information and the routes. When a node receives a non-duplicate join request, it saves the source ID in its routing table and then it rebroadcasts the join request. When the desired multicast destination receives the join request, it builds a join table which is broadcasted to neighbouring nodes. When a node receives the join table it understands that it is on the route to the source in case that the next node ID of one of the entries is the same of its ID. This node is part of the forwarding group, thus it builds its own join table which is also propagated to neighbouring nodes after setting of Forwarding Group Flag (FG-Flag) [113]. Each part of the forwarding group propagates the join table until it reaches the multicast source via the shortest path. This method updates the routes from sources to destinations and builds the forwarding group. Fig. 4.6 shows the request and the reply cycles of ODMRP.

After building up the forwarding group and routes, a source can propagate packets to destinations by selected routes and forwarding groups. When a node receives the multicast data packet, it forwards the multicast data packet if the following two conditions are true: the multicast packet is not a duplicate and the setting of FG-Flag for the multicast group is still available. Nodes in the forwarding group will be non-forwarding nodes if they have not received join tables before their FG-Flag are expired. Thus, no explicit control packets need to join or leave the group. However, ODMRP discovers multicast routes only in the presence of data packets to be delivered to a multicast destination. Route discovery is based on the request and reply cycles where multicast route information is stored in all intermediate nodes on the multicast path. For example, there are three sources (S1, S2, S3) as shown in Fig. 4.7. These sources send multicast data packets to three destinations (D1, D2, D3) through forwarding group (N1, N2, N3).

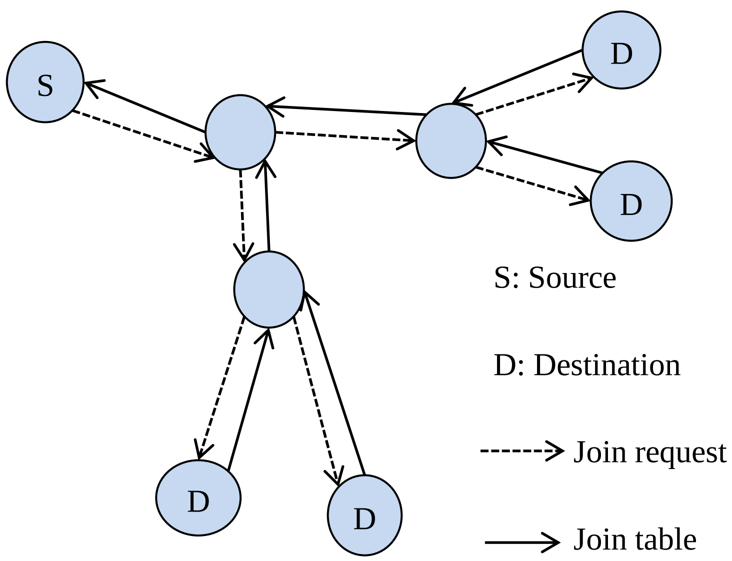


Fig. 4.6 The request and the reply cycles of ODMRP

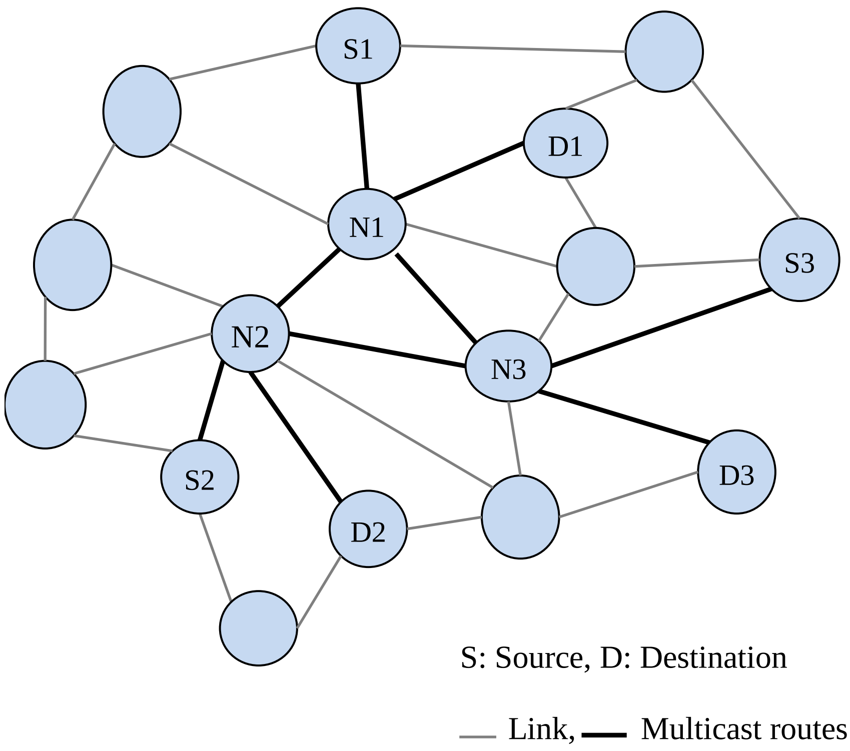
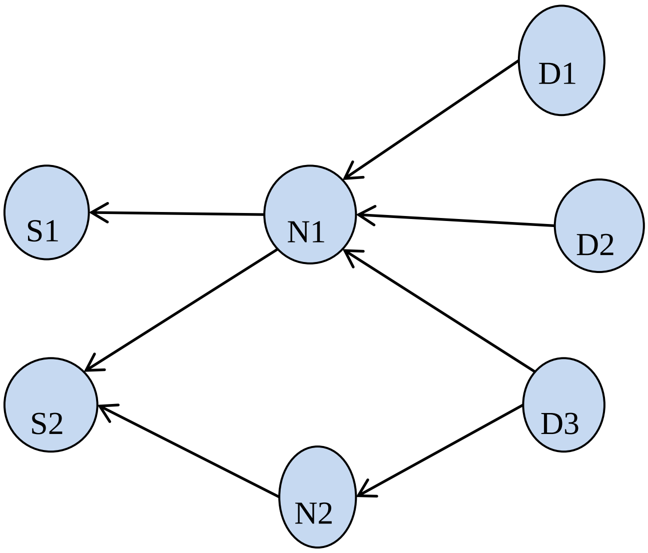


Fig. 4.7 An example of the ODMRP

The route from S1 to D3 is S1, N1, N3, D3. If the route between N1 and N3 is broken, in ODMRP, the route S1, N1, N2, N3, D3 will be in exchange for the route S1, N1, N3, D3. While in a tree configuration, the route from S1 to D3 is broken until the tree is reconfigured.

An example of join table forwarding is illustrated in Fig. 4.8. Multicast destinations D1 and D2 send their join tables to multicast sources S1 and S2 through N1. Multicast destination D3 sends its join table to S1 through N1 and to S2 through N2. When source S2 wants to send data to D3, it broadcasts the join request. When node N2 receives the join request, it saves the source ID in its routing table and then it rebroadcasts the join request. The desired destination D3 receives the join request; it builds a join table which is broadcasted to neighbouring nodes. When an intermediate node N2 receives the join table of D3, it sets the FG-Flag and builds its join table because the next node ID of one of the entries is the same of its ID.

Note that the channel overhead is decreased in case where node N1 broadcasts the join table once even though it receives three join tables from three destinations D1, D2, D3 because these multicast destinations have the same links to the source. ODMRP requires periodic flooding of join request to build and refresh routes which often cause contention and collisions. Thus it is important to find the optimal flooding interval.



Join table of D3	
Source	Next hop
S1	N1
S2	N2

Join table of N2	
Source	Next hop
S2	S2

Fig. 4.8 An example of a join table forwarding

4.3.3 Applying Mobility Prediction

The routing protocol must adapt to frequent changing network topologies to set reliable routes. The route establishment time should be short, with minimal overhead. Predicting the availability of a route in a network can significantly improve the capabilities of the existing routing protocols. The On-Demand Multicast Routing Protocol with Mobility Prediction (ODMRP-MP) was proposed in [114]. The period of time two neighbouring nodes will remain connected is predicted by using mobility parameters of these two nodes provided by GPS [67], [68]. With the predicted time of route disconnection, join request is only flooded when route breaks of on-going data sessions are imminent. A join request is tailed by the location, speed, and direction of the source. The source sets the minimum LET field to the maximum LET value. The neighbouring node, which receives a join request, predicts the link expiration time between itself and the previous hop. It rebroadcasts the join request, but this join request will include the minimum between this value and the MIN LET indicated by the receiving join request, this minimum value is RET. The node also updates the location and mobility information field according to its own information. If a forwarding group node receives multiple Join tables with different RET values from the same source to multiple destinations, it selects the minimum RET among them and sends its own join table with the chosen RET value attached. Then the source can build new routes by flooding a join request when route breaks of on-going data sessions are imminent (i.e. before the minimum RET approaches). The Destinations only send join tables after receiving join request. The selection of the minimum refresh interval and the maximum refresh interval should be adaptive to network situations (e.g., traffic type, traffic load, mobility pattern, mobility speed, channel capacity, etc.)

In this scheme, instead of using the minimum delay path, we can choose a route that is the most stable (i.e. the one with the largest RET). An example of route selection algorithm is presented in Fig. 4.9. Two routes are available from the source S to the destination D. Route 1 has a path of (S-A-B-C-D) and route 2 has a path of (S-A-E-C-D). The route expiration time of route 1 is 2 (min (3; 2; 4; 5)= 2) while that of route 2 is 1 (min (3; 5; 1; 5)=1). The destination selects the route with the maximum RET, and hence route 1 is selected. The multicast destination should wait for a convenient period of time to receive all possible routes and then the route which has maximum RET is selected to be included in a join table.

The route expiration time prediction could become incorrect because a node can accelerate, decelerate and change its direction while it is traveling. Also, the mobility information obtained from GPS may not always be accurate. If we suppose that there is no sudden change of direction and the mobility information obtained from GPS is accurate, the predicted route expiration time of a node could be always accurate.

ODMRP-MP is more effective compared to ODMRP because ODMPR-MP reconstructs the routes before topology changes (i.e. before the minimum RET approaches), thus most data are received without being lost.

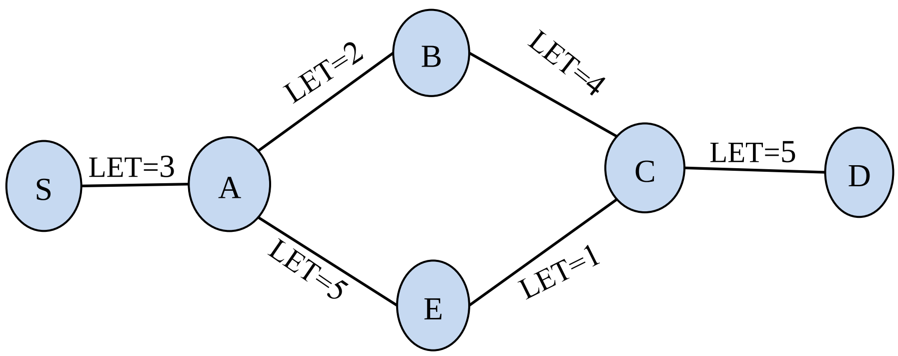


Fig. 4.9 Route selection example

4.4 Integration of Smart Antennas in Ad Hoc Network

Due to the need for increasing exchange and share data, users demand easy connectivity and fast network wherever they are. Recently, users are interested in interconnecting all their personal electronic devices together using MANET. As a result, the throughput capacity of ad hoc network can be limited because of interference. When a smart antenna system is integrated into such network, significant spatial reuse of the wireless channel can be achieved, decreasing the interference and thereby increasing the capacity of the network [115]-[117]. In this chapter, a short review of the challenges at medium access control caused by integration of smart antennas system in ad hoc networks is given.

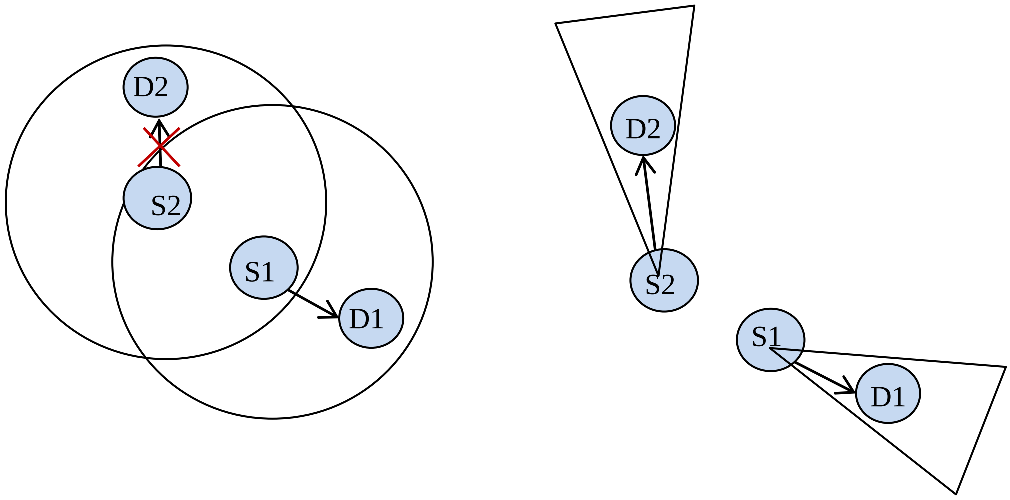
4.4.1 Smart Antenna

Typically, an ad hoc network uses omnidirectional antennas, which can transmit and receive signals equally from all directions. Since two nodes communicate using a given channel, all the other neighbouring nodes keep inactive. Thus the throughput capacity of an ad hoc network that uses such antennas is limited [118]. Smart antennas allow the energy to be transmitted or received in a particular direction as opposed to disseminating energy in all directions [119]. The capacity of MANETs is constrained by the interference between concurrent transmissions from neighbouring nodes.

The ability of smart antennas to direct their radiation energy toward the direction of the intended node while suppressing interference can significantly increase the throughput capacity compared to a network equipped with omnidirectional antennas because they allow the communication channel to be reused [120]. In other words, nodes with smart antennas focus only on the desired nodes and allow the neighbouring nodes to communicate, Fig. 4.10.b. In contrast nodes with omnidirectional antennas keep the neighbouring nodes inactive during their transmission as shown in Fig. 4.10.a [121].

There are two types of directional antennas systems: switched beam (sectorized) antenna systems and steerable beam system (adaptive).

In the switched beam (sectorized) antenna systems, multiple fixed beams are possible. These systems present a predetermined set of beams which can be selected as appropriate. For a switched beam antenna with K beams, the width of each beam is $2\pi/K$ radians. A directional transmission would then cover one of these k fixed sectors as illustrated in Fig. 4.11.



(a) (b)

Fig. 4.10 Throughput capacity of a network with a) omnidirectional antennas, b) a network with smart antennas

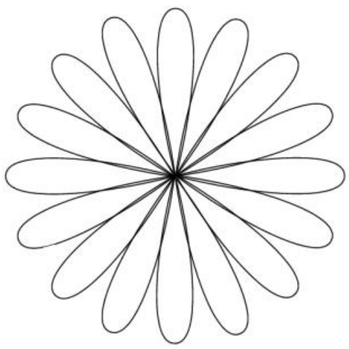


Fig. 4.11 Switched beam antenna system

The other type is steerable beam system (adaptive) in which the main lobe of the antenna can be focused toward the user of interest and nulls in the direction of the interference. Thus, if a node wants to communicate with its neighbour, it can adaptively steer its beam so as to point the main lobe towards that neighbour in a mobile scenario as well [122].

Gain and directivity are intimately related in antennas. Gain is a measure of the increase in power. The gain of a directional antenna is typically higher than that of an omnidirectional antenna. Directional antennas can have a larger directional range as compared to an omnidirectional antenna [123]. The gain of a directional antenna in a particular direction $\vec{d}$ is defined as:

$$G(\vec{d}) = \eta \frac{U(\vec{d})}{U_{ave}}, \tag{7}$$

where η is the efficiency of the antenna which accounts for losses. $U(\vec{d})$ is the energy in the direction $\vec{d}$. U_{ave} is the energy over all directions. The direction of peak gain is referred to the main lobe of the antenna.

4.4.2 MAC Protocol Using Directional Antennas

Suppose that there is a MANET of n MNs, where each MN has smart antennas with non-overlapping directions and all nodes use the same wireless channel. The antennas of a node cover all directions. The RTS and CTS messages are assumed to contain location information of both the sender and destination; this, in turn, helps transmit (or receive) the DATA and ACK messages directionally [7].

There are two approaches proposed for directional medium access control in [123]: Aggressive Collision Avoidance approach and Conservative Collision Avoidance approach. In the aggressive collision avoidance approach, a node can start a new transmission in spite of receiving an RTS or CTS sent by other nodes. The handshake is used only for ensuring that the destination is not busy sending or receiving. While in conservative collision avoidance approach a node is always prevented from transmitting when it receives an RTS or CTS. The performance evaluation shows that both approaches outperform the IEEE 802.11 MAC with omnidirectional communications. These approaches suffer from high collisions rate because of their dependence on omnidirectional mode for the transmission or reception of control packets in order to establish directional links. Another approach has been studied in [124]. This is called Destination Oriented Multiple Access (ROMA) which uses Multi-Beam Antenna Arrays (MBAA) as shown in Fig. 4.12. ROMA computes a link activation schedule in each time slot using two-hop topology information. Thereby significant improvement in network throughput and delay can be achieved. However, ROMA does not take into account nodes mobility.

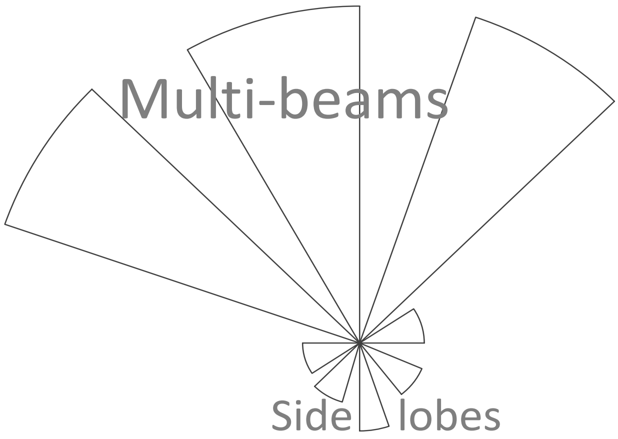


Fig. 4.12 The multi-beam Antenna array

In [125] Location and Mobility Aware (LMA) MAC protocol are developed for VANETs. The predictive location and mobility of the vehicles are adapted to provide robust communication links while using the directional beams. The LMA protocol predicts the transmission angle between the transmitter and destination. The predicted angle is not accurate because the moving angle of the destination can be changed during the data transmission thus causing data loss. This protocol can be enhanced by using the Directional Beacon (DB) mechanism which makes the MNs get the update of mobility information via DB after any change of one node moving angle or speed. If nodes move regularly, the predicted angle can be more accurate. However, new challenges appeared with the directional communications, such as deafness and hidden terminal problems.

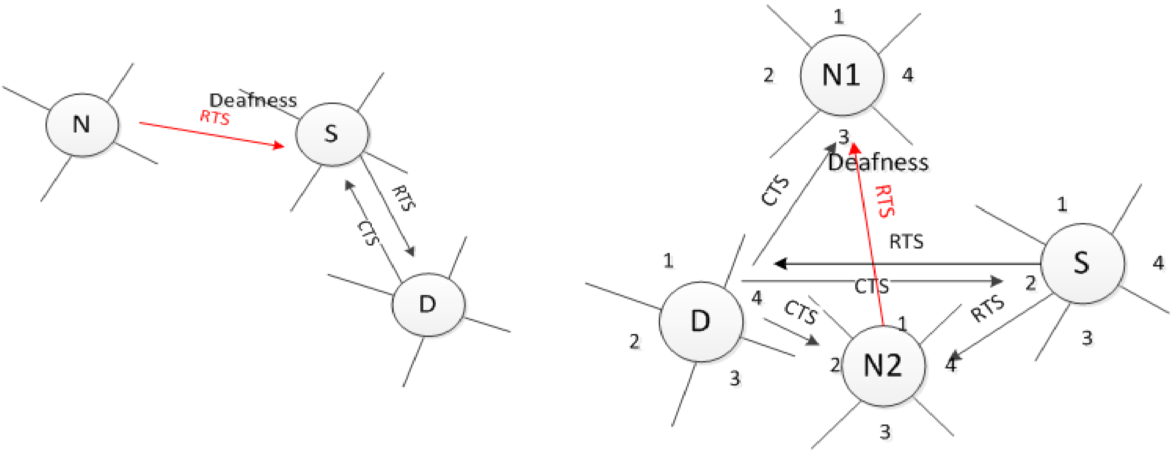


Fig. 4.13 Two scenarios of deafness

Two types of deafness are shown in the Fig. 4.13. The first scenario shows that node N does not know about the transmission between S and D, so if it has a packet to send to node S it sends RTS but node N does not hear. In the second scenario, node N1 sets its Directional Network Allocation Vector (DNAV) for beam 3 because of receiving CTS from the destination. Node N2 sets DNAV for beams 4 and 2 because of receiving RTS and CTS. If node N2 has a packet to send to N1, it starts sending RTS to N1 from beam 1, thus the deafness occurs.

A new directional MAC protocol has been proposed in [126], it includes a new scheme to inform its neighbours who were deaf because of other communications. Thus it solves the deaf node problem. Moreover, it prevents the hidden node problem. Each node has Multiple Beam Smart Antenna (MBSA) with non-overlapping directions that cover all directions around the node (2π rad). If a node wants to send a packet, the RTS/CTS handshake occurs directionally between the source and the destination. If it is completed, the communicating nodes send RTS/CTS simultaneously through all beams except the data communicating beams. Then they start transmitting data using the beam pointed each other and prevent the other beams from transmission and reception. When the neighbours hear a packet they set DNAV for that beam. After transmitting data, the idle neighbour of the node, which has just completed their transmission, will send a Neighbour Information Packet (NIP) to this node to be aware of on-going communication in the network. Thus by using this method of simultaneous transmission of RTS/CTS and transmission of NIP, the deaf and hidden node problem is prevented.

4.5 Neural Network for Mobility Prediction

Artificial Neural Networks (ANNs) are an information processing model that is inspired by the way biological nervous systems work, such as the brain, on a computer. It was founded by McCulloch and co-workers beginning in the early 1940s [128], [129].

ANNs are used in many applications to solve some real world problems. ANNs solve many engineering problems such as classifications, prediction, pattern recognition, and non-linear problems [130], [131]. Neural networks with its learning and generalization ability may act as a suitable tool to predict the location of an MN. The training process requires a set of examples of proper network behaviour (network inputs x and target outputs t). The process of training a neural network involves tuning the values of the weights and biases of the network to optimize network performance. The default performance function for feed-forward networks is Mean Square Error (MSE). MSE is the average squared error between the network outputs and the target outputs. If t is a vector of N targets and y is a vector of N predictions. MSE is defined as follows:

$$\mathrm{MSE} = \frac{1}{N}\sum_{i=1}^{N}(t_i - y_i)^2 \tag{8}$$

A neural network can be used to fit both linear and non-linear relationships in functions, this is the main reason of using the neural network [132]. In [64], Kaaniche and Kamoun introduced a neural network based method for mobility prediction in ad hoc networks. They used a time series prediction technique based on recurrent neural network a backpropagation multilayer neural network to estimate the future location of MN based on time series observations as the input of the network. This method consists of a three-layer and recurrent neural network. It used back propagation through time algorithm for training. In [133], the authors also predicted the mobility parameters using the time series. They proposed AutoRegressive Hello protocol (ARH) which is a mobility prediction scheme based hello protocol in mobile ad hoc networks. Each node predicts its own position by an ever-updated autoregression-based mobility model. The node sends ‘hello’ message for location update only when the predicted location is too different from the actual one. Each location update corresponds to a ‘hello’ message transmission

An adaptive learning automata-based mobility prediction method was presented in [134]. This method made prediction based on the Gauss–Markov random process, and exploiting the correlation of the mobility parameters over time. It used a continuous valued reinforcement scheme to learn how to estimate the future mobility characteristics based only on the mobility history. The mobility distribution parameters are assumed to be unknown, it means that this method does not need a prior knowledge of the mobility parameters. This proposed algorithm can be tuned to duplicate a wide spectrum of the mobility patterns with various degree of randomness (memory), and also includes the Random Walk, Random Waypoint and Brownian motion mobility models which are the three most popular mobility patterns widely used in mobile ad hoc networks.

Authors of [135], [136] proposed ELMs to fit and predict mobility of nodes in MANETs. This method predicted the future node positions and future distances between neighbouring nodes. The ELMs capture better the existing interaction/correlation between the Cartesian coordinates of the arbitrary nodes leading to more realistic and accurate mobility prediction based on several standard mobility models (Gauss-Markov, Random Waypoint, Random Walk and Manhattan). Prediction accuracy in terms of MSE was found to be 0.0623 for Random Waypoint Mobility model. This method gives more accurate results but fails when mobility history is missing [33].

Evolutionary algorithms are metaheuristic algorithms that provide quasioptimal solutions in a reasonable time. They have been applied to solve optimization problems related to a type of complex network like MANETs. A good overview of the application of evolutionary algorithms for MANETs can be found in [137].

4.5.1 Bayesian Neural Network

As it will be shown in more detail in Chapter 6, a Bayesian technique was used for learning or training ANNs because it offers as few assumptions as possible about the form that fits the data, while trying to simulate its shape [138]. The Bayesian method assumes that the function modelled should be continuous and differentiable. This method can address issues like regularization (overfitting or not) and model selection / comparison, without the need for a separate cross-validation data set. Also, this method can find the significance of each input which refers to the amount of variation in the output that can be caused by a particular input.

Bayes' Theorem

Bayes' theorem provides a direct method of calculating the probability of a hypothesis based on its prior probability, the probabilities of observing various data given the hypothesis, and the observed data itself [139]. The Bayes' rule can be used to determine the conditional probability of hypothesis h given data D:

$$P(h|D) = \frac{P(D|h)P(h)}{P(D)}, \tag{9}$$

in this formula, $P(h|D)$ is the conditional probability of h given data D. $P(h)$ is the prior probability of h before having seen the data D. $P(D)$ is the prior probability of D (probability that D will be observed). $P(D|h)$ is the probability of the data D for given h and is called the likelihood [140]. In general this ($P(D|h)$) will provide an entire distribution over possible values of D rather that the single most likely value of D.

We can apply this process to neural networks and calculate the probability distribution over the network weights given the training data, $P(w|D)$. Acquiring the weights in the Bayesian neural networks means changing the weights from the prior $P(w)$ to the conditional $P(w|D)$, as a result of observing data. With this method, error bars can also be placed on the output of the network, by considering the shape of the output distribution, $P(y|D)$.

Bayesian methods allow considering an entire distribution of answers instead of a single answer. Optimization methods focus on determining a single weight assignment that minimizes some error function (typically a least squared-error function). This is equivalent to finding a maximum of the likelihood function, i.e. finding a w^* that maximizes the probability of the data given those weights $P(D|w^*)$.

The next position of the MN X_{i+1} in MANETs is the parameter of interest in our work. It can be expressed as a non-linear function f of a number of experimental parameters in the database:

$$X_{i+1} = f(X_i, S_i, X_{i-1}, S_{i-1}, X_{i-2}, S_{i-2} \dots), \tag{10}$$

where $X_i, X_{i-1}, X_{i-2}, \dots$ are the positions of MN at time $(i, i-1, i-2, \dots)$ and $S_i, S_{i-1}, S_{i-2}, \dots$ are values of speed at time $(i, i-1, i-2, \dots\dots)$, respectively.

As it will be shown in section 7.3, only the current position and speed (X_i, S_i) are used to find the next position of MN. Therefore, the next position can be expressed as a non-linear function f of a current position and speed:

$$X_{i+1} = f(X_i, S_i) \tag{11}$$

4.6 Summary

The mobility prediction techniques and applications were presented in the introduction to this chapter. Many of these techniques lack the ability to predict random movements. However, all prediction methods depend on the fact that node movements are not completely random. The most applications of the mobility prediction in MANET were summarized as estimation of the link availability time, path reliability, route duration, network partitioning prediction, and routing enhancement.

A number of routing protocols have been developed for MANETs. Some of them are AODV, ODMRP and ODMRP-MP routing protocols which were shortly described in section 4.3. The AODV is simple, but efficient and effective routing protocol for MANETs and therefore it is used in my simulations as will be further discussed in Chapter 7.

A review of integration of directional antennas in ad hoc networks was presented in this chapter. Directional antennas allow the communication channel to be reused, therefore throughput capacity is enhanced compared to a network using omnidirectional antennas.

The end of this chapter was dealing with the neural network based method for mobility prediction in MANETs. Brief definition of principles of the Bayesian technique for learning ANNs was given because it is used for prediction in this work as will be seen in sections 6.2 and 7.3.

5 THESIS OBJECTIVES

With the development of new network technologies and techniques, MANETs are getting into the focus of interest very quickly. MANET networks have no fixed network infrastructure, which guarantees free movement to the end-stations in the entire network. On the other hand, this free movement of stations raises several new issues that must not have been solved in fixed network infrastructures. The majority of the nodes forming a MANET network are mobile and therefore they are usually powered by a battery supply. Due to this fact, there will be a particular emphasis on performance in the design of the prediction algorithm. Selecting and employing mobility models and mobility prediction techniques are extremely important in MANETs because it describes the node’s movement (node’s location, transmission range) from time to time.

Following the recent state of the art the purpose of study is twofold: (i) design of a prediction algorithm of the future movement of mobile stations in MANETs based on a virtual map and (ii) artificial neural network development for movement prediction in MANETs.

The main goal of this dissertation thesis is to evaluate the proposed mobility prediction methods. In order to achieve this goal, the following subtasks have been formulated:

- Evaluation of mobility prediction method for AODV routing protocol by comparison of differences in performance evaluation of the traditional AODV routing protocol and the Modified AODV (MAODV) which uses proposed mobility prediction method.
- Determination of the behaviour of the mobility prediction method using data collected from the node which moves according to Random Waypoint Mobility model and Random Walk Mobility model.
- Determination of the impact of the mobility prediction method on MNs' parameters such as delay, throughput and packet delivery ratio.
- Determination of the behaviour of neural network based method using data collected from the node which moves according to Random Waypoint Mobility.

6 PROPOSED MOBILITY PREDICTION ALGHORITHMS

In order to achieve the aim of the thesis, this chapter describes the prediction algorithm of the future movement of nodes based on the information of current network status. Since the nodes move continually with only a limited amount of energy, it is necessary to focus on the QoS of the communication process during designing new methods and functions.

6.1 Mobility Prediction Using Virtual Map

The prediction method supposes that each node can build its virtual map depending on its location over the time. This method is called mobility prediction using virtual map. A node uses the next step of moving to update neighbours within the transmission range and then the neighbours investigate this location information to estimate the new direction of this node. This solution joins the map based movement prediction with the usage of directional antennas. The method for prediction should be implemented and evaluated as will be shown in Chapter 7.

6.1.1 Mathematical Preliminaries

This section describes the mathematical terms I have used for mobility prediction.

6.1.1.1 Markov Chains

A Markov chain is a discrete random process with the Markovian property. It represents a transition between a finite number of probable states, where the next state depends only on the present state of the system and not on the past states [141] (first order Markov chain model). The statistical properties of the system help to predict the next steps. The state transition represents the changes of the system.

Assume a set of states $S = \{s_1, s_2, \dots s_n\}$, P_{ij} is the probability of movement from state s_i to state s_j. This probability does not depend on which state the chain was in before the current state where:

$$(i,j) \in S^2, p_{ij} \geq 0 \;\; \forall (i,j) \in S^2 \tag{12}$$

$$\sum_{j=s} p_{ij} = 1 \; \forall \; i \in S \tag{13}$$

Fig. 6.3 displays a Markov chain with three states $S = \{s_1, s_2, s_3\}$, the transition probabilities between these states can be represented as a matrix *P*:

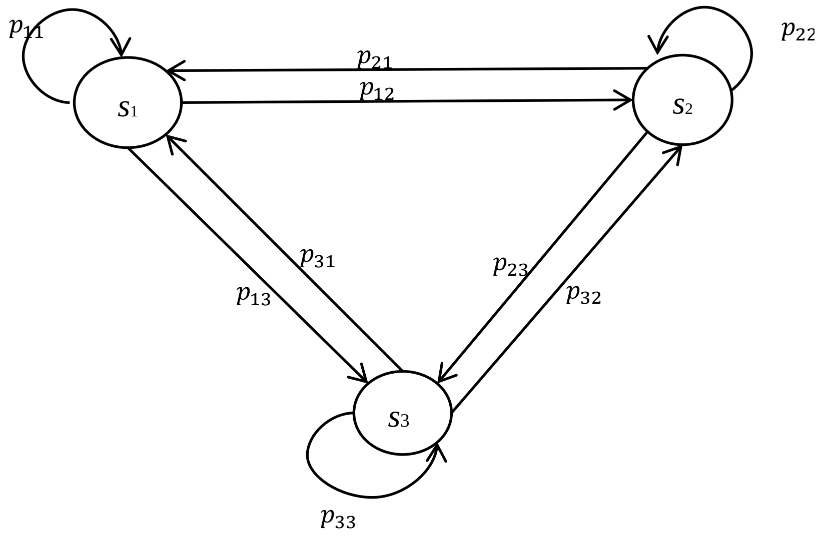


Fig. 6.1 A Markov chain for three states

$$P = \begin{bmatrix} p_{11} & p_{12} & p_{13} \\ p_{21} & p_{22} & p_{23} \\ p_{31} & p_{32} & p_{33} \end{bmatrix} \tag{14}$$

The first row of the transition matrix represents the probability of transition from $s_1$to s_1, s_2 or s_3. Similarly for the second and third rows.

In the case where the probability $p_{ij}(t)$ does not depend on when the transition occurred, we are talking about a homogeneous Markov where $p_{ij}(t) = p_{ij}$.

6.1.1.2 Maximum Likelihood Estimation

Maximum Likelihood Estimation (MLE) is the procedure of estimating the parameters of a statistical model given observations, by finding the parameter values that maximize the likelihood of making the observations given the parameters. The likelihood of a set of data is the probability of obtaining that particular set of data, given the chosen probability distribution model. This expression contains the unknown model parameters. The values of these parameters that maximize the sample likelihood are known as the Maximum Likelihood Estimates [142], [143]. Consider a Markov chain model of a random sequence states space, the likelihood function for a parameter w is denoted $L(w)$:

$$L(w) = P(s|w) = \prod_{i=1}^{n} P(s_i|w), \tag{15}$$

where s is observed data vector $(s_1, s_2, \dots, s_n)$. Maximizing $L(w)$ with respect to w will give the MLE estimation:

$$MLE = max_s[L(w)] = max_s \prod_{i=1}^{n} P(s_i|w) \tag{16}$$

For computational convenience, the MLE is obtained by maximizing the log-likelihood function, because the two functions, $ln[L(w)]$ and $L(w)$, are monotonically related to each other so the same MLE is obtained by maximizing either one:

$$MLE = max_s\left(ln\left[\prod_{i=1}^{n} P(s_i|w)\right]\right) = max_s\left(\sum_{i=1}^{n} ln[\,P(s_i|w)]\right) \tag{17}$$

6.1.2 Mobility Prediction Method

As a matter of fact, MN moves randomly without reference to a defined map. Thus I suppose that each node can build its virtual map depending on its location over the time [144]. Suppose there is a location X_1, $X_{2, \ldots\ldots}$ X_n of n independent observations. It is assumed that each MN has a Location Table (LT) in which the location information of its neighbours is temporarily stored.

At each location i, the probability of node's movement to another location j is calculated as:

$$P_{ij} = P(X_j|X_i) = P(L_{n+1} = X_j|L_n = X_j), \tag{18}$$

where $j = 0, 1, \ldots \ldots, n$. Based on Maximum Likelihood Estimation the node selects the next location which is the most probable outcome P_{ij}.

When node A intends to send a message to node B, the (RTS|CTS) handshaking occurs. The RTS packet includes location information of A: the time instant of RTS transmission t_r, ID_A, current position $L_A(t_r) = (x_A(t_r), y_A(t_r))$, and speed S_A. In addition, the most probable coordinates of the next position $L_{A^*}(t_r) = (x_{A^*}(t_r), y_{A^*}(t_r))$ calculated according to the A's virtual map are also transmitted, Fig. 6.2. Therefore node B saves A's information in its LT:

$$LT_B\,(A) = (t_r, ID_A, L_A(t_r), S_A, L_{A^*}(t_r), TTL_A\,), \tag{19}$$

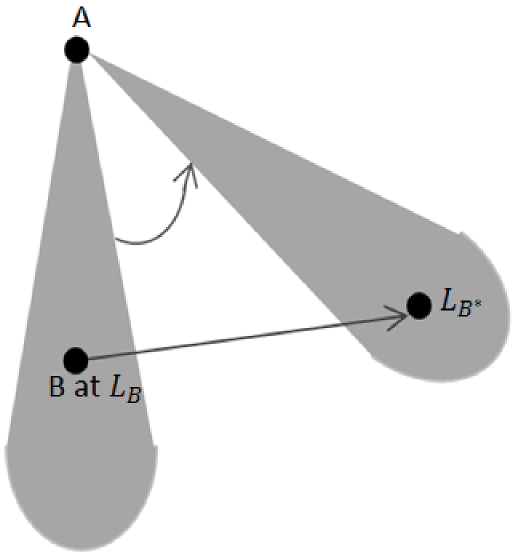


Fig. 6.2 The antenna of node A is pointed to the predicted direction of node B

where TTL_A is Time To Live of node A, TTL_A = 0 at each registration of $LT_B\ (A)$ and then TTL_A increases with time. When it exceeds a certain threshold T, the $LT_B\ (A)$ is deleted from the MN's LT.

The CTS packet from B includes location information of B, and node A saves this information in its LT:

$$LT_A\ (B) = (t_c, ID_B, L_B(t_c), S_B, L_{B^*}(t_c), TTL_B\ ). \tag{20}$$

Hence the distance between the position of B and $\mathrm{L_{B^*}(t_c)}$ is:

$$D = \sqrt{(x_{B^*} - x_B)^2 + (y_{B^*} - y_B)^2}. \tag{21}$$

Node A can estimate the position of node B at t_{c+1} from Fig. 6.3 using similarity of triangles:

$$\frac{x_B(t_{c+1}) - x_B(t_c)}{x_{B^*}(t_c) - x_B(t_c)} = \frac{y_B(t_{c+1}) - y_B(t_c)}{y_{B^*}(t_c) - y_B(t_c)} = \frac{(t_{c+1} - t_c)S_B}{D}. \tag{22}$$

As a result,

$$\begin{gathered} x_B(t_{c+1}) = x_B(t_c) + \left(x_{B^*}(t_c) - x_B(t_c)\right)\frac{(t_{c+1} - t_c)S_B}{D}. \\ y_B(t_{c+1}) = y_B(t_c) + (y_{B^*}(t_c) - y_B(t_c))\frac{(t_{c+1} - t_c)S_B}{D}. \end{gathered} \tag{23}$$

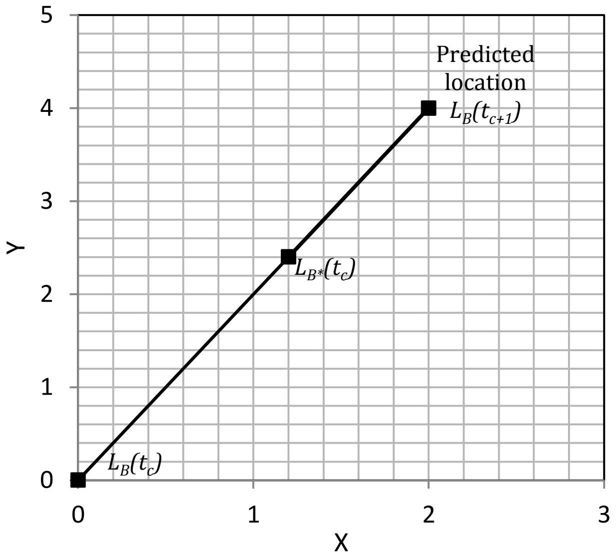


Fig. 6.3 Prediction of location using the current and the next position

Therefore, the transmission angle of data from node A to B at t_{c+1} is:

$$\theta_{data}(t_{c+1}) = \tan^{-1}\frac{y_A(t_{c+1}) - y_B(t_{c+1})}{x_{\mathrm{A}}(t_{c+1}) - x_B(t_{c+1})}. \tag{24}$$

Then the antenna of nodes A and B is pointed to the predicted direction to send data. However, during the transmission between A and B, it is possible that position $\mathrm{L_{B^*}}(\mathrm{t_c})$ is achieved by B and B will continue its movement to the next new position, thus the directional beacon of node B will distribute this new information to B's neighbours within the transmission range, thereby node A updates B's location information to estimate the new direction of node B.

This scheme provides an adaptive location prediction mechanism. It proactively predicts future locations of communicating nodes and minimizes location updating, thereby reducing communication delay.

The main technique to evaluate the performance of MANETs has been the use of network simulator NS-2 version 2.35. Because the real testbeds need a high investment in terms of hardware, and, more importantly, the replication of real mobile conditions is very difficult in a controlled environment like a laboratory. This algorithm has been implemented in NS-2 in order to be able to investigate its impact on network performance with a number of experiments. A summary overview of NS-2 is given in the Annex A. The entire simulation work is conducted and implemented on a Linux (Ubuntu distribution) operating system. For a realistic simulation of the network, two mobility models (Random Walk and RWP models) are chosen to illustrate the performance advantage from the proposed mobility prediction algorithm.

BonnMotion is used to create mobility scenarios [145]. BonnMotion is a Java software which creates and analyses mobility scenarios and is most commonly used as a tool for the investigation of mobile ad hoc network characteristics. The parameters for the scenario can be specified through the command line. For example, the Random Walk model is generated as follows:

bm -f randwalkfile RandomWalk -n 20 -x 500 -y 500 -t 10 -d 100 -h 5 -i 1000

where:

- f: output filename
- n: 20 number of nodes
- x: 500 width of the simulation movement field (m)
- y: 500 length of the simulation movement field (m)
- t: 10 time limited mode (s)
- d: 100 duration of simulation (s)
- h: 5 maximum speed (m/s)
- i: 1000 number of seconds to skip from starting point

The scenarios can also be exported for several network simulators, such as GloMoSim/QualNet, COOJA, MiXiM, and NS-2. Command to convert output file to NS-2 format:

bm NSFile – f randwalkfile

this command generates movement file and parameter file for Random Walk Mobility model. This file can be used as an input to the Tcl script.

6.2 Bayesian Neural Network for MANETs

The aim of the research has been to develop prediction model for mobility in the MANETs using the data collected from location patterns. MacKay's Bayesian framework for backpropagation is a practical and powerful means to improve the generalization ability of Artificial Neural Network (ANN) [146]. The software used for training Bayesian neural networks is called Model Manager [147]. Model Manager is a graphical interface developed by MacKay. It incorporates further practical methods which further contribute to the successful completion of modelling. The database is randomly divided into training and testing sets, to ensure that both the half used for training and testing contains similar information. The Model Manager enables to form final model. This final model is built from a set of multiple submodels. The optimum number of submodels to form the set is determined depending on the combined test error of all the members of the set. This method further attempts to find the appropriate level of complexity from the data, and to ensure a robust solution is found. The Bayesian method assumes that the function modelled should be continuous and differentiable. MacKay has shown that a sufficiently complex three layer network using hyperbolic tangent functions in the form of (27), is able to imitate any such function [148]. Therefore, the three layer neural network, as characterised in section 6.2.1, is used to predict the future location of a node. The training a neural network involves tuning the values of the weights based on the data given. For modeling the mobility behaviour, the literature offers a set of mobility [13]. In this work I used Random Waypoint Mobility to construct location patterns. The Random Waypoint is one of the most popular mobility models used to evaluate the mobility prediction in MANET, because it is flexible and simulates in a realistic way the movement of people. The movement of an ad hoc MN can be described by two coordinates (x, y) and speed (s). Numerical inputs have been available from 200 patterns obtained from the simulation of Random Waypoint model. The first 150 patterns are used for training network and the rest for predicting. I have used three inputs (the current coordinates and speed (x, y, s)) to predict the next coordinates.

For completing training without overfitting, the mobility prediction model prepares the database before training as follows: 1) the database is randomly divided into training and testing sets, 2) the minimum and maximum of each variable and the target are searched, 3) the inputs are normalized within a range of ±0.5. The aim of normalizing is to compare the sensitivity of the prediction results for different inputs without biasing the comparison because of the different magnitudes of the set of inputs. The normalizing is done as follows [149]:

$$I_n = \frac{I - I_{min}}{I_{max} - I_{min}} - 0.5, \tag{25}$$

where I is the unnormalized input, I_{max} and I_{min} are the maximum and minimum values in the database for a particular input. I_n is the normalized value.

6.2.1 Structure of the Neural Network

In the three-layer feed-forward neural network the first layer consists of the inputs to the network, the next layer consists of a number of non-linear operators h_i which form the hidden layer, and the third layer consists of the output function. Fig. 6.4 shows an example of feed-forward neural network which contains three layers.

Data move in only from the input nodes, through the hidden nodes and to the output node. There are no loops in the network; every node in a layer is connected with all the nodes in the previous layer, hence called feed-forward ANN [150]. Each connection may have a different weight. The activation function for i^{th} node is given by following equation:

$$h_i = \tanh\left(\sum_j w_{ij}^{(1)} I_i + \theta_i^{(1)}\right). \tag{26}$$

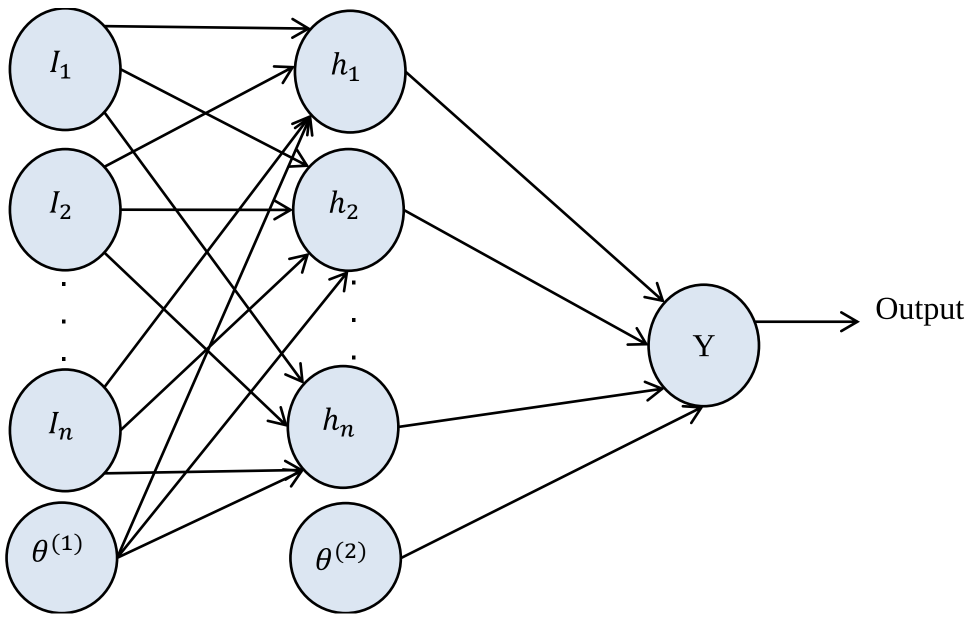


Fig. 6.4 The architecture of feed-forward neural network

The output is given by equation:

$$Y = \sum_i w_{ij}^{(2)} h_i + \theta^{(2)}, \tag{27}$$

where I_i are inputs, and w_{ij} are the weights which define the network. The superscripts $^{(1)}$ and $^{(2)}$ denote weights and biases in the hidden layer and in the output layer. The optimum value for w is obtained through training the network. The parameters θ are known as biases.

The complexity of any neural network increases by increasing the number of hidden units. During the training phase the inputs are known, the output is known, and the weight can be examined. In order to find the interactions between inputs and output a model makes predictions and visualises the behaviour which emerges from various combinations of inputs.

One of the problems that can occur during neural network training is called overfitting, which leads to an unjustified level of accuracy and thus, a high level of complexity. If a model is too complex it may give poor generalization (overfitting). Overfitting occurs when the network has memorized the training set but has not learned to generalize to new inputs. Overfitting produces a relatively small error on the training set but a much larger error when new data is presented to the network. Training a network includes finding a set of weights and biases which balances between complexity and accuracy as illustrated by the following equation [150]:

$$M(w) = \alpha E_w + \beta E_D \,, \tag{28}$$

where E_w is an organizer of the complexity. It forces the network to use small weights and limited number of hidden units:

$$E_w = \frac{1}{2}\sum_{ij} w_{ij}^2. \tag{29}$$

E_D is the overall error between target output values and network output values:

$$E_D = \frac{1}{2}\sum_{k} \left(t^k - y^k\right)^2, \tag{30}$$

where t^k represents the set of targets for the set of inputs I^k, and y^k represents the set of corresponding network outputs. In (28), α and β represent control parameters which define the trade-off between complexity and accuracy of the model. The training algorithm updates the weights and biases to minimise a combination of squared errors and weights and then determines the correct combination to produce a network that generalizes well.

Training the neural network is achieved by modifying the weights to fit the functions to the data using backpropagation gradient descent optimisation procedures [150], to minimise an objective function. This is achieved by calculating and optimising an objective function (28). The fitting method infers a probability distribution for the weights from the data presented instead of identifying one best set of weights. It is popular to use the test error (sum squared error) as the default metric for choosing the best model, but in fact this may be a misleading criterion [148]. In many applications there will be an opportunity not to simply make a scalar prediction, but rather to make a prediction with error bars, or maybe an even more complicated predictive distribution. It is then reasonable to compare models in terms of their predictive performance as measured by the log predictive probability of the testing data. Under the Log Predictive Error (LPE), the penalty for making a wild prediction is much less if that wild prediction is accompanied by appropriately large error bars. The test error does not have this advantage. Assuming that for each set k the model gives a prediction according to the normal distribution with average $y^{(k)}$ and variance $\left(\sigma_y^{(k)}\right)^2$, the LPE is calculated as follows [151]:

$$LPE = \frac{1}{2}\sum_k \left[\frac{(t^{(k)} - y^{(k)})^2}{(\sigma_y^{(k)})^2} + \log\left(\sqrt{2\pi}\left(\sigma_y^{(k)}\right)^2\right)\right], \tag{31}$$

where $\sigma_y^{(k)}$ is related to the uncertainty of fitting for the set of inputs $I^{(k)}$. This error penalizes unexpected predictions to a lesser extent when they have large error bars (uncertainties).

When a neural network is used, it is important to distinguish between the two types of error. Noise means little changes in the results when the experiment is repeated a number of times due to uncontrolled variables. The noise is constant, so it does not contribute to the evaluation of the behaviour of the model. The second type of error is uncertainty, which refers to doubtfulness in the mathematical functions capable of representing the same data. Once more data are observed, the uncertainty can decrease, allowing the predictions made by the network to become more accurate. Modelled uncertainties are presented as error bars. The average of the error bars is calculated as follows:

$$E_{bar} = \frac{1}{N}\sum_{i=1}^{N} E_i, \tag{32}$$

where N represents the total number of predictions and E_i the error accompanying each prediction. The root mean square residual (RMS) was used to evaluate the final model and it was calculated as follows:

$$R_{test} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(t_i - y_i)^2}, \tag{33}$$

where t_i and y_i are the target value and network output respectively.

The final model is built from a set of multiple models. The model which is a member in this set will be called as submodel. The optimum number of submodels to form the set is determined depending on the combined test error of all the members of the set. The prediction or the output of a set of submodels is the mean prediction of its members ($\bar{y}$):

$$\bar{y} = \frac{1}{L}\sum_{l} y^{(l)}, \qquad (34)$$

and the variance σ^2 is given by equation:

$$\sigma^2 = \frac{1}{L}\sum_{l}(\sigma_y^{(l)})^2 + \frac{1}{L}\sum_{l}(y^{(l)} - \bar{y})^2, \qquad (35)$$

where $\bar{y}$ represents the mean prediction of submodels which form a set, L is the number of submodels in the set and the exponent l refers to the submodel used to give the corresponding prediction $y^{(l)}$.

6.3 Summary

This chapter presented the theoretical description and used mathematical calculus of proposed algorithms for prediction of the future movement of nodes based on the information of current network status. The first method joins the map based movement prediction with the usage of directional antennas. It proactively predicts future positions of communicating nodes and minimizes location updating.

The second method uses artificial neural network for movement prediction in MANETs. It depends on Bayesian technique for training neural network. Only the current position and speed are used to find the next position of MN. It has been implemented in Model Manager for training Bayesian neural networks.

Both proposed prediction approaches are further implemented into the Random Walk and Random Waypoint models in Chapter 7.

7 PROPOSED MOBILITY PREDICTION METHODS' IMPLEMENTATION AND THE RESULTS

This chapter describes the implementation of the mobility prediction algorithm described in section 6.1. The algorithm has been implemented in NS-2 in order to be able to investigate its impact on network performance with a number of experiments. NS-2 models a realistic mobility of the nodes and also includes an accurate model of the IEEE 802.11 Distributed Coordination Function (DCF) wireless MAC protocol.

7.1 Evaluation of Mobility Prediction Method Using Virtual Map

For a realistic simulation of the network, two mobility models (Random Walk and Random Waypoint models) are chosen to illustrate the performance advantage of the proposed mobility prediction algorithm. A network area of (500×500) m^2 is built on the simulation platform and simulation time is for 100 sec. Traffic type is Transmission Control Protocol (TCP). The simulation parameters considered for the performance evaluation of the mobility prediction algorithm are summarized in Table 7-1.

Table 7-1 Simulation parameters used in this study

Platform	Linux (Ubuntu 14.04)
Simulation Tool	Network Simulator 2 (NS-2)
Simulation area	500x500
Wireless mac interface	802_11
Propagation model	Two ray ground
Channel	Wireless channel
Max packet in interface queue	50
Connection Type	TCP
Application	FTP
Antenna Type	Antenna/DirAntenna
simulation time	100 sec
Maximum Speed	1.5m/sec

The mobile node is considered to be carried by people or by autonomous system. In order to evaluate the proposed mobility prediction method, I provide discussion about network simulation based performance evaluation of the traditional AODV routing protocol and AODV with proposed mobility prediction method. MAODV will refer to AODV with the mobility prediction method. The simulation environment consists of five different numbers of nodes which are 20, 40, 60, 80 and 100 MNs. Both Random Walk model and RWP model are also chosen to illustrate the performance metrics.

Using NS-2, I created an adaptive location predictive algorithm using new_app parameter. A new procedure using source, destination and all given parameter is generated. It takes an element to store location information and then it finds the shortest path using current position and range of changing position. The procedure tests if the neighbour's history is available then updates all node current position, next position, virtual map and location storage table. Otherwise add all the history of neighbour node in neighbour history file. The procedure stores all the current position, change position, next hop and all in a list.

Performance Metric

Simulations evaluate proposed mobility prediction based on an end to end delay, throughput, and packet delivery ratio using NS-2 simulation tools:

1. The average end to end delay is the average time taken for a data packet to be successfully transmitted over a MANET from source to destination. Mathematically, the average end to end delay is computed as:

$$E(D) = \frac{\sum_{i=1}^{n}(R_i - S_i)}{n} \qquad (36)$$

   where $E(D)$ is the average end to end delay, n is the number of data packets successfully transmitted over the network, R_i is the time at which a packet with unique identifier i is received and S_i is the time at which a packet with unique identifier i is sent. The average end to end delay should be low for high performance.

2. Packet Delivery Ratio (PDR) is the ratio of packets successfully received to the total sent and is determined as:

$$PDR = \frac{Number\ of\ Packets\ Received}{Number\ of\ Packets\ Sent} * 100 \qquad (37)$$

For better performance, the PDR should be high.

3. Throughput is defined as the total packets delivered over the total simulation time. In other words; it is the rate of successful data delivery across a network. The throughput is usually measured in terms of bits per second. A higher throughput implies better QoS of the network.

The following sections illustrate how the performance of the proposed method varies across different mobility models, number of nodes and speeds.

7.1.1 Performance metrics of Random Walk Model

This analysis includes the simulation of 20, 40, 60, 80, 100 nodes. A network area of $(500 \times 500)\ m^2$ is built on the simulation platform and total simulation time is 100 sec. Traffic type is TCP. The values of performance metrics are given in Table 7-2 to evaluate the performance of network for Random Walk model with traditional AODV. A Tcl script is written in NS-2 for simulation of network model. Some important NS-2 commands used for the simulation are as follows:

```
set val(chan)   Channel/WirelessChannel    ;        # channel type
set val(prop)   Propagation/TwoRayGround   ;        # radio-propagation model
set val(netif)  Phy/WirelessPhy            ;        # network interface type
set val(mac)    Mac/802_11                 ;        # MAC type
set val(ifq)    Queue/DropTail/PriQueue    ;        # interface queue type
set val(ll)     LL                         ;        # link layer type
set val(ant)    Antenna/DirAntenna         ;        # antenna model
set val(ifqlen) 50                         ;        # max packet in ifq
set val(rp)     AODV                       ;        # routing protocol
```

The randomwalk parameter file and randomwalk movement file, that were generated by bonnmotion tool, specify the parameters for the mobility scenario. These files are called using NS-2 command as follows:

```
source randomwalk20n-ALL.ns_params
source randomwalk20n-ALL.ns_movements
```

The parameter file has complete set of parameters used for the simulation, and the movement file contains the movement data.

Table 7-2 The values of performance metrics of Random Walk model for AODV

Number of nodes	Average end to end delay (sec)	Packet delivery ratio	Throughput (Kbps)
20	0.383	65.66	51.58
40	0.225	88.71	93.69
60	0.395	80.66	80.11
80	0.688	75.00	134.49
100	0.323	88.75	211.46

In order to measure the performance of this network using AODV as routing protocol without the proposed mobility prediction method, the values are obtained from the trace file generated by NS-2. Using the AWK script the data is processed and is used to measure the performance metrics. The values of these metrics are listed in Table 7-2. Table 7-3 presents the performance of this network using MAODV as routing protocol with the proposed mobility prediction method.

Table 7-3 The values of performance metrics of Random Walk model for MAODV

Number of nodes	Average end to end delay (sec)	Packet delivery ratio	Throughput (Kbps)
20	0.153	97.75	67.01
40	0.157	95.66	95.35
60	0.385	88.00	93.03
80	0.484	80.00	133.45
100	0.227	90.64	125.97

As it can be seen, Fig. 7.1-Fig. 7.3 illustrate the comparative between AODV and MAODV for Random Walk model. The range of number of nodes is between 20 and 100. The red line provides the AODV and the blue line demonstrate the MAODV. The upward and downward arrows refer to the corresponding percentage change for each number of nodes.

Fig. 7.1 shows that the AODV routing protocol has delay higher without the mobility prediction than the delay with the mobility prediction. The delay includes all possible delays i.e. buffering route discovery latency, queuing at the interface queue, retransmission delay at MAC and propagation delay. In case of MAODV, the delay increases when the number of nodes increases between 40 and 80 nodes. Until 80 nodes the delay of MAODV decreases when the number of nodes increases. Based on this result, the MAODV has better delay performance than the AODV. For 20 nodes, the delay of MAODV is 60% lower than the delay of AODV.

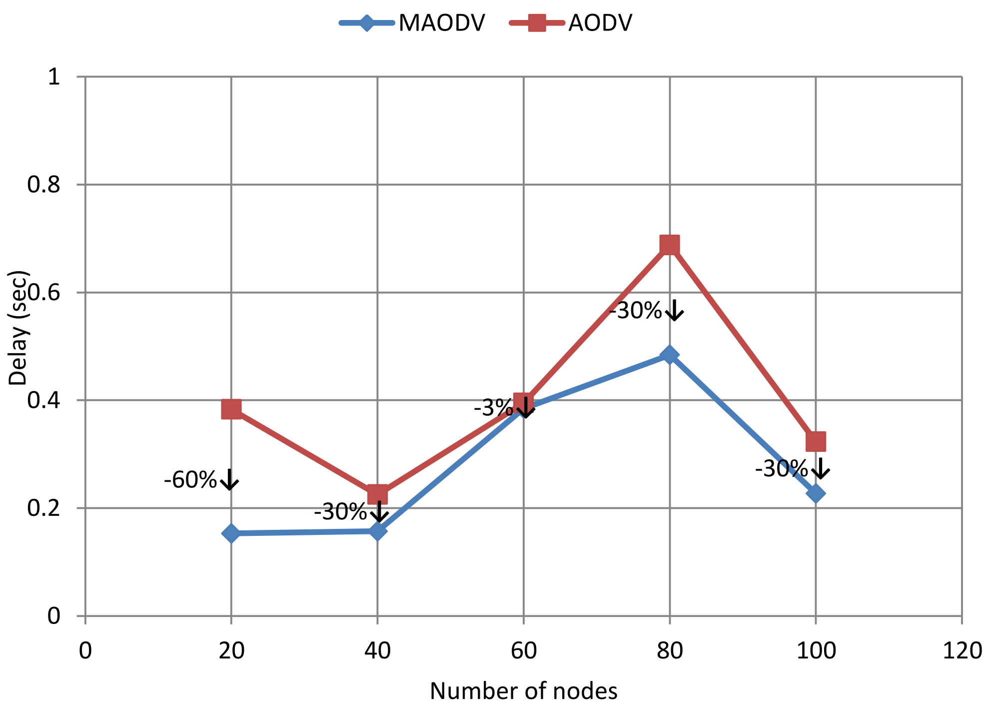


Fig. 7.1 Comparison of average end to end delay of Random Walk model

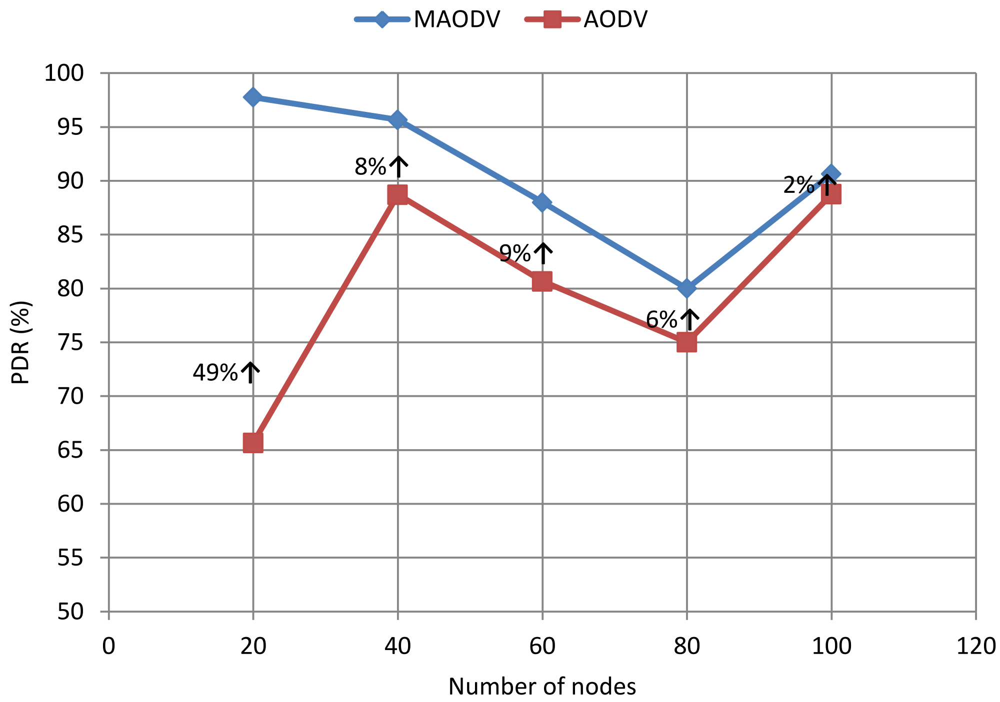


Fig. 7.2 Comparison of packet delivery ratio of Random Walk model

Fig. 7.2 shows that the PDR of MAODV is higher than PDR of AODV. For 20 nodes, the PDR of MAODV is 49% greater than the PDR of AODV.

The curve of the throughput for Random Walk model is illustrated in Fig. 7.3. For number of nodes greater than 60, the throughput of AODV increases when the number of nodes increases. The throughput for MAODV is better than the throughput of AODV for the range of number of nodes between 20 and 80. While for 100 nodes, the throughput of AODV is better than the throughput of MAODV. Because as the number of nodes increases in a limited area of network, there is always better and better path between source and destination and therefore there is no need to predict the mobility of a node for high number of nodes. In other words, since the time to estimate the position of all other neighbouring nodes might be more significant before the data transmission, the throughput decreases.

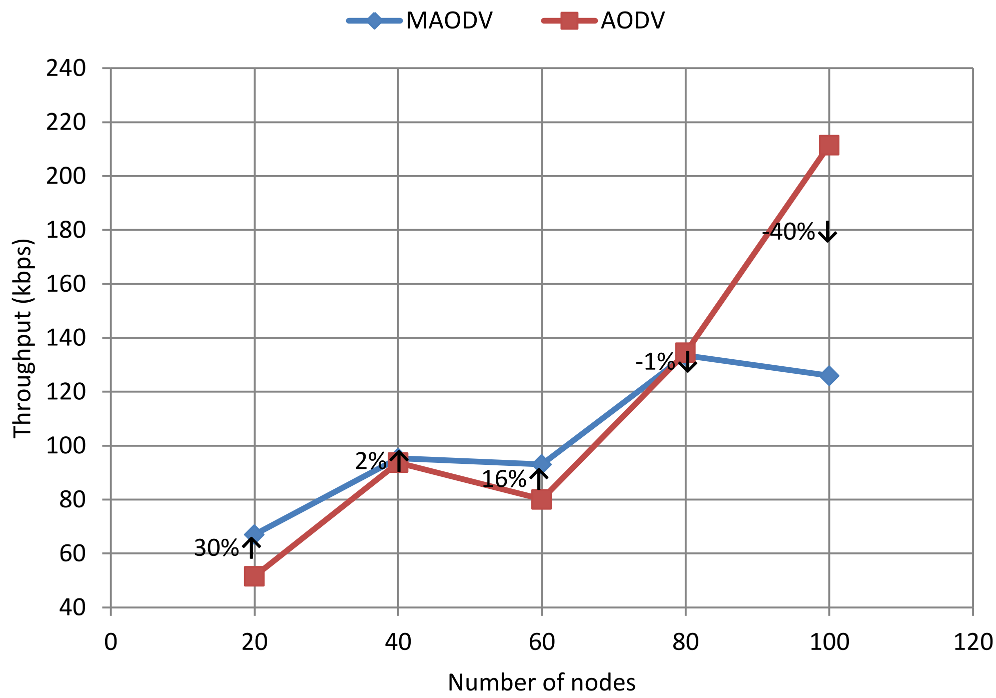


Fig. 7.3 Comparison of throughput of Random Walk model

The MAODV performed better as compared to AODV. This is due to the fact that MAODV takes into account the next position of the destination to point the antenna to the predicted direction to send data. Therefore, the probability of packet drop decreases, the delay decreases and the performance improves.

7.1.2 Performance metrics of Random Waypoint Model

This analysis includes the simulation of 20, 40, 60, 80, 100 nodes. A network area of 500 by 500 m^2 is built on the simulation platform and total simulation time is 100 sec. Traffic type is TCP. Same scenario (Random Waypoint model) is used for both AODV and MAODV. The values of performance metrics are listed in Table 7-4 to evaluate the performance of network for Random Waypoint model with traditional AODV.

The Random Waypoint movement file for node movement, generated by BonnMotion tool, is called using NS-2 commands as follows:

source randomwaypoint20n-ALL.ns_movements;

this file contains the movement data.

Table 7-4 The values of performance metrics of Random Waypoint model for AODV

Number of nodes	Average end to end delay (sec)	Packet delivery ratio	Throughput (Kbps)
20	0.131	90.63	68.60
40	0.156	68.86	84.64
60	0.132	98.71	124.25
80	0.130	98.63	174.87
100	0.121	98.68	216.19

Table 7-5 The values of performance metrics of Random Waypoint model for MAODV

Number of nodes	Average end to end delay (sec)	Packet delivery ratio	Throughput (Kbps)
20	0.123	98.70	69.50
40	0.127	98.66	95.85
60	0.125	98.68	146.79
80	0.130	98.63	169.44
100	0.123	98.75	178.51

From Fig. 7.4, the delay in both AODV and MAODV is almost the same for the range of number of nodes between 60 and 100. But for number of nodes between 20 and 60, AODV has delay higher than MAODV. Based on this result, the MAODV has better delay performance than the AODV for the range of number of nodes between 20 and 60. For 40 nodes, the delay of MAODV is 19% lower than the delay of AODV. The number of nodes does not have influence on the delay for the range of number of nodes between 60 and 100.

The PDR in both AODV and MAODV is almost the same for the range of number of nodes between 60 and 100 as shown in Fig. 7.5. For the range of number of nodes between 20 and 60, the MAODV performed better as compared to AODV.

Based on the result of Fig. 7.6, the throughput of both AODV an MAODV increases when the number of nodes increases. For the range of number of nodes between 20 and 80, the throughput of MAODV performed better as compared to the throughput of AODV. But AODV has throughput higher than MAODV for number of nodes equal to 100.

As it can be seen for the range of number of nodes between 20 and 60, MAODV performed better than AODV in term of delay, PDR and throughput. While for number of nodes greater than 60, the performance of MAODV is not improved. This is due to the fact that as the number of nodes increases in a limited area of network, the distance between source and destination decreases and therefore the probability of packet drop decreases. Subsequently, there is no need to predict the mobility of a node for high number of nodes.

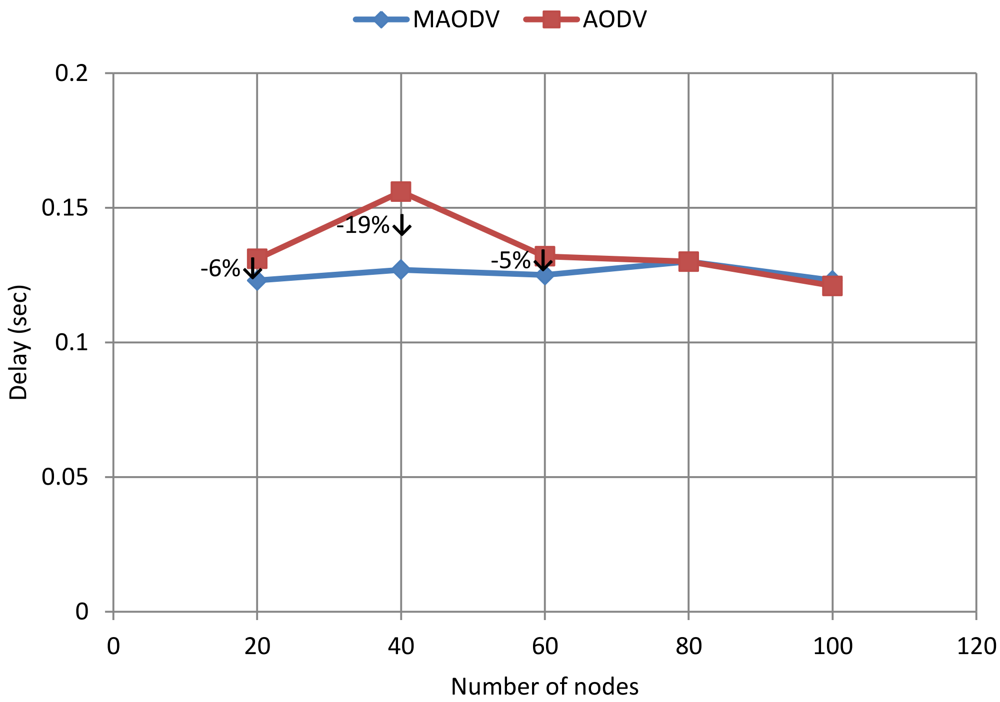


Fig. 7.4 Comparison of average end to end delay of Random Waypoint model

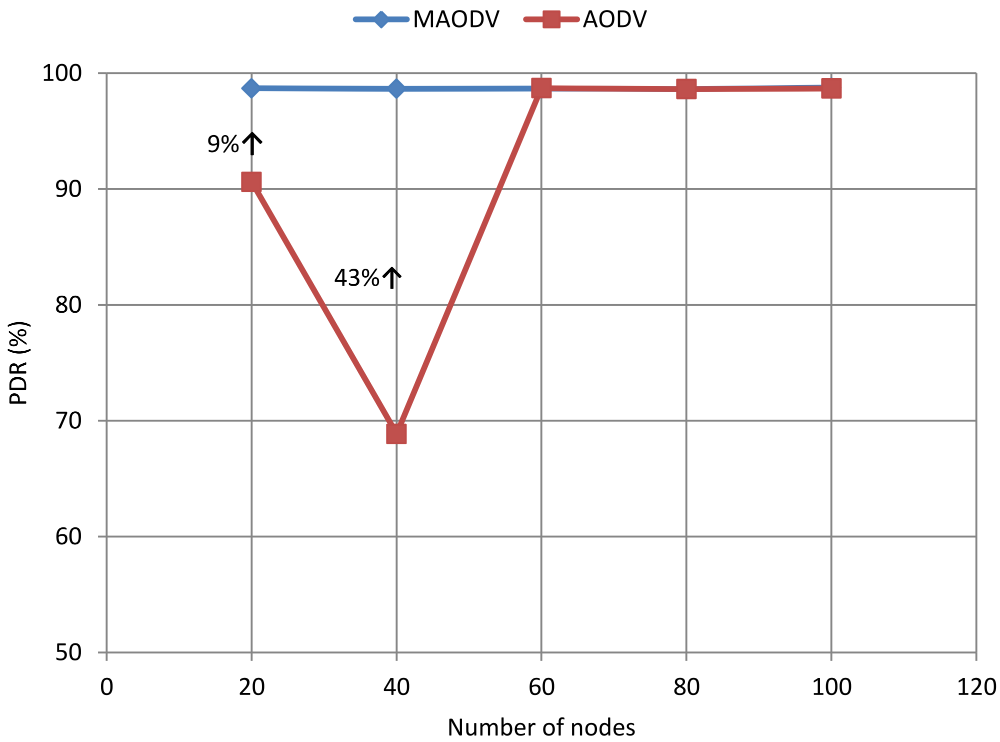


Fig. 7.5 Comparison of packet delivery ratio of Random Waypoint model

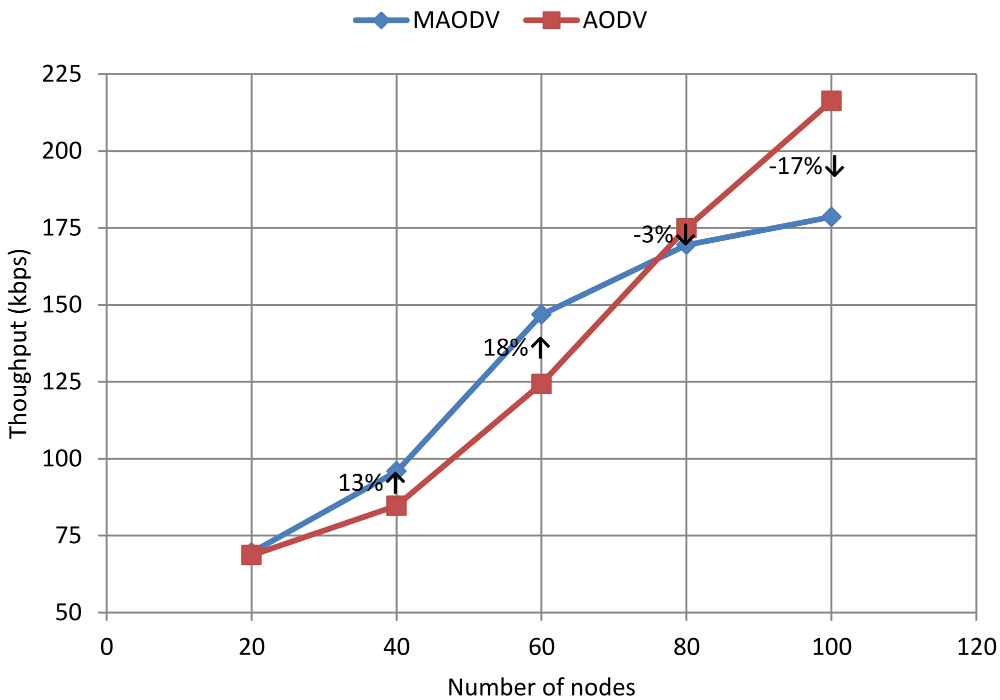


Fig. 7.6 Comparison of throughput of Random Waypoint model

7.1.3 Improvement of the Proposed Method MAODV

After analysis of traditional AODV and the proposed MAODV the results are shown in Table 7-6 and Table 7-7. These tables show improvement in performance metrics of the proposed method MAODV for Random Walk and Random Waypoint models, the range of number of nodes is between 20 and 80.

As can be seen in Table 7-6 with reference to the performance for Random Walk, I observed that the delay of the proposed method MAODV is 30% lower than the delay of AODV. The packet delivery ratio of MAODV has 13% improvement compared to AODV. The results of simulation indicate to 8% improvement in throughput of MAODV compared to AODV.

Table 7-6 Improvement of the proposed method MAODV for Random Walk

	AODV	MAODV	Improvement [%]
Average end to end delay [sec]	0.403±0.23	0.281±0.17	30%
Packet delivery ratio [%]	79.76±6.86	90.41±8.88	13%
Throughput [kbps]	89.97±20.40	97.21±33.22	8%

Table 7-7 Improvement of the proposed method MAODV for Random Waypoint

	AODV	MAODV	Improvement [%]
Average end to end delay [sec]	0.134±0.013	0.126±0.004	6%
Packet delivery ratio [%]	91.10±14.89	98.68±0.04	8%
Throughput [kbps]	113.09±45.12	120.40±49.97	6%

The performance of MAODV for Random Waypoint has been improved as well, see Table 7-7. In comparison to the AODV, the delay has 6% improvement, the packet delivery ratio of MAODV has 8% improvement and the throughput has 6% improvement.

Hence the MAODV performed better as compared to AODV for both mobility models. As I mentioned before that MAODV takes into account the next position of the destination to point the antenna to the predicted direction to send data. Therefore the probability of packet drop decreases, the delay decreases and the performance improves.

7.2 Determination of the Behaviour of the Mobility Prediction Method

This section investigates how the proposed mobility prediction method behaves with different mobility models (Random Waypoint, Random Walk). These mobility models are the most frequently used mobility model in MANET simulations. The simulation parameters considered for the performance evaluation of the mobility prediction algorithm are summarized before in Table 7-1. The analysis based on comparing the different metrics of the protocols mobility models that I described previously in section 3.

7.2.1 Effect of Varying Number of Nodes

The simulation environment consists of five different numbers of nodes which are 20, 40, 60, 80 and 100 MNs. Through the simulation of Random Walk model and Random Waypoint model, compare their performance in the number of nodes. The values of performance metrics are listed in Table 7-3 to evaluate the performance of network for Random Walk model. Table 7-5 shows the values of performance metrics to evaluate the performance of network for Random Waypoint model.

Random Walk Mobility model and Random Waypoint Mobility model both are actually same mobility models apart from the pause time which is zero in Random Walk Mobility model. The max pause time used in Random Waypoint simulation was 60 sec. Their speed, direction and angle of motion are similar to each other.

The comparison of the two investigated mobility models, Random Walk model and Random Waypoint model, is shown in Fig. 7.7- Fig. 7.9 for the range of number of nodes between 20 and 100. The red line provides the Random Walk and the blue line demonstrates the Random Waypoint model. The upward and downward arrows refer to the corresponding percentage change for each number of nodes.

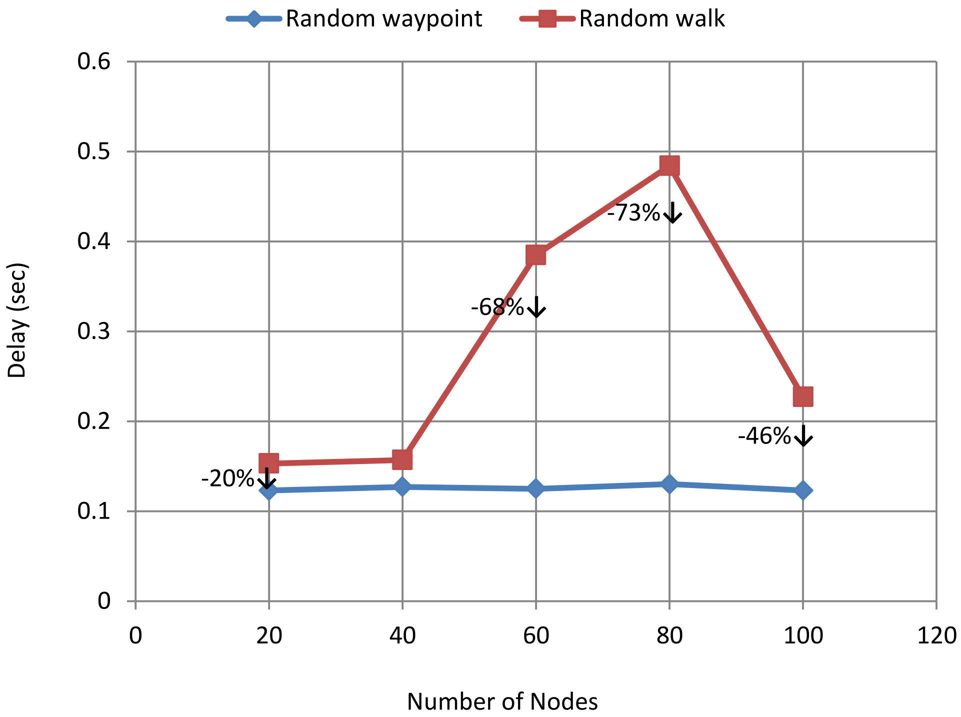


Fig. 7.7 Average end to end delay for different number of nodes

Fig. 7.7 shows average time which is taken by a data packet to reach the destination. In case of Random Walk model (red curve), the delay increases when the number of nodes increases between 20 and 80 nodes. Until 80 nodes the delay on the Random Walk decreased when the number of nodes increases. This is because the density of nodes is increased, therefore, the probability of finding neighbouring nodes (when a node wants to send data to a particular node) increases when there are more nodes in the network. The delay of the Random Waypoint model (blue curve) is remained unchanged and less as compared with the Random Walk model. From this density of nodes the Random Waypoint performed better as compared to Random Walk model.

The curve of the packet delivery ratio is shown in Fig. 7.8, the Random Waypoint model performed better in delivering packet data to the destination. The Random Waypoint model almost ensures a successful transfer between nodes 20 and nodes 100. The PDR for Random Waypoint is remained unchanged. And then the PDR for the Random Walk model decreased with increasing the number of nodes.

The curve of the throughput for both mobility models is illustrated in Fig. 7.9. The Random Waypoint shows throughput higher than the Random Walk model.

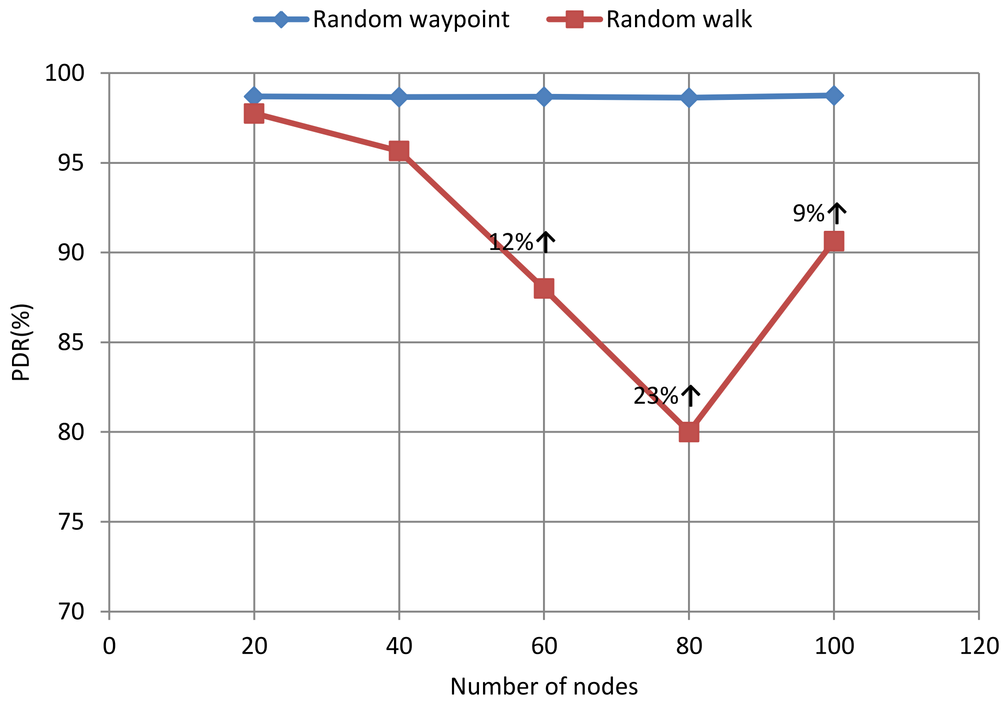


Fig. 7.8 Packet delivery ratio for different number of nodes

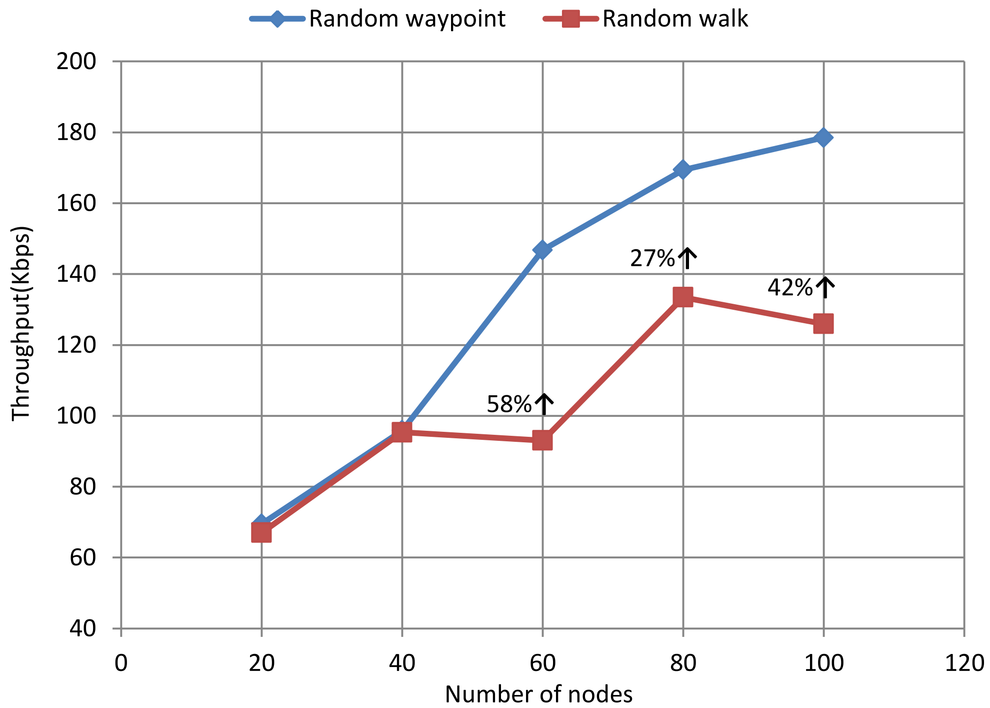


Fig. 7.9 Throughput for different number of nodes

Table 7-8 shows the comparison of the two investigated mobility models: Random Walk and Random Waypoint models when the number of nodes varies between 20 and 80 nodes. The proposed mobility prediction method MAODV has been used with these two models. The results of simulation indicate to 71% improvement in delay, 21% improvement in packet delivery ratio and 37% improvement in throughput of Random Waypoint compared to Random Walk.

Table 7-8 The comparison of the mobility models by varying number of nodes

	Random Walk	Random Waypoint	Improvement [%]
Average end to end delay [sec]	0.281±0.166	0.126±0.003	71%
Packet delivery ratio [%]	90.41±8.88	98.68±0.06	21%
Throughput [kbps]	102.96±33.22	132.02±54.51	37%

As it can be seen, Random Waypoint Mobility performs better as compared to Random Walk model, because the pause time in Random Waypoint Mobility model decreases the mobility and so as the path breakage which enhances the performance compared to Random Walk.

7.2.2 Effect of Varying Maximum Speed of Node Movement

A network area of 500 by 500 m^2 is built on the simulation platform and simulation is for 100 sec and 20 nodes. Since the mobile node is considered to be carried by people or by autonomous system, the simulation environment consists of different values of the maximum speed of a node movement which are 1.5, 5, 10, 15, 20, 25 m/sec.

Table 7-9 shows the values of performance metrics to evaluate the performance of network for Random Walk model. The values of performance metrics are given in Table 7-10 to evaluate the performance of network for Random Waypoint model. The comparison of Random Walk model and Random Waypoint model is shown in Fig. 7.10- Fig. 7.12 for the range of speed between 1.5 and 25 m/sec.

Table 7-9 The values of performance metrics of Random Walk model

Node max speed (m/sec)	Average end to end delay (sec)	Packet delivery ratio (%)	Throughput (Kbps)
1.5	0.150	97.57	67.01
5	0.124	98.75	63.79
10	0.217	48.90	62.98
15	0.183	48.75	61.14
20	0.195	33.75	61.66
25	0.190	33.75	61.57

Table 7-10 The values of performance metrics of Random Waypoint model

Node max speed (m/sec)	Average end to end delay (sec)	Packet delivery ratio (%)	Throughput (Kbps)
1.5	0.123	98.72	69.5
5	0.130	98.61	67.62
10	0.129	98.61	66.69
15	0.124	98.72	65.00
20	0.124	98.72	65.00
25	0.123	98.73	63.65

As shown in Fig. 7.10 average end to end delay of both mobility models. It shows that delay slightly increases for increasing the node speed of Random Walk model, while delay of Random Waypoint is remained unchanged and less as compared with the Random Walk model. For speed higher than 15m/sec, the delay of Random Waypoint is about 36% lower than the delay of Random Walk.

Based on the result of Fig. 7.11, the node with Random Waypoint has still better and constant packet delivery performance than the Random Walk which provide less packet delivery for speed higher than 5m/sec.

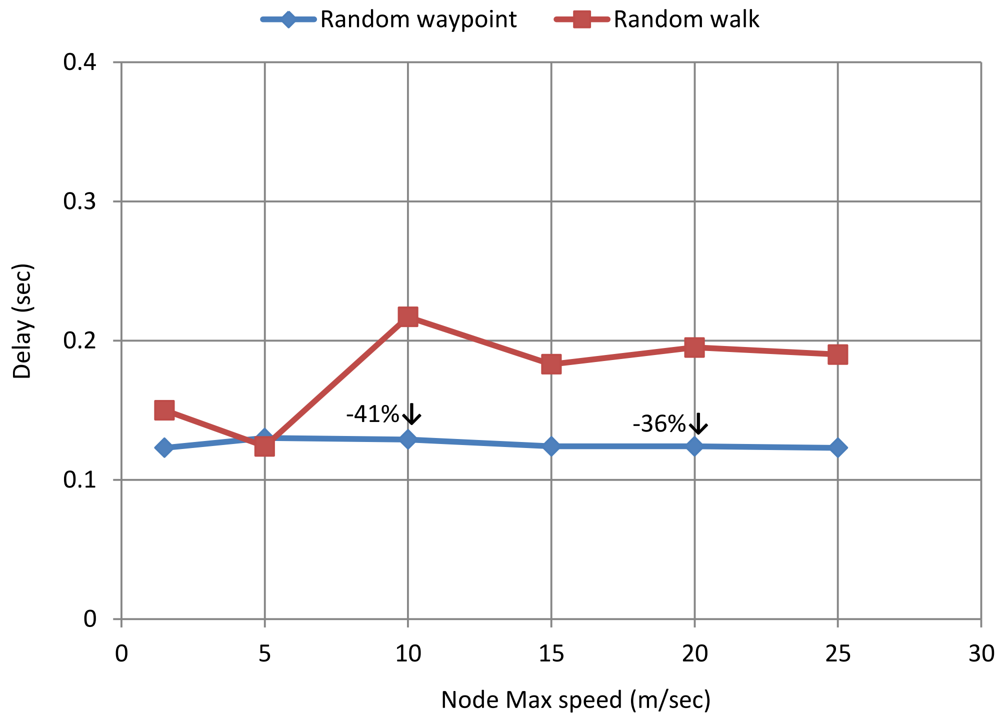


Fig. 7.10 Average end to end delay for different values of maximum speed

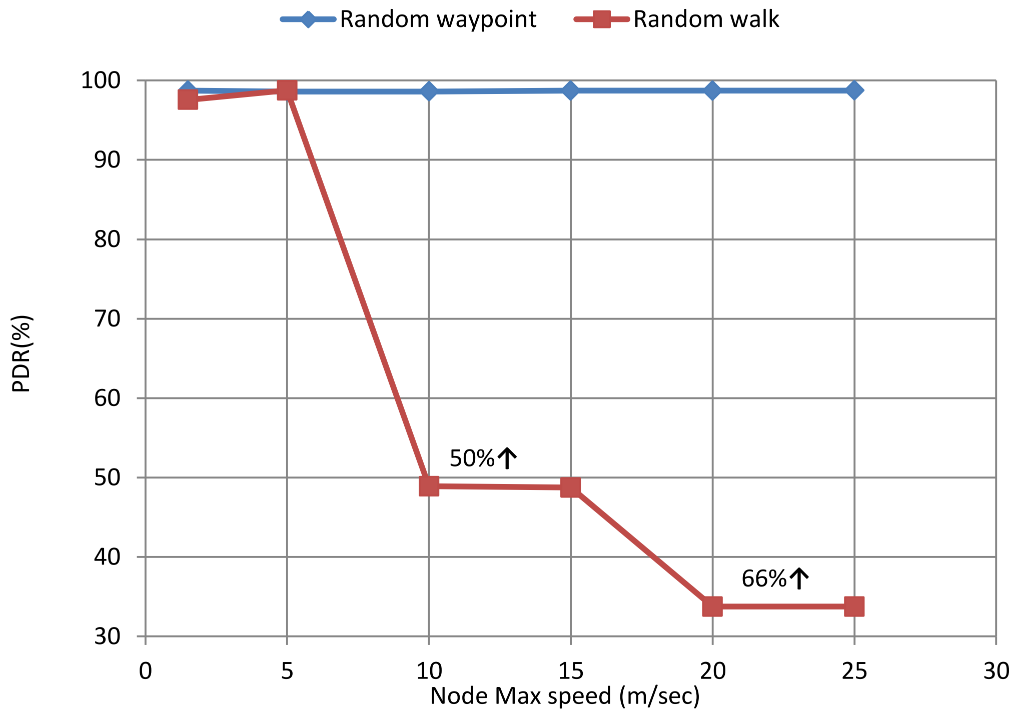


Fig. 7.11 Packet delivery ratio for different values of maximum speed

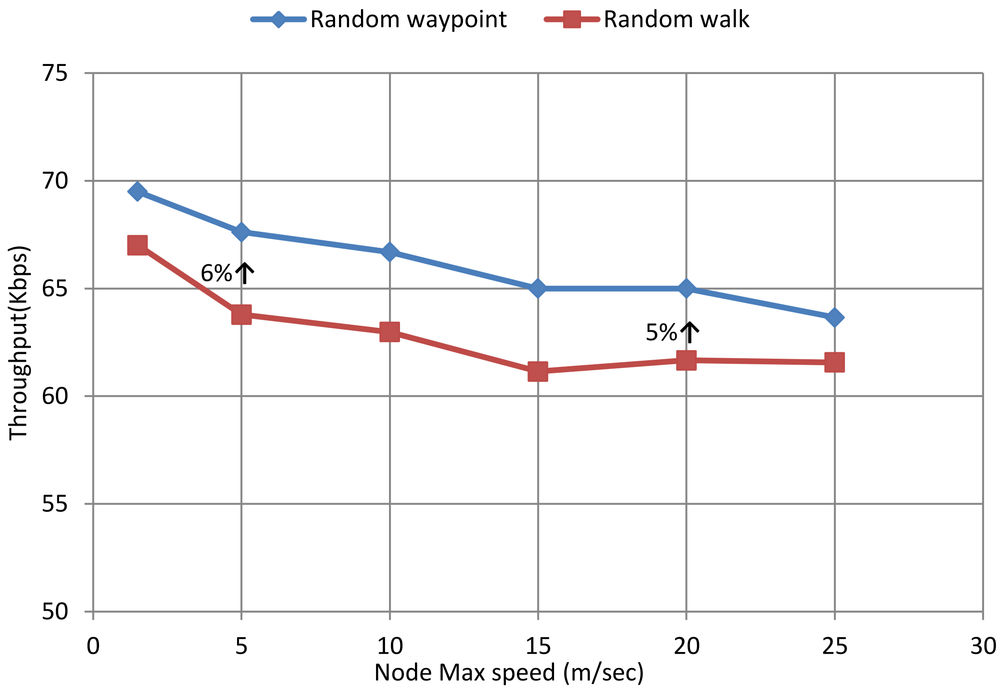


Fig. 7.12 Throughput for different values of maximum speed

In Fig. 7.12, Random Waypoint shows higher throughput than Random Walk. The throughput of Random Walk decreases when the speed of nodes increases. The throughput decreases for increasing the node speed of both mobility models.

Table 7-11 shows the comparison of the two investigated mobility models: Random Walk and Random Waypoint models when the value of the maximum speed of a node movement varies between 1.5 and 25 m/sec.

Table 7-11 The comparison of the mobility models for different values of speed

	Random Walk	Random Waypoint	Improvement [%]
Average end to end delay [sec]	0.177±0.047	0.126±0.004	6%
Packet delivery ratio [%]	60.25±7.58	98.69±0.06	87%
Throughput [kbps]	63.03±0.66	66.24±1.31	4%

The simulation results show 6% improvement in delay, 87% improvement in packet delivery ratio and 4% improvement in throughput of Random Waypoint compared to Random Walk. As it can be seen, Random Waypoint Mobility performs better as compared to Random Walk model, because the pause time in Random Waypoint Mobility model decreases the mobility and therefore ensures more stable link between neighbouring nodes.

7.3 Bayesian Neural Network Results

This section determines the behaviour of neural network based method which was described in section 4.5. The Bayesian technique is used for training neural network. Only the current position and speed are used to find the next position of MN. It has been implemented in Model Manager for training Bayesian neural networks.

Consider an ad hoc node moves according to Random Waypoint Mobility model within area of (500×500) m^2. The speed range is [$speed_{min}$-40m/s], $0 \leq speed_{min} \leq 40\text{m/s}$ (see Fig. 7.13). The coordinates and speed are saved every second, starting from 0 second to 200 seconds. Therefore 200 patterns are available. The first 150 patterns are used for training network and the rest for predicting. This model used the current coordinates and speed (x, y, s) as input variables, and the next coordinates as targets. The minimum and maximum of each input variable and the target data are shown in Table 7-12.

The Bayesian method can find the significance of each input, thus there is no need to exclude any variable prior to the analysis. The variables which have little effect in explaining the output will be linked to small weights.

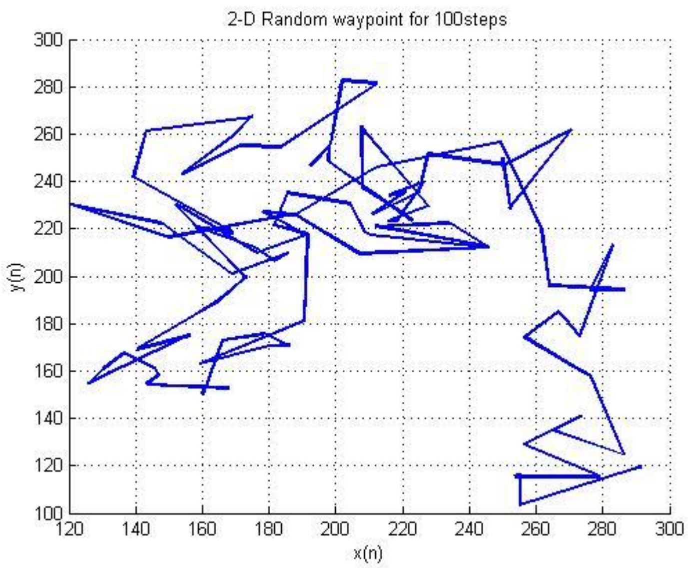


Fig. 7.13 Example of movement of an MN.

Table 7-12 The minimum and maximum values for the database

	Min	Max	MinNo	MaxNo	Average	StDev
x_n	34.6	285.2	136	6	135.8	47.47
y_n	97.6	430.4	54	105	251.7	75.69
s_n	0.3	40	1	59	19.1	11.73
Target x_{n+1}	34.6	285.2	135	5	134.8	46.82
Target y_{n+1}	97.6	430.4	53	104	251.2	75.98

After analysing and preparing the database, many submodels have been developed. Each submodel contained a set of parameters which defined the function that best fits the data with which the submodel has been developed. Each submodel had a given number of hidden units and different seed, thus different results were obtained by the particular submodel. Seeds are the initial weights which have been used as guesses. The time required to train a submodel grows exponentially with the number of hidden units (typically, training a submodel with one hidden unit took a few seconds, while many hours were required for 100 hidden units).

The number of hidden units is assumed in range of 1 to 20. In the training phase, one hundred networks were trained with hidden units ranging from one to twenty and five seeds in each case, as shown in Table B-1and Table B-2. Each submodel made a prediction differently and these submodels were sorted by decreasing LPE, the submodel with the highest log predictive error is the best one.

Following the mathematical theory of neural network described in section 6.2 and using (30) and (31), the optimal submodel is selected based on minimum test error or maximum log predictive error. The test error and log predictive error during the training phase in dependence on the number of hidden units are shown in Fig. 7.14 and Fig. 7.15 respectively. Fig. 7.15 shows the changing of LPE of x_{n+1} with the number of hidden units. The LPE has an optimum value at about one hidden unit, as the test error has a minimum value at one hidden unit, Fig. 7.14. The best submodel according to the LPE was then selected and tested on the output side. Test the submodel with one hidden unit and two seeds, see Fig. 7.16.

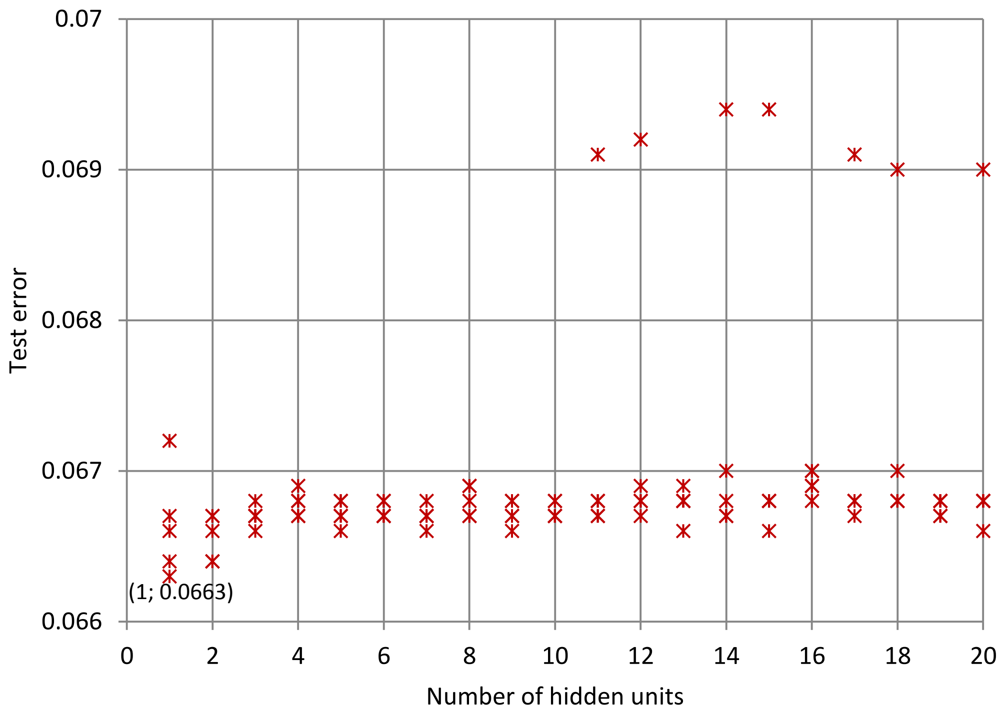


Fig. 7.14 The test error during the training phase

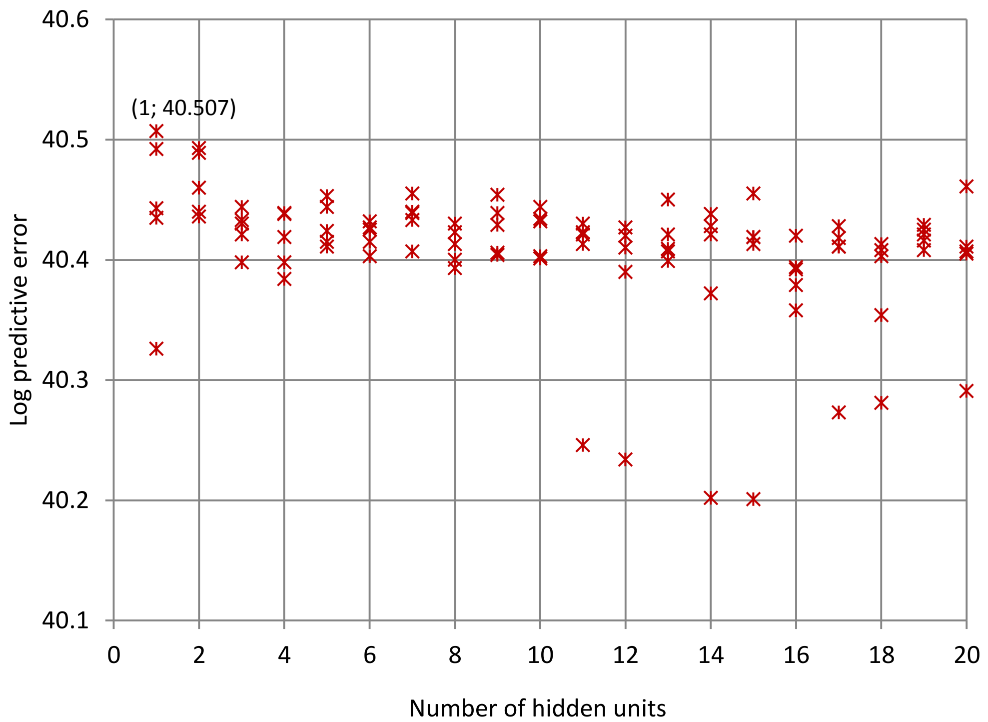


Fig. 7.15 The log predictive error during the training phase

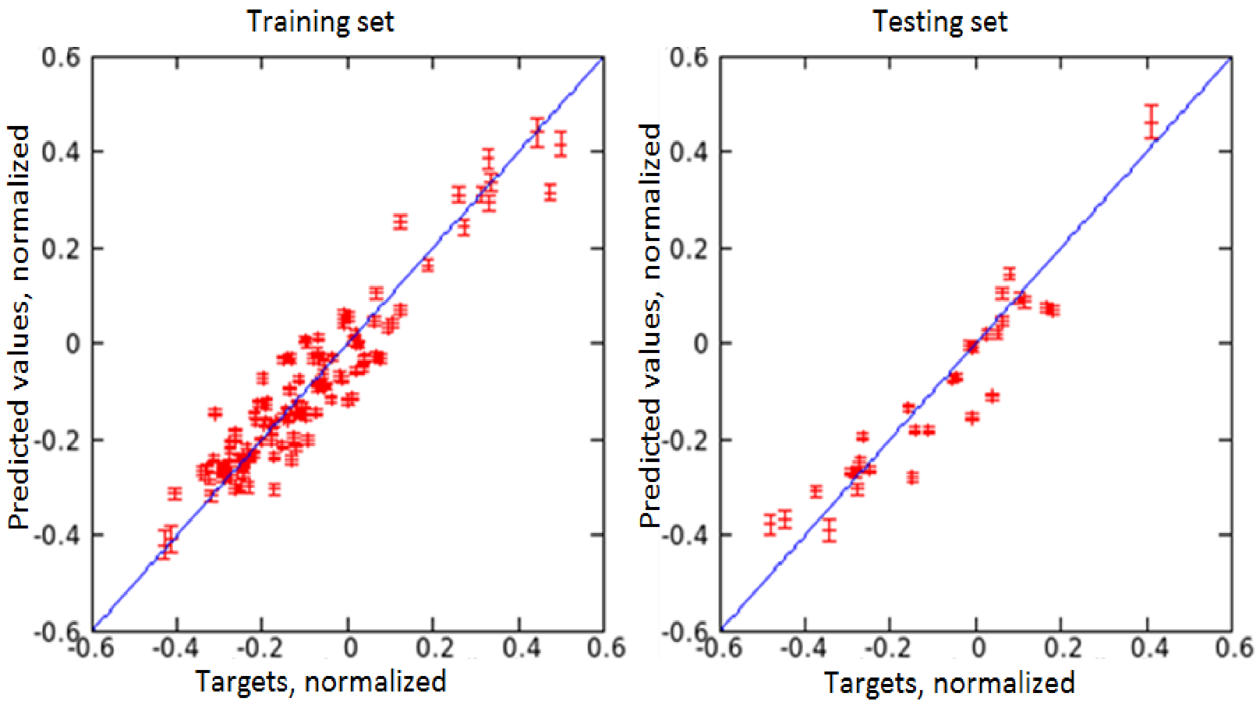


Fig. 7.16 The predictions by the best submodel for: a) training set, b) testing set

As mentioned before, the database is randomly divided into training and testing sets, to ensure that both the half used for training and testing contains similar information. For testing the best submodel, the Model Manager makes prediction using whole database (training set and testing set) as illustrated in Fig. 7.16 here in normalized values. These processes have been done during the training stage.

These submodels sometimes had domains in which they displayed good prediction, and others in which they did not. It is important to mention that a set of submodels with a given numbers of hidden units and different seed is called a model or a set of multiple submodels. Their predictions are combined to give the best overall result possible.

The final model has been built from a set of multiple submodels. The maximum number of submodels included in my sets equals to 20. The optimum number of submodels to form the set is determined based on the combined test error of all the members of the set. These methods were further attempts to find the appropriate level of complexity from the data, and to ensure a robust solution is found. The sets, which are built, are as many as there are submodels: the first set contains only the first submodel, the second set contains the two first submodels etc.

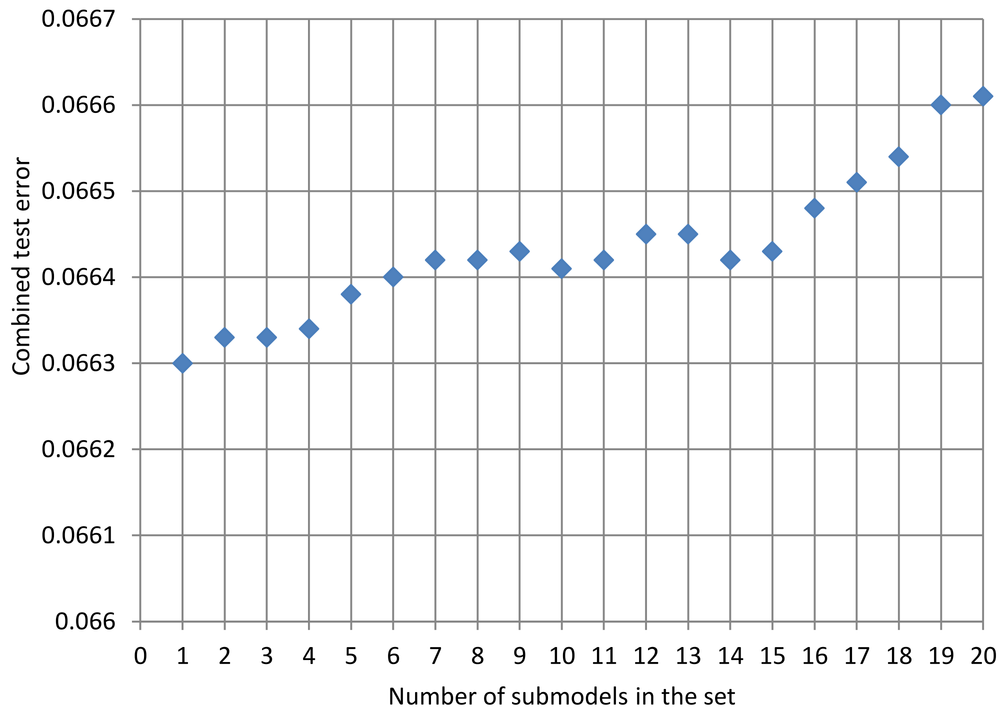


Fig. 7.17 The combined test error for the sets of different sizes

Usually, the error is minimized by using more than one submodel in the set. However in this case, the optimum number of submodels to form the set was found to be one as shown in Fig. 7.17. Therefore, the test error for the best submodel is 0.0663, and the combined test error of the set is 0.0663; this is test error estimated just for set containing only one submodel, and hence the test error for the best submodel and the combined test error of the set are the same.

The best way to evaluate the set of submodels is by making predictions and comparing predictions with target data. Fig. 7.18 shows a plot of target versus the predicted output using the selected set; predictions are made by using 50 patterns as input variables. Fig. 7.19 shows the prediction performance of the proposed model for Random Waypoint Mobility. It is clear that the universal approximation capability of the set enables them to track the nodes mobility once training is completed.

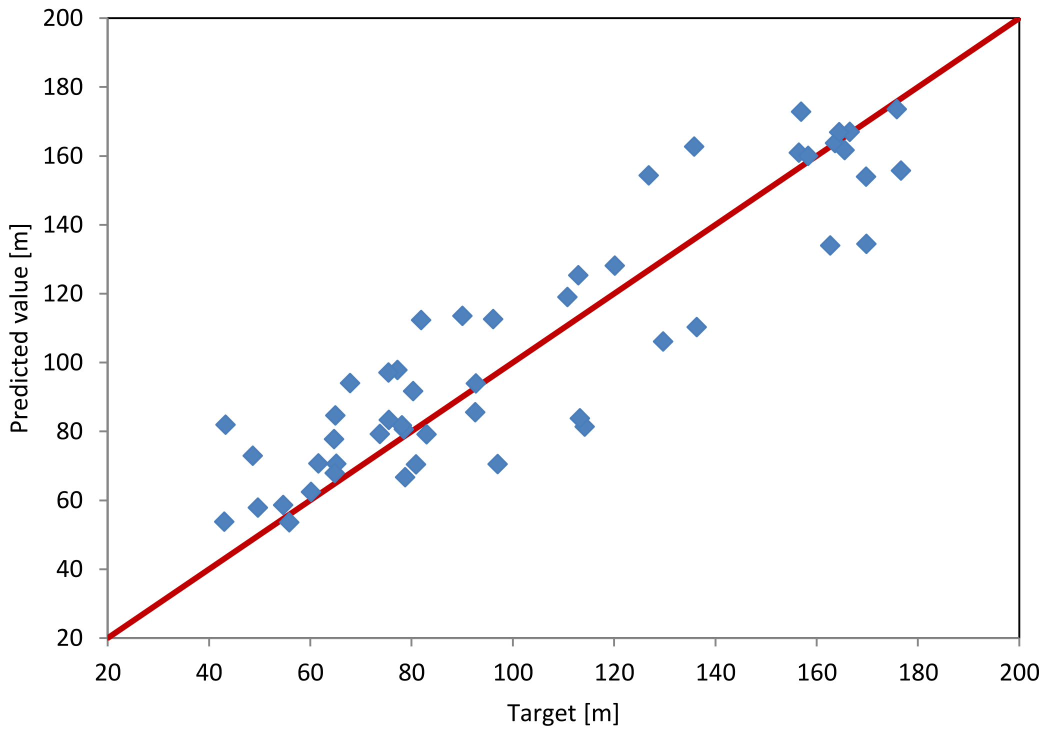


Fig. 7.18 Prediction of next position x_{n+1} in [m] against measured value in [m]

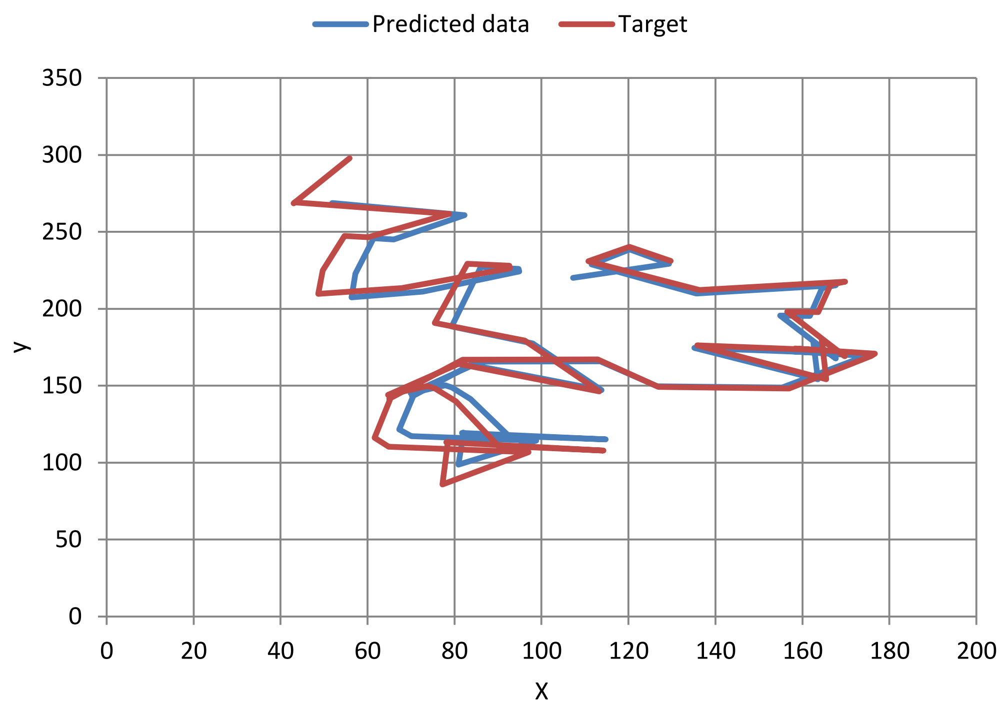


Fig. 7.19 The prediction result of Random Waypoint Mobility

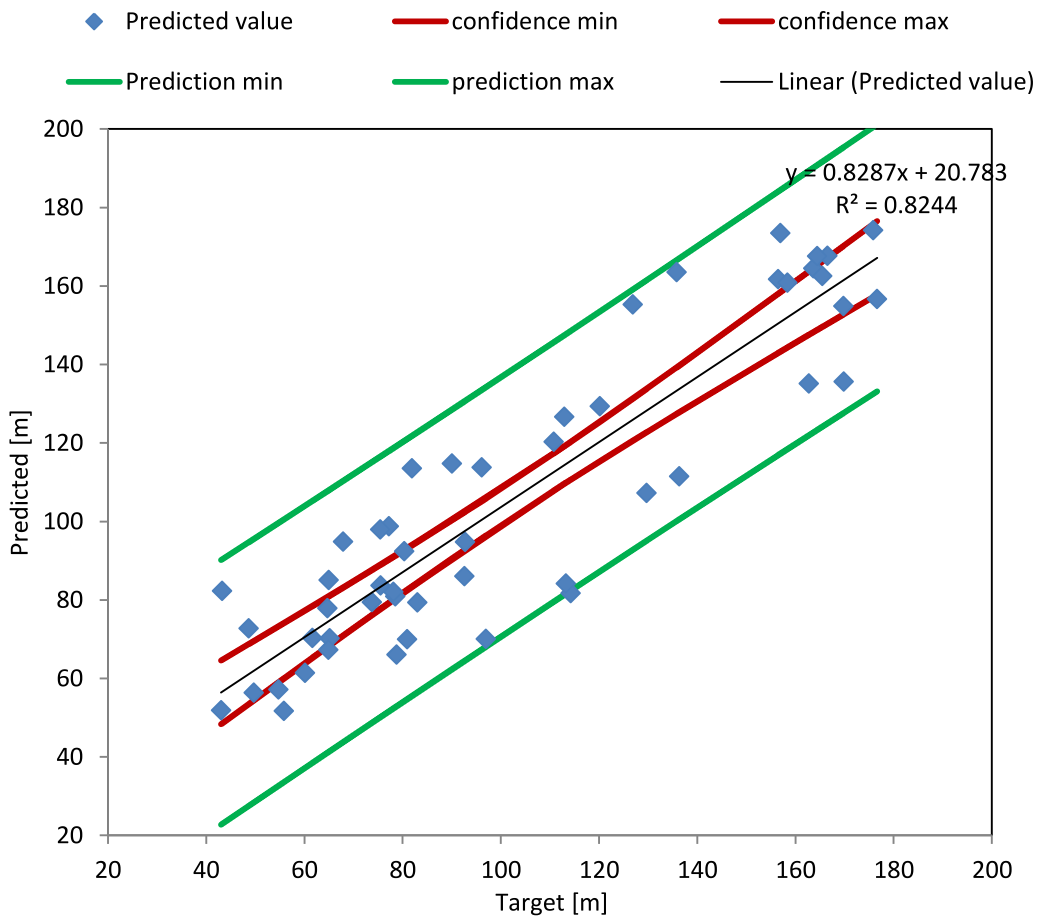


Fig. 7.20 The confidence and prediction bands of linear regression at confidence level 95%.

In Fig. 7.20, confidence and prediction intervals are used to characterize the results. It was performed the linear regression of data (predicted and measured). The linear regression line has an equation of the form:

$$y = 0.8287x + 20.783,$$
$$R^2 = 0.8244, \tag{38}$$

where y is the predicted value, x is the target value and R^2 is the correlation coefficient. The confidence level used is 95% around the average. This means that there is a 95% probability that the true linear regression line of the population will lie within the confidence interval of the regression line calculated from data. The confidence interval of the prediction presents a range for the mean rather than the distribution of individual data points. It does not tell the likely range of all values, just how much the average value is likely to fluctuate. The correlation of measured and predicted through the best model has been found acceptable.

The results of these predictions are shown in the Table C-3. Note that these data have not been included into the training and verification phase of the neural network model development.

7.4 Summary

This chapter presented the implementation of the proposed mobility prediction methods and the results. The mobility prediction method using virtual map has been implemented in NS-2. Both Random Walk model and RWP model were chosen to illustrate the performance metrics of the proposed mobility prediction algorithm. The performance metrics used are end to end delay, throughput, and packet delivery ratio.

Section 7.1 presented the improvement of the proposed method MAODV as compared with the traditional AODV routing protocol. Simulations evaluated proposed mobility prediction based on an end to end delay, throughput, and packet delivery. The MAODV performed better as compared to AODV. This is due to the fact that MAODV takes into account the next position of the destination to point the antenna to the predicted direction to send data. Therefore the probability of packet drop decreases, the delay decreases and the performance improves.

Section 7.2 investigates how the proposed mobility prediction method behaves with both mobility models (random waypoint, Random Walk). The simulation results were exported for different number of nodes and for different values of maximum speed. The chosen numbers of nodes were 20, 40, 60, 80 and 100 MNs. The values of maximum speed of a node movement were 1.5, 5, 10, 15, 20, 25 m/sec. The simulation results showed that Random Waypoint Mobility performs better as compared to Random Walk model. This is because of the pause time in Random Waypoint Mobility model, which decreases the mobility and so as the path breakage which enhances the performance compared to Random Walk.

In section 7.3, the neural network based method has been implemented in Model Manager for training Bayesian neural networks. The best model was selected and tested on the output side. The final model has been built from a set of multiple submodels. It has been found that the optimum number of submodels to form the set is one. The final model was used to make predictions of 50 patterns as input variables. The results of these predictions are shown in Table C-3. The prediction values were compared with target values. Confidence and prediction intervals are used to characterize the results. It has been found that the correlation of measured and predicted through the best model is acceptable.

8 CONCLUSION

The main goal of this thesis was to define mobility prediction methods for MANET networks: the first method is based on the virtual map and the second method uses a neural network.

In the virtual map based scheme, each node can build its virtual map of moving depending on its location over the time. Thereby, the MNs use this map to define the next step in the control message between the sending nodes. The predicted information about the expected movement of stations is used for finding the optimal path from the source to the destination. I evaluated the mobility prediction method for AODV routing protocol by comparison of differences in performance evaluation of the traditional AODV routing protocol and the MAODV which uses proposed mobility prediction method. Also Random Walk Mobility model and Random Waypoint Mobility model have been carried out for this comparison. Simulation results illustrated that the performance of the proposed method enhanced in term of delay, throughput and PDR for the range of number of nodes between 20 and 80. Table 8-1 summarizes the improvement of MAODV compared to AODV for Random Walk Mobility model and Random Waypoint Mobility model.

Table 8-1 Improvement of the proposed method MAODV

	Average end to end delay	Packet delivery ratio	Throughput
Improvement for Random walk	30%	13%	8%
Improvement for Random Waypoint	6%	8%	6%

Comparative results of the Random Walk Mobility model and Random Waypoint Mobility model have been carried out via NS-2 software simulations. These mobility models are represented in BonnMotion. I investigated existing mobility models, and how the prediction solution in this research can be applied to them. Simulations respectively realize performances in terms of average end to end delay, packet delivery ratio and network throughput under different mobility model. The proposed prediction concept is implemented over AODV routing protocol. Simulation results illustrated that the performance of the proposed method varies across two mobility models, number of nodes and speeds. The proposed method provides an adaptive location prediction mechanism. It proactively predicts future locations of communicating nodes and minimizes location updating, thereby reduces communication delay. From the simulation analysis I can conclude that Random Waypoint model is the best model which outperforms Random Walk model in term of end to end delay, throughput, and packet delivery ratio.

In the neural network based method, an ANN for movement prediction has been developed in MANETs. The Bayesian technique was used for training ANNs. It has been implemented in Model Manager for training Bayesian neural networks. Training a network includes finding a set of weights and biases which gives a trade-off between complexity and accuracy. The best way to evaluate the final model is done by making predictions and comparing predictions with target data. The predictions are made by using 50 patterns which have not been included into the training phase. I found out that the universal approximation capability of the set enables them to track the nodes mobility once training is completed. The Correlation of measured and predicted through the best model has been found acceptable.

REFERENCES

[1] M. Conti and S. Giordano, ‘Mobile Ad Hoc Networking: Milestones, Challenges, and New Research Directions’, *IEEE Commun. Mag.*, vol. 52, no. 1, pp. 85–96, Jan. 2014.

[2] M. Conti, C. Boldrini, S. S. Kanhere, E. Mingozzi, E. Pagani, P. M. Ruiz, and M. Younis, ‘From MANET to People-Centric Networking: Milestones and Open Research Challenges’, *Comput. Commun.*, vol. 71, no. 1, pp. 1–21, Nov. 2015.

[3] J. M. Garcia-Campos, D. G. Reina, S. L. Toral, N. Bessis, F. Barrero, E. Asimakopoulou, and R. Hill, ‘Performance Evaluation of Reactive Routing Protocols for VANETs in Urban Scenarios Following Good Simulation Practices’, in *2015 9th International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing*, 2015, pp. 1–8.

[4] M. Kumar and R. Mishra, ‘An Overview of MANET : History , Challenges and Applications’, *Indian J. Comput. Sci. Eng.*, vol. 3, no. 1, pp. 121–125, 2012.

[5] A. Bang and P. Ramteke, ‘MANET : History,Challenges And Applications’, *Int. J. Appl. or Innov. Eng. Manag.*, vol. 2, no. 9, pp. 249–251, 2013.

[6] N. Makhlouf and P. Vajsar, ‘MAC Protocols in Mobile Ad Hoc Networks’, *Int. J. Adv. Telecommun. Electrotech. Signals Syst.*, vol. 1, no. 1, pp. 9–12, 2012.

[7] N. Makhlouf and P. Vajsar, ‘MAC Protocls for MANET Networks with Directional Antennas’, in *Conference of Informatics and Management Sciences*, 2013, pp. 301–304.

[8] S.S.Dhenakaran and A.Parvathavarthini, ‘An Overview of Routing Protocols in Mobile Ad-Hoc Network’, *Int. J. Adv. Res. Comput. Sci. Softw. Eng. An Overv. Routing Protoc. Mob. Ad-Hoc Netw.*, vol. 3, no. 2, pp. 251–259, 2013.

[9] N. Javaid, Z. A. Khan, U. Qasim, M. Jamil, M. Ishfaq, and T. A. Alghamdi, ‘Modeling Routing Overhead of Reactive Protocols at Link Layer and Network Layer in Wireless Multihop Networks’, *Math. Probl. Eng.*, vol. 2015, pp. 1–14, 2015.

[10] R. Ahuja, A. B. Ahuja, and P. Ahuja, ‘Performance Evaluation and Comparison of AODV and DSR Routing Protocols in MANETs under Wormhole Attack’, in *Second International Conference on Image Information Processing (ICIIP-2013)*, 2013, pp. 699–702.

[11] A. Boukerche, B. Turgut, N. Aydin, M. Z. Ahmad, L. Bölöni, and D. Turgut, 'Routing Protocols in Ad Hoc Networks: a Survey', *Comput. Networks,* vol. 55, no. 13, pp. 3032–3080, 2011.

[12] Y. Zhang, T. Yan, J. Tian, Q. Hu, G. Wang, and Z. Li, 'TOHIP: A Topology-Hiding Multipath Routing Protocol in Mobile Ad Hoc Networks', *Ad Hoc Networks,* vol. 21, pp. 109–122, 2014.

[13] F. Bai and A. Helmy, 'A Survey of Mobility Models in Wireless Adhoc Networks', *Wirel. Ad Hoc Sens. Networks*, pp. 1–30, 2004.

[14] J. Hosek, K. Molnar, and P. Vajsar, 'Map-Based Direct Position Control System for Wireless Ad-Hoc Networks', *Telecommun. Syst.*, vol. 55, no. 4, pp. 537–549, 2014.

[15] A. Sharma and E. J. Singh, 'Mobility Models for Manet : Mathematical Perspective', *Int. J. Adv. Res. Eng. Appl. Sci.,* vol. 2, no. 5, pp. 59–68, 2013.

[16] S. Kaur, M. T. Student, and C. Sharma, 'An Overview of Mobile Ad hoc Network : Application , Challenges and Comparison of Routing Protocols', vol. 11, no. 5, pp. 7–11, 2013.

[17] A. Mchergui, T. Moulahi, B. Alaya, and S. Nasri, 'A Survey and Comparative Study of QoS Aware Broadcasting Techniques in VANET', *Telecommun. Syst.,* vol. 66, no. 2, pp. 253–281, Oct. 2017.

[18] B. P. Maratha, T. R. Sheltami, and K. Salah, 'Performance Study of MANET Routing Protocols in VANET', *Arab. J. Sci. Eng.,* vol. 42, no. 8, pp. 3115–3126, Aug. 2017.

[19] P. Ramachandran and M. Dinakaran, 'Signal Strength and Residual Power Based Optimum Transmission Power Routing for Mobile Ad hoc Networks', *Procedia Comput. Sci.,* vol. 92, pp. 168–174, 2016.

[20] X. Lu, D. Towsley, P. Lio, and Z. Xiong, 'An Adaptive Directional Mac Protocol for Ad Hoc Networks Using Directional Antennas', *Sci. China Inf. Sci.*, vol. 55, no. 6, pp. 1360–1371, 2012.

[21] H. Zhai and Y. Fang, 'Medium Access Control Protocols in Mobile Ad Hoc Networks : Problems and Solutions 1 Department of Electrical and Computer Engineering', *Handb. Theor. Algorithmic Asp. Sensor, Ad Hoc Wirel. Peer-to-Peer Networks*, pp. 1–31, 2005.

[22] A. Azari, J. S. Harsini, and F. Lahouti, ‘Performance Analysis of Ad-Hoc Routing in Heterogeneous Clustered Multi-Hop Wireless Networks’, *Comput. Networks*, vol. 80, pp. 143–154, Apr. 2015.

[23] M. K. Denko, ‘Mobility Prediction Schemes in Wireless Ad Hoc Networks’, in *Advanced Wired and Wireless Networks*, Boston: Kluwer Academic Publishers, 2001, pp. 171–186.

[24] F. Yu and V. Leung, ‘Mobility-Based Predictive Call Admission Control and Bandwidth Reservation in Wireless Cellular Networks’, *Comput. Networks*, vol. 38, pp. 577–589, 2002.

[25] J. Ananthi and V. Ranganathan, ‘A Review : On Mobility Prediction for Wireless Networks’, *Int. J. Emerg. Technol. Adv. Eng.*, vol. 3, no. 4, pp. 891–902, 2013.

[26] A. ur Rehman Khan, S. M. Bilal, and M. Othman, ‘A Performance Comparison of Open Source Network Simulators for Wireless Networks’, in *2012 IEEE International Conference on Control System, Computing and Engineering*, 2012, pp. 34–38.

[27] B. Hawelka, I. Sitko, P. Kazakopoulos, and E. Beinat, ‘Collective Prediction of Individual Mobility Traces for Users with Short Data History’, *PLoS One*, vol. 12, no. 1, p. e0170907, Jan. 2017.

[28] M. Papandrea and S. Giordano, ‘Location Prediction and Mobility Modelling for Enhanced Localization Solution’, *J. Ambient Intell. Humaniz. Comput.*, vol. 5, no. 3, pp. 279–295, Jun. 2014.

[29] S. Suraj, R. Tapaswi, S. Yousef, K. K. Pattanaik, and M. Cole, ‘Mobility Prediction in Mobile Ad Hoc Networks Using A Lightweight Genetic Algorithm’, *Wirel. Networks*, vol. 22, no. 6, pp. 1797–1806, 2016.

[30] W. Creixell and K. Sezaki, ‘Mobility Prediction Algorithm for Mobile Ad Hoc Network Using Pedestrian Trajectory Data’, in *2004 IEEE Region 10 Conference TENCON 2004.*, 2004, vol. B, no. 1, pp. 668–671.

[31] T. T. Son, H. Le Minh, G. Sexton, N. Aslam, and Z. Ghassemlooy, ‘Bayesian Model for Mobility Prediction to Support Routing in Mobile Ad-Hoc Networks’, in *2013 IEEE 24th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC)*, 2013, pp. 3186–3190.

[32] H. Abu-Ghazaleh and A. Alfa, ‘Application of Mobility Prediction in Wireless

Networks Using Markov Renewal Theory', *IEEE Trans. Veh. Technol.*, vol. 59, no. 2, pp. 788–802, 2010.

[33] L. Ghouti, 'Mobility Prediction in Mobile Ad Hoc Networks Using Neural Learning Machines', *Simul. Model. Pract. Theory*, vol. 66, pp. 104–121, Aug. 2016.

[34] C. Comito, D. Falcone, and D. Talia, 'Mining Human Mobility Patterns from Social Geo-tagged Data', *Pervasive Mob. Comput.*, vol. 33, pp. 91–107, Dec. 2016.

[35] T. K. Vu and S. Kwon, 'On-Demand Routing Algorithm with Mobility Prediction in the Mobile Ad-hoc Networks', *Netw. Internet Archit.*, p. 10, Sep. 2016.

[36] Y. Cong, X. Zhou, and R. A. Kennedy, 'Interference Prediction in Mobile Ad Hoc Networks With a General Mobility Model', *IEEE Trans. Wirel. Commun.*, vol. 14, no. 8, pp. 4277–4290, Aug. 2015.

[37] T. Camp, J. Boleng, and V. Davies, 'A Survey of Mobility Models for Ad Hoc Network Research', *Wirel. Commun. Mob. Comput.*, vol. 2, no. 5, pp. 483–502, Aug. 2002.

[38] S. Madi and H. Al-Qamzi, 'A survey on realistic mobility models for Vehicular Ad Hoc Networks (VANETs)', in *10th IEEE International Conference on Networking, Sensing and Control (ICNSC)*, 2013, pp. 333–339.

[39] C. Y. Aung, B. C. Seet, M. Zhang, L. F. Xie, and P. H. J. Chong, 'A Review of Group Mobility Models for Mobile Ad Hoc Networks', *Wirel. Pers. Commun.*, vol. 85, no. 3, pp. 1317–1331, Dec. 2015.

[40] R. R. Roy, 'Random Walk Mobility', in *Handbook of Mobile Ad Hoc Networks for Mobility Models*, 2010, pp. 35–63.

[41] Y. Cai, X. Wang, Z. Li, and Y. Fang, 'Delay and Capacity in MANETs under Random Walk Mobility Model', *Wirel. Networks*, vol. 20, no. 3, pp. 525–536, Apr. 2014.

[42] J. Ariyakhajorn, P. Wannawilai, and C. Sathitwiriyawong, 'A Comparative Study of Random Waypoint and Gauss-Markov Mobility Models in the Performance Evaluation of MANET', *2006 Int. Symp. Commun. Inf. Technol. Isc.*, pp. 894–899, 2006.

[43] R. R. Colletti and R. M. de Moraes, 'Evaluation of Link Lifetime for the Random Waypoint Mobility Model', in *2013 19th IEEE International Conference on Networks (ICON)*, 2013, no. 1, pp. 1–5.

[44] W. Navidi and T. Camp, ‘Stationary Distributions for the Random Waypoint Mobility Model’, *IEEE Trans. Mob. Comput.*, vol. 3, no. 1, pp. 99–108, 2004.

[45] S. J. Doyle, T. K. Forde, and L. E. Doyle, ‘Spatial Stationarity of Link Statistics in Mobile Ad Hoc Network Modelling’, in *14th IEEE International Symposium on Modeling, Analysis, and Simulation*, 2006, pp. 43–50.

[46] C. Engineering, ‘Effects of Mobility in Hierarchical Mobile Ad Hoc Networks’, in *Consumer Communications and Networking Conference*, 2009, pp. 3–7.

[47] P. S. Prasad, ‘Mobility Modeling, Prediction and Resource Allocation in Wireless Networks’, Auburn University, 2010.

[48] R. Roy, ‘Realistic Random Direction Mobility’, in *Handbook of Mobile Ad Hoc Networks for Mobility Models*, 2010, pp. 223–244.

[49] E. Royer, M. Melliar-smith, and E. Moser, ‘An Analysis of the Optimum Node Density for Ad hoc Mobile Networks’, in *IEEE International Conference on Communications*, 2001, pp. 857–861.

[50] X. Zhang, L. Wu, Y. Zhang, and D. K. Sung, ‘Interference Dynamics in MANETs with a Random Direction Node Mobility Model’, in *IEEE Wireless Communications and Networking Conference (WCNC)*, 2013, pp. 3788–3793.

[51] X. Hong, M. Gerla, G. Pei, and C.-C. Chiang, ‘A Group Mobility Model for Ad Hoc Wireless Networks’, in *Proceedings of the 2nd international workshop on Modeling, analysis and simulation of wireless and mobile systems - MSWiM ’99*, 1999, pp. 53–60.

[52] J. Liu, N. Kato, J. Ma, and T. Sakano, ‘Throughput and Delay Tradeoffs for Mobile Ad Hoc Networks with Reference Point Group Mobility’, *IEEE Trans. Wirel. Commun.*, vol. 14, no. 3, pp. 1266–1279, Mar. 2015.

[53] D. Kumar, A. Srivastava, and S. C. Gupta, ‘Routing in Ad Hoc Networks under Reference Point Group Mobility’, in *2013 European Modelling Symposium*, 2013, pp. 595–598.

[54] D. Son, S. Member, and A. Helmy, ‘The Effect of Mobility-Induced Location Errors on Geographic Routing in Mobile Ad Hoc and Sensor Networks : Analysis and Improvement Using Mobility Prediction’, *Ieee Journals Mag.*, vol. 3, no. 3, pp. 233–245, 2004.

[55] M. Yu, A. Malvankar, W. Su, and S. Y. Foo, ‘A link Availability-Based QoS-Aware Routing Protocol for Mobile Ad Hoc Sensor Networks’, *Comput. Commun.*, vol. 30, no. 18, pp. 3823–3831, Dec. 2007.

[56] A. Venkateswaran, V. Sarangan, T. F. La Porta, and R. Acharya, ‘A Mobility-Prediction-Based Relay Deployment Framework for Conserving Power in MANETs’, *IEEE Trans. Mob. Comput.*, vol. 8, no. 6, pp. 750–765, 2009.

[57] Y. Hu and G. Leus, ‘Self-Estimation of Path-Loss Exponent in Wireless Networks and Applications’, *IEEE Trans. Veh. Technol.*, vol. 64, no. 11, pp. 5091–5102, 2015.

[58] Z. Zhou, Z. Peng, J. H. Cui, Z. Shi, and A. Bagtzoglou, ‘Scalable Localization with Mobility Prediction for Underwater Sensor Networks’, *IEEE Trans. Mob. Comput.*, vol. 10, no. 3, pp. 335–348, 2011.

[59] N. Nowsheen, G. Karmakar, and J. Kamruzzaman, ‘MPDF: Movement Predicted Data Forwarding Protocol for Underwater Acoustic Sensor Networks’, in *Proceedings of the 20th Asia-Pacific Conference on Communication, APCC 2014*, 2015, pp. 100–105.

[60] P. Bhuvaneswari, S. Karthikeyan, B. Jeeva, and M. A. Prasath, ‘An Efficient Mobility Based Localization in Underwater Sensor Networks’, in *Proceedings - 4th International Conference on Computational Intelligence and Communication Networks*, 2012, pp. 90–94.

[61] J. Yan, X. Zhang, X. Luo, Y. Wang, C. Chen, and X. Guan, ‘Asynchronous Localization with Mobility Prediction for Underwater Acoustic Sensor Networks’, *IEEE Trans. Veh. Technol.*, vol. PP, no. 99, pp. 1–12, 2017.

[62] F. Somaa, C. Adjih, I. El Korbi, and L. A. Saidane, ‘A Bayesian Model for Mobility Prediction in Wireless Sensor Networks’, in *2016 International Conference on Performance Evaluation and Modeling in Wired and Wireless Networks (PEMWN)*, 2016, pp. 1–7.

[63] L. Dekar and H. Kheddouci, ‘A Cluster Based Mobility Prediction Scheme for Ad Hoc Networks’, *Ad Hoc Networks*, vol. 6, no. 2, pp. 168–194, Apr. 2008.

[64] H. Kaaniche and F. Kamoun, ‘Mobility Prediction in Wireless Ad Hoc Networks using Neural Networks’, *J. Telecommun.*, vol. 2, no. 1, pp. 95–101, Apr. 2010.

[65] A. Rhim and Z. Dziong, ‘Routing Based on Link Expiration Time for Manet Performance Improvement’, in *Communications (MICC), 2009 IEEE 9th Malaysia*

International Conference on Communications, 2009, no. 9, pp. 555–560.

[66] I. Ahmed, K. E. Tepe, and B. K. Singh, 'Reliable Coverage Area Based Link Expiration Time (LET) Routing Metric for Mobile Ad Hoc Networks', *Ad Hoc Networks*, vol. 28, pp. 466–476, 2010.

[67] P. Singh, 'ODSRP-LET : On-Demand Secure Routing Protocol based on Link Expiration Time', *Int. J. Comput. Appl.*, vol. 100, no. 13, pp. 14–23, 2014.

[68] W. Su, S. Lee, and M. Gerla, 'Mobility Prediction and Routing in Ad Hoc Wireless Networks', *Int. J. Netw. Manag.*, vol. 11, no. 1, pp. 3–30, 2001.

[69] C. Doss, A. Jennings, and N. Shenoy, 'A Review of Current Mobility Prediction Techniques for Ad Hoc Networks', in *The Fourth IASTED International Multi-Conference*, 2004, pp. 536–542.

[70] S. Jiang, D. He, and J. Rao, 'A Prediction-Based Link Availability Estimation for Mobile Ad Hoc Networks', in *IEEE Computer and Communications Societies*, 2001, pp. 1745–1752.

[71] A. Mcdonaldllitt and T. Znatitpll, 'Predicting Node Proximity in Ad-Hoc Networks: A Least Overhead Adaptive Model for Selecting Stable Routes', in *Mobile and Ad Hoc Networking and Computing*, 2000, pp. 29–33.

[72] H. Sathyaraj and C. Doss, 'Route Maintenance using Mobility Prediction for Mobile Ad hoc Networks', in *IEEE International Conference on Mobile Adhoc and Sensor Systems*, 2005, p. 6.

[73] K. Wang and B. Li, 'Group Mobility and Partition Prediction in Wireless Ad-Hoc Networks', in *IEEE International Conference on Communications*, 2002, pp. 1017–1021.

[74] M. Tarasov and J. Seitz, 'A Network Partitioning Recovery Process in Mobile Ad-Hoc Networks', in *IEEE 7th International Conference on Wireless and Mobile Computing, Networking and Communications*, 2011, pp. 32–36.

[75] R. Begleiter, R. El-Yaniv, and G. Yona, 'On Prediction using Variable Order Markov Models', *J. Artif. Intell. Res.*, vol. 22, pp. 385–421, 2004.

[76] S. Gambs, M.-O. Killijian, and M. Núñez Del Prado Cortez, 'Next Place Prediction using Mobility Markov Chains', in *the First Workshop on Measurement, Privacy, and*

Mobility, 2012, pp. 0–5.

[77] A. Hadachi, O. Batrashev, A. Lind, G. Singer, and E. Vainikko, ‘Cell Phone Subscribers Mobility Prediction Using Enhanced Markov Chain Algorithm’, in *IEEE Intelligent Vehicles Symposium (IV)*, 2014, no. 4, pp. 1049–1054.

[78] F. Lassabe and D. Charlet, ‘Predictive Mobility Models based on Kth Markov Models’, in *Computer Performance Engineering*, 2006, pp. 303–306.

[79] S. Bellahsene and L. Kloul, ‘A New Markov-Based Mobility Prediction Algorithm for Mobile Networks’, in *Computer Performance Engineering*, vol. 6342, 2010, pp. 37–50.

[80] Y. Li, D. Jin, Z. Wang, P. Hui, L. Zeng, and S. Chen, ‘A Markov Jump Process Model for Urban Vehicular Mobility: Modeling and Applications’, *IEEE Trans. Mob. Comput.*, vol. 13, no. 9, pp. 1911–1926, Sep. 2014.

[81] A. Asahara, K. Maruyama, A. Sato, and K. Seto, ‘Pedestrian-Movement Prediction Based on Mixed Markov-Chain Model’, in *Proceedings of the 19th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*, 2011, p. 25.

[82] M. Ni, Z. Zhong, and D. Zhao, ‘MPBC: A Mobility Prediction-Based Clustering Scheme for Ad Hoc Networks’, *IEEE Trans. Veh. Technol.*, vol. 60, no. 9, pp. 4549–4559, Nov. 2011.

[83] N. Makhlouf, ‘Exploiting Neural Networks for Mobility Prediction in Mobile Ad Hoc Networks’, *Elektrorevue J.*, vol. 18, no. 3, pp. 63–67, 2016.

[84] D. Macone, G. Oddi, and A. Pietrabissa, ‘Ad Hoc Networks MQ-Routing : Mobility- , GPS- and Energy-Aware Routing Protocol in MANETs for Disaster Relief Scenarios’, *Ad Hoc Networks*, vol. 11, no. 3, pp. 861–878, 2013.

[85] R. Fernandes and R. D’Souza G. L., ‘A New Approach to Predict user Mobility Using Semantic Analysis and Machine Learning’, *J. Med. Syst.*, vol. 41, no. 188, p. 12, Dec. 2017.

[86] S. Saeed, K. Jumari, M. Ismail, and A. Al-Hemyari, ‘Challenges and Solutions of QoS Provisioning for Real Time Traffic in Mobile Ad Hoc Networks’, in *International Conference on Computer and Information Science, ICCIS*, 2012, vol. 2, pp. 765–770.

[87] J. E. Macker and M. S. Corson, ‘Mobile Ad Hoe Networrking and the IETF’, *Mob.*

Comput. Commun. Rev., vol. 2, no. 1, pp. 9–14, 1998.

[88] R. Asokan, 'A review of Quality of Service (QoS) Routing Protocols for Mobile Ad Hoc Networks', in *International Conference on Wireless Communication and Sensor Computing (ICWCSC)*, 2010, pp. 1–6.

[89] N. Makhlouf, Ľ. Nagy, and P. Vajsar, 'Prediction of Movement of Wireless Nodes in Mobile Ad-hoc Networks', in *4th IEEE Intl' Conference on Cognitive Infocommunicaitons*, 2013, pp. 115–120.

[90] T. Petig, E. M. Schiller, and P. Tsigas, 'Self-stabilizing TDMA Algorithms for Wireless Ad-hoc Networks without External Reference', in *2014 13th Annual Mediterranean Ad Hoc Networking Workshop (MED-HOC-NET)*, 2014, pp. 87–94.

[91] L. Chen and W. Heinzelman, 'A Survey of Routing Protocols that Support QoS in Mobile Ad Hoc Networks', *IEEE Netw.*, vol. 21, no. 6, pp. 30–38, 2007.

[92] D. Vir, S. K. Agarwal, and S. A. Imam, 'Quantitative Analyses and Evaluation of MANET Routing Protocols in Effect of Varying Mobility Model using Qual-Net Simulator', in *the World Congress on Information and Communication Technologies*, 2012, pp. 915–921.

[93] G. A. Walikar and R. C. Biradar, 'A Survey on Hybrid Routing Mechanisms in Mobile Ad Hoc Networks', *J. Netw. Comput. Appl.*, vol. 77, pp. 48–63, 2017.

[94] A. Swidan, H. B. Abdelghany, R. Saifan, and Z. Zilic, 'Mobility and Direction Aware Ad-hoc on Demand Distance Vector Routing Protocol', *Procedia Comput. Sci.*, vol. 94, pp. 49–56, 2016.

[95] V. Balaji and V. Duraisamy, 'Ant Optimized Link Quality for Ad Hoc on Demand Distance Vector', *Wirel. Pers. Commun.*, vol. 79, no. 1, pp. 763–771, 2014.

[96] N. Javaid, A. Bibi, A. Javaid, and S. A. Malik, 'Modeling Routing Overhead Generated by Wireless Proactive Routing Protocols', in *2011 IEEE GLOBECOM Workshops (GC Wkshps)*, 2011, pp. 1072–1076.

[97] P. Vajsar, N. Makhlouf, J. Hošek, M. Bartl, P. Morávek, and R. Figurny, *Implementation of OLSR-Based QoS Support into MANET*. 2012.

[98] L. Liu, L. Zhu, L. Lin, and Q. Wu, 'Improvement of AODV Routing Protocol with QoS Support in Wireless Mesh Networks', *Phys. Procedia*, vol. 25, pp. 1133–1140,

2012.

[99] A. A. Al-khatib and R. Hassan, ‘Performance Evaluation of AODV, DSDV, and DSR Routing Protocols in MANET Using NS-2 Simulator’, in *International Conference of Reliable Information and Communication Technology*, 2018, pp. 276–284.

[100] N. Rathod and N. Dongre, ‘MANET Routing Protocol Performance for Video Streaming’, in *2017 International Conference on Nascent Technologies in Engineering (ICNTE)*, 2017, pp. 1–5.

[101] S. Vanthana and V. S. J. Prakash, ‘Comparative Study of Proactive and Reactive AdHoc Routing Protocols Using Ns2’, in *2014 World Congress on Computing and Communication Technologies*, 2014, pp. 275–279.

[102] S. Yang, R. He, Y. Wang, S. Li, and B. Lin, ‘OPNET-based Modeling and Simulations on Routing Protocols in VANETs with IEEE 802.11p’, in *2nd International Conference on Systems and Informatics*, 2015, pp. 536–541.

[103] R. Khan, A. M. N. Azad S., and A. Vishwakarma, ‘Enhancement of MANET Routing Protocol’, in *2014 Conference on IT in Business, Industry and Government (CSIBIG)*, 2014, pp. 1–6.

[104] P. Vajsar, P. Masek, J. Hosek, N. Makhloufu, and J. Leu, ‘Design of QoS Model for Mobile Ad-hoc Network’, in *2013 36th International Conference on Telecommunications and Signal Processing (TSP)*, 2013, pp. 111–117.

[105] P. Gite, ‘Link Stability Prediction for Mobile Ad-Hoc Network Route Stability’, in *International Conference on Inventive Systems and Control (ICISC)*, 2017, pp. 1–5.

[106] Y. Ko and N. H. Vaidya, ‘Location-Aided Routing (LAR) in Mobile Ad Hoc Networks’, *Wirel. Networks*, vol. 6, no. 4, pp. 307–321, 2000.

[107] S. Basagni, I. Chlamtac, V. Syrotiuk, and B. Woodward, ‘A Distance Routing Effect Algorithm for Mobility (DREAM)*’, in *Proceedings of the 4th annual ACM/IEEE international conference on Mobile computing and networking*, 1998, pp. 76–84.

[108] W. Abushiba and P. Johnson, ‘Performance Comparison of Reactive Routing Protocols for Ad Hoc Network’, in *2015 Forth International Conference on e-Technologies and Networks for Development (ICeND)*, 2015, pp. 1–5.

[109] S. Liu, Y. Yang, and W. Wang, ‘Research of AODV Routing Protocol for Ad Hoc

Networks1', *AASRI Procedia*, vol. 5, pp. 21–31, 2013.

[110] S. Sridhar, R. Baskaran, and P. Chandrasekar, 'Energy Supported AODV (EN-AODV) for QoS Routing in MANET', *Procedia - Soc. Behav. Sci.*, vol. 73, pp. 294–301, 2013.

[111] B. D. Shivahare, C. Wahi, and S. Shivhare, 'Comparison Of Proactive And Reactive Routing Protocols In Mobile Adhoc Network Using Routing Protocol Property ':, *Int. J. Emerg. Technol. Adv. Eng.*, vol. 2, no. 3, pp. 356–359, 2012.

[112] S. Lee, M. Gerla, and C. Chiang, 'On-Demand Multicast Routing Protocol', in *Wireless Communications and Networking Conference, 1999. WCNC IEEE.*, 1999, pp. 1298–1302.

[113] N. Makhlouf, 'Enhancing On-Demand Multicast Routing Protocols using Mobility Prediction in Mobile Ad-hoc Network', *Int. J. Adv. Telecommun. Electrotech. Signals Syst.*, vol. 3, no. 2, pp. 40–43, Aug. 2014.

[114] S. Lee, W. Su, and M. Gerla, 'Ad hoc Wireless Multicast with Mobility Prediction', in *Eight International Conference on Computer Communications and Networks*, 1999, pp. 4–9.

[115] X. Wang and K. Kar, 'Throughput Modelling and Fairness Issues in CSMA/CA Based Ad-Hoc Networks', *Proc. - IEEE INFOCOM*, vol. 1, no. C, pp. 23–34, 2005.

[116] R. R. Choudhury, X. Yang, N. H. Vaidya, and R. Ramanathan, 'Using Directional Antennas for Medium Access Control in Ad Hoc Networks', in *Proceedings of the 8th annual international conference on Mobile computing and networking - MobiCom '02*, 2002, p. 59.

[117] R. Govindarajula, *Multiple Access Techniques for Mobile Ad Hoc Networks*. Arizona State University, 2001.

[118] P. Gupta and P. R. Kumar, 'The Capacity of Wireless Networks', *IEEE Trans. Inf. Theory*, vol. 46, no. 2, pp. 388–404, 2000.

[119] M. a Abdala and A. K. Al-zuhairy, 'Integration of Smart Antenna System in Mobile Ad Hoc Networks', *Int. Jouranl Mach. Learn. Comput.*, vol. 3, no. 4, pp. 342–346, 2013.

[120] S. Yi, Y. Pei, and S. Kalyanaraman, 'On the Capacity Improvement of Ad Hoc Wireless Networks Using Directional Antennas', *Proc. 4th ACM Int. Symp. Mob. ad hoc Netw. Comput. - MobiHoc '03*, pp. 108–116, 2003.

[121] N. Makhlouf, ‘Challenges of Integration of Smart Antennas in Ad Hoc Network’, *The 21st Conference Student EEICT*. pp. 428–433, 2015.

[122] Y. Chen, J. Liu, X. Jiang, and O. Takahashi, ‘Throughput Analysis in Mobile Ad Hoc Networks with Directional Antennas’, *Ad Hoc Networks*, vol. 11, no. 3, pp. 1122–1135, 2013.

[123] R. Ramanathan, ‘On The Performance of Ad Hoc Networks with Beamforming Antennas’, *Proc. 2nd ACM Int. Symp. Mob. ad hoc Netw. Comput. - MobiHoc ’01*, pp. 95–105, 2001.

[124] L. Bao and J. J. Garcia-Luna-Aceves, ‘Receiver-Oriented Multiple Access in Ad Hoc Networks with Directional Antennas’, *Wirel. Networks*, vol. 11, pp. 67–79, 2005.

[125] K. Feng, ‘LMA : Location- and Mobility-Aware Medium-Access Control Protocols for Vehicular Ad Hoc Networks Using Directional Antennas’, vol. 56, no. 6, pp. 3324–3336, 2007.

[126] M. N. Alam, M. A. Hussain, and K. S. Kwak, ‘Neighbor Initiated Approach for Avoiding Deaf And Hidden Node Problems in Directional Mac Protocol for Ad-Hoc Networks’, *Wirel. Networks*, vol. 19, pp. 933–943, 2013.

[127] P. Masek, K. Zeman, J. Hosek, Z. Tinka, N. Makhlouf, A. Muthanna, N. Herencsar, and V. Novotny, ‘User Performance Gains by Data Offloading of LTE Mobile Traffic onto Unlicensed IEEE 802.11 Links’, in *38th International Conference on Telecommunications and Signal Processing (TSP)*, 2015, pp. 117–121.

[128] L. I. Perlovsky, *Neural Networks and Intellect: Using Model-Based Concepts*, Oxford Uni. 2001.

[129] K. S. Kasiviswanathan, K. P. Sudheer, and J. He, ‘Quantification of Prediction Uncertainty in Artificial Neural Network Models’, in *Artificial Neural Network Modelling*, Springer International Publishing, 2016, pp. 145–159.

[130] T. H. Oong, N. Ashidi, and M. Isa, ‘Adaptive Evolutionary Artificial Neural Networks for Pattern Classification’, *IEEE Trans. NEURAL NETWORKS*, vol. 22, no. 11, pp. 1823–1836, 2011.

[131] G. Ou and Y. L. Murphey, ‘Multi-Class Pattern Classification Using Neural Networks’, *Pattern Recognit.*, vol. 40, no. 1, pp. 4–18, 2007.

[132] J. Biesterfeld, E. Ennigrou, and K. Jobmann, ‘Neural Networks for Location Prediction in Mobile Networks’, in *Proceedings of the International Workshop on Applications of Neural Networks to Telecommunications (IWANNT’97)*, 1997, pp. 207–214.

[133] X. Li, N. Mitton, and D. Simplot-ryl, ‘Mobility Prediction Based Neighborhood Discovery in Mobile Ad Hoc Networks’, *Lect. Notes Comput. Sci.*, vol. 6640, pp. 147–159, 2011.

[134] J. A. Torkestani, ‘Mobility Prediction in Mobile Wireless Networks’, *J. Netw. Comput. Appl.*, vol. 35, no. 5, pp. 1633–1645, Sep. 2012.

[135] L. Ghouti, T. R. Sheltami, and K. S. Alutaibi, ‘Mobility Prediction in Mobile Ad Hoc Networks Using Extreme Learning Machines’, *Procedia Comput. Sci.*, vol. 19, pp. 305–312, 2013.

[136] L. Ghouti, ‘Mobility Prediction Using Fully-Complex Extreme Learning Machines’, in *European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning*, 2014, pp. 23–25.

[137] D. G. Reina, P. Ruiz, R. Ciobanu, S. L. Toral, B. Dorronsoro, and C. Dobre, ‘A Survey on The Application of Evolutionary Algorithms for Mobile Multihop Ad Hoc Network Optimization Problems’, *Int. J. Distrib. Sens. Networks*, vol. 12, no. 2, pp. 1–13, Feb. 2016.

[138] J. De Freitas, ‘Bayesian Methods for Neural Networks’, University of Cambridge, 2003.

[139] K.-R. Koch, ‘Bayes’ Theorem’, in *Bayesian Inference with Geodetic Applications*, 1990, pp. 4–8.

[140] R. M. Neal, *Bayesian Learning for Neural Networks*, 1st ed. Springer Science & Business Media, 2012.

[141] A. Papoulis and S. U. Pillai, *Probability, Random Variables and Stochastic Processes*, 4th ed. McGraw-Hill Europe, 2002.

[142] A. Hald, ‘probability and least squares’, *Stat. Sci.*, vol. 14, no. 2, pp. 214–222, May 1999.

[143] I. J. Myung, ‘Tutorial on Maximum Likelihood Estimation’, *J. Math. Psychol.*, vol. 47, no. 1, pp. 90–100, Feb. 2003.

[144] N. Makhlouf, ‘A Mobility Prediction Method for Mobile Ad Hoc Networks Using Directional Antennas’, in *Reviewed Proceedings of the Interdisciplinary Scientific International Conference for PhD students and assistants*, 2015, pp. 1365–1371.

[145] N. Aschenbruck, R. Ernst, E. Gerhards-Padilla, and M. Schwamborn, ‘BonnMotion: a Mobility Scenario Generation and Analysis Tool’, in *Proceedings of the 3rd International ICST Conference on Simulation Tools and Techniques*, 2010, pp. 1–31.

[146] D. MacKay, ‘A Practical Bayesian Framework for Backpropagation Networks’, *Neural Comput.*, vol. 4, no. 3, pp. 448–472, May 1992.

[147] T. Sourmail, ‘Softwares for Bayesian neural networks training’, 2004. [Online]. Available: http://thomas-sourmail.net/neural_networks.html. [Accessed: 01-Jan-2016].

[148] D. Mackay, ‘Bayesian Non-linear Modeling with Neural Networks’, in *Mathematical Modelling of Weld Phenomena 3*, 1997, pp. 359–389.

[149] Y. Dodge, *The Oxford Dictionary of Statistical Terms*, 6th ed. Oxford University Press, 2006.

[150] D. Svozil, V. Kvasnicka, and J. Pospichal, ‘Introduction to Multi-layer Feed-forward Neural Networks’, *Chemom. Intell. Lab. Syst.*, vol. 39, no. 1, pp. 43–62, Nov. 1997.

[151] M. Peet, ‘Bayesian Neural Networks as Implemented in Model Manager’, *University of Cambridge*, 2015. [Online]. Available: http://mathewpeet.org/notes/BNNMM/. [Accessed: 01-Jan-2016].

[152] T. Issariyakul and E. Hossain, *Introduction to Network Simulator NS2*. Boston, MA: Springer US, 2009.

[153] İ. Bekmezci, O. K. Sahingoz, and Ş. Temel, ‘Flying Ad-Hoc Networks (FANETs): A survey’, *Ad Hoc Networks*, vol. 11, no. 3, pp. 1254–1270, May 2013.

[154] N. Makhlouf, ‘Mobility Prediction Using Virtual Map for MANET Networks’, *Elektrorevuei J.*, vol. 20, no. 3, pp. 78-83, 2018.

LIST OF ABBREVIATIONS

3GPP	3^{rd} Generation Partnership Project
ACK	ACKnowledgement
ANN	Artificial Neural Network
AODV	Ad Hoc On-Demand Distance Vector
ARH	AutoRegressive Hello
CDMA	Code Division Multiple Access
CTS	Clear-To-Send
DB	Directional Beacon
DCF	Distributed Coordination Function
DNAV	Directional Network Allocation Vector
DREAM	Distance Routing Effect Algorithm for Mobility
DSDV	Destination-Sequenced Distance-Vector
DSR	Dynamic Source Routing
E(D)	The average End to End delay
ELM	Extreme learning Machine
ELS	Enhanced Localization Solution
FG	Forwarding Group
FTP	File Transfer Protocol
GPS	Global Positioning System
HMM	Hidden Markov Model
LAR	Location-Aided Routing
LET	Link Expiration Time
LMA	Location and Mobility Aware

LPE	Log Predictive Error
LT	Location Table
LTE	Long Term Evolution
MAC	Medium Access Protocol
MANET	Mobile Ad hoc NETworks
MAODV	Modified Ad Hoc On-Demand Distance Vector
MaxSR	Spatial Reuse Maximizer
MBAA	Multi-Beam Antenna Arrays
MBM	Map-Based Movement
MBSA	Multiple Beam Smart Antenna
MLE	Maximum likelihood Estimation
MLP	MultiLayer Perceptron
MM	Markov Model
MMM	Mixed Markov chain Model
MN	Mobile Node
MRD	Modified Random Direction
MSE	Mean Square Error
MSRCC	Spatial Reuse Maximizer in Cooperative Communication
NAM	Network Animator
NIP	Neighbor Information Packet
NS-2	Network Simulator-2
ODMRP	On-Demand Multicast Routing Protocol
OLSR	Optimized Link State Routing
PDA	Personal Digital Assistant

PDR	Packet Delivery Ratio
QoS	Quality of Service
RAT	Radio Access Technology
RDM	Random Direction Mobility
RERR	Rout ERRor
RET	Route Expiration Time
RL	Reinforcement Learning
ROMA	Receiver Oriented Multiple Access
RPGM	Reference Point Group Mobility
RREP	Route REPly
RREQ	Route REQuest
RTS	Request-To-Send
RWM	Random Walk Mobility
RWP	Random Waypoint
SMS	Short Message Service
SNR	Signal-to-Noise Ratio
TDMA	Time Division Multiple Access
TORA	Temporally ordered routing algorithm
TTL	Time To Live
VANET	Vehicular Ad Hoc Network
VoIP	Voice over IP
ZRP	Zone Routing Protocol

AUTHOR’S PUBLICATIONS

1. N. Makhlouf, ‘Mobility Prediction Using Virtual Map for MANET Networks’, *Elektrorevue J.*, vol. 20, no. 3, pp. 78-83, 2018. ISSN: 1213-1539.
2. N. Makhlouf, ‘Exploiting Neural Networks for Mobility Prediction in Mobile Ad Hoc Networks’, *Elektrorevue J.*, vol. 18, no. 3, pp. 63–67, 2016. ISSN: 1213-1539.
3. N. Makhlouf, ‘A Mobility Prediction Method for Mobile Ad Hoc Networks Using Directional Antennas’, in *Reviewed Proceedings of the Interdisciplinary Scientific International Conference for PhD students and assistants*, 2015, pp. 1365–1371. ISBN: 978-80-87952-10- 8
4. P. Masek, K. Zeman, J. Hosek, Z. Tinka, N. Makhlouf, A. Muthanna, N. Herencsar, and V. Novotny, ‘User Performance Gains by Data Offloading of LTE Mobile Traffic onto Unlicensed IEEE 802.11 Links’, in *38th International Conference on Telecommunications and Signal Processing (TSP)*, 2015, pp. 117–121. ISBN: 978-1-4799-8497- 8
5. N. Makhlouf, ‘Challenges of Integration of Smart Antennas in Ad Hoc Network’, *The 21st Conference Student EEICT*. pp. 428–433, 2015. ISBN: 978-80-214-5148-3.
6. N. Makhlouf, ‘Enhancing On-Demand Multicast Routing Protocols using Mobility Prediction in Mobile Ad-hoc Network’, *Int. J. Adv. Telecommun. Electrotech. Signals Syst.*, vol. 3, no. 2, pp. 40–43, Aug. 2014. ISSN: 1805- 5443.
7. N. Makhlouf and P. Vajsar, ‘MAC Protocls for MANET Networks with Directional Antennas’, in *Conference of Informatics and Management Sciences*, 2013, pp. 301–304. ISBN: 978-80-554-0648- 0.
8. L. Nagy, V. Novotný, J. Uramová, N. Makhlouf, ‘Performance Analysis of IMS Network: The Proposal of new Algorithms for S-CSCF Assignment’, in the *journal of Advances in Electrical and Electronic Engineering*, 2013, ISSN: 1804- 3119.
9. P. Vajsar, P. Masek, J. Hosek, N. Makhloufu, and J. Leu, ‘Design of QoS Model for Mobile Ad-hoc Network’, in *2013 36th International Conference on Telecommunications and Signal Processing (TSP)*, 2013, pp. 111–117. ISBN: 978-1-4799-0403-7
10. N. Makhlouf, Ľ. Nagy, and P. Vajsar, ‘Prediction of Movement of Wireless Nodes in Mobile Ad-hoc Networks’, in *4th IEEE Intl’ Conference on Cognitive*

Infocommunicaitons, 2013, pp. 115–120. ISBN: 978-1-4799-1544- 6

11. N. Makhlouf and P. Vajsar, 'MAC Protocols in Mobile Ad Hoc Networks',in the *International Journal of Advances in Telecommunications, Electrotechnics, Signals and Systems,.*, vol. 1, no. 1, pp. 9–12, 2012. ISSN: 1805- 5443.
12. P. Vajsar, N. Makhlouf, J. Hošek, M. Bartl, P. Morávek, and R. Figurny, *Implementation of OLSR-Based QoS Support into MANET*. Research in Telecommunication Technologies. 2012. ISBN: 978-80-554-0570- 4

ANNEXES

ANNEX A

Network Simulator NS-2

NS-2 is a free and open-source event-driven simulator used specifically for research projects. Since its inception in 1989, NS-2 has continuously gained tremendous interest from industry, academia, and government [152]. It can be downloaded from the NS-2 homepage. NS-2 provides a highly modular platform for simulation of wired as well as wireless networks. NS-2 contains modules for many network components such as MAC, routing, transport layer protocol, application, etc. The activities of the network are processed and queued in form of events, in a scheduled order. These events are then processed as per scheduled time that increases along with the processing of events. However, the simulation is not real time; it is considered virtual. Network Simulator is mainly written in two languages. They are C++ and OTcl. OTcl is the object-oriented version of Tool Command language. It is used to build the network structure and topology. C++ is the most important and kernel part of the NS-2. A Tcl script is written in NS-2 for simulation of network model. When this Tcl script is run it creates two files trace file and Nam file. The Nam file is distributed with NS-2 simulator to read an input file and draw the network events graphically. It provides visual interpretation of overall network. Whereas trace file stores different events statistics such as each individual packets arrival time, departs or is dropped , information about protocol agent, traffic agent , source and destination nodes address etc., which can be used to measure a protocol performance, see Fig.A.1. In order to extract a subset of the data of interest and analyse them from trace file, different tools are available such as grep, AWK, sed, Perl. I used AWK programming script to analyse the simulation results.

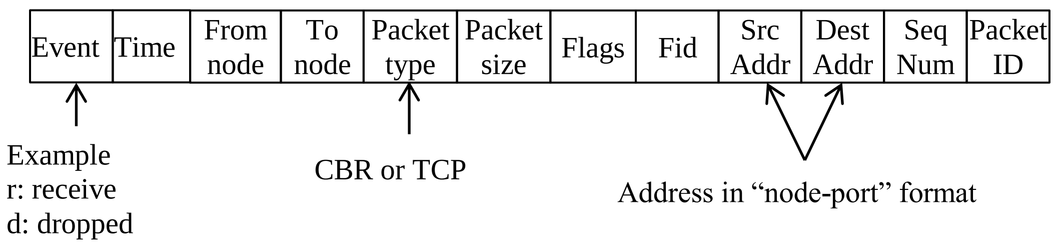


Fig. A.1 An example of a trace file

ANNEX B

Bayesian probability theory

Bayesian probability theory provides a mathematical framework for peforming inference, or reasoning, using probability. The foundations of Bayesian probability theory were laid down some 200 years ago by scientists such as Bernoulli, Bayes, and Laplace, but it has been held suspect or controversial by modern statisticians. The last few decades though have seen the occurrence of a “Bayesian revolution,” and Bayesian probability theory is now commonly employed (oftentimes with stunning success) in many scientific disciplines, from astrophysics to neuroscience. It is most often used to judge the relative validity of hypotheses in the face of noisy, sparse, or uncertain data, or to adjust the parameters of a specific model.

Table B-1 The ranking of the 100 submodels depending on the LPE for x_{n+1}

Number of submodel	Number of hidden units	TE	LPE
1	1	0.0663	40.507
2	2	0.0664	40.493
3	1	0.0664	40.492
4	2	0.0664	40.489
5	20	0.0666	40.461
6	2	0.0666	40.46
7	7	0.0666	40.455
8	15	0.0666	40.455
9	9	0.0666	40.454
10	5	0.0666	40.453
11	13	0.0666	40.45
12	3	0.0666	40.444
13	5	0.0667	40.444
14	10	0.0667	40.444

Number of submodel	Number of hidden units	TE	LPE
15	1	0.0666	40.443
16	2	0.0667	40.44
17	7	0.0667	40.44
18	4	0.0667	40.439
19	7	0.0667	40.439
20	9	0.0667	40.439
21	4	0.0667	40.438
22	14	0.0667	40.438
23	2	0.0667	40.436
24	1	0.0667	40.435
25	10	0.0667	40.434
26	3	0.0667	40.433
27	7	0.0667	40.433
28	6	0.0667	40.432
29	10	0.0667	40.432
30	3	0.0667	40.43
31	8	0.0667	40.43
32	11	0.0667	40.43
33	9	0.0667	40.429
34	19	0.0667	40.429
35	14	0.0667	40.428
36	17	0.0667	40.428
37	6	0.0667	40.427

Number of submodel	Number of hidden units	TE	LPE
38	12	0.0667	40.427
39	6	0.0667	40.425
40	19	0.0667	40.425
41	5	0.0667	40.424
42	8	0.0667	40.423
43	11	0.0667	40.423
44	3	0.0667	40.421
45	11	0.0668	40.421
46	13	0.0668	40.421
47	14	0.0668	40.421
48	19	0.0668	40.421
49	12	0.0668	40.42
50	16	0.0668	40.42
51	4	0.0668	40.419
52	15	0.0668	40.419
53	15	0.0668	40.419
54	17	0.0668	40.418
55	19	0.0668	40.416
56	5	0.0668	40.415
57	6	0.0668	40.415
58	8	0.0668	40.413
59	11	0.0668	40.413
60	15	0.0668	40.413

Number of submodel	Number of hidden units	TE	LPE
61	18	0.0668	40.413
62	5	0.0668	40.411
63	17	0.0668	40.411
64	17	0.0668	40.411
65	20	0.0668	40.411
66	12	0.0668	40.41
67	13	0.0668	40.409
68	18	0.0668	40.408
69	19	0.0668	40.408
70	7	0.0668	40.407
71	13	0.0668	40.407
72	20	0.0668	40.407
73	9	0.0668	40.406
74	20	0.0668	40.405
75	9	0.0668	40.404
76	6	0.0668	40.403
77	10	0.0668	40.403
78	18	0.0668	40.403
79	10	0.0668	40.401
80	8	0.0669	40.4
81	13	0.0669	40.399
82	3	0.0668	40.398
83	4	0.0668	40.398

Number of submodel	Number of hidden units	TE	LPE
84	16	0.0669	40.394
85	8	0.0669	40.393
86	16	0.0669	40.392
87	12	0.0669	40.39
88	4	0.0669	40.384
89	16	0.067	40.379
90	14	0.067	40.372
91	16	0.067	40.358
92	18	0.067	40.354
93	1	0.0672	40.326
94	20	0.069	40.291
95	18	0.069	40.281
96	17	0.0691	40.273
97	11	0.0691	40.246
98	12	0.0692	40.234
99	14	0.0694	40.202
100	15	0.0694	40.201

Table B-2 The ranking of the 100 submodels depending on the LPE for y_{n+1}

Number of submodel	Number of hidden units	TE	LPE
1	3	0.0544	41.34
2	12	0.0544	41.34
3	5	0.0544	41.337

Number of submodel	Number of hidden units	TE	LPE
4	2	0.0544	41.327
5	18	0.0545	41.322
6	11	0.0545	41.32
7	12	0.0545	41.318
8	12	0.0545	41.318
9	19	0.0545	41.306
10	14	0.0545	41.305
11	20	0.0545	41.304
12	11	0.0545	41.301
13	18	0.0545	41.299
14	2	0.0545	41.298
15	7	0.0545	41.298
16	8	0.0545	41.296
17	14	0.0545	41.296
18	13	0.0545	41.295
19	16	0.0545	41.294
20	19	0.0545	41.293
21	19	0.0545	41.289
22	12	0.0545	41.286
23	4	0.0545	41.284
24	18	0.0545	41.283
25	11	0.0545	41.281
26	15	0.0545	41.281

Number of submodel	Number of hidden units	TE	LPE
27	14	0.0546	41.28
28	16	0.0546	41.279
29	8	0.0545	41.278
30	9	0.0546	41.276
31	7	0.0546	41.275
32	13	0.0546	41.275
33	3	0.0546	41.274
34	6	0.0546	41.274
35	8	0.0546	41.274
36	3	0.0546	41.272
37	10	0.0546	41.272
38	4	0.0546	41.271
39	6	0.0546	41.271
40	13	0.0546	41.27
41	10	0.0546	41.269
42	6	0.0546	41.268
43	16	0.0546	41.267
44	17	0.0546	41.266
45	20	0.0546	41.266
46	20	0.0556	41.266
47	9	0.0546	41.265
48	14	0.0546	41.265
49	17	0.0546	41.263

Number of submodel	Number of hidden units	TE	LPE
50	9	0.0546	41.262
51	15	0.0546	41.262
52	4	0.0546	41.261
53	10	0.0546	41.259
54	4	0.0546	41.257
55	15	0.0546	41.257
56	5	0.0546	41.254
57	16	0.0546	41.254
58	11	0.0546	41.252
59	9	0.0546	41.251
60	13	0.0546	41.251
61	13	0.0546	41.251
62	2	0.0546	41.249
63	10	0.0546	41.249
64	3	0.0546	41.247
65	1	0.0546	41.246
66	17	0.0546	41.246
67	16	0.0546	41.245
68	15	0.0546	41.243
69	5	0.0546	41.242
70	6	0.0546	41.242
71	20	0.0546	41.242
72	2	0.0546	41.24

Number of submodel	Number of hidden units	TE	LPE
73	1	0.0546	41.239
74	1	0.0546	41.239
75	11	0.0546	41.239
76	1	0.0546	41.238
77	19	0.0546	41.237
78	2	0.0546	41.236
79	18	0.0546	41.235
80	10	0.0547	41.234
81	3	0.0547	41.229
82	7	0.0547	41.229
83	8	0.0547	41.229
84	7	0.0547	41.225
85	17	0.0547	41.222
86	12	0.0547	41.219
87	5	0.0547	41.216
88	7	0.0547	41.214
89	4	0.0547	41.212
90	5	0.0547	41.204
91	6	0.0547	41.204
92	8	0.0547	41.2
93	9	0.0547	41.2
94	19	0.0561	41.171
95	17	0.0561	41.164

Number of submodel	Number of hidden units	TE	LPE
96	14	0.0548	41.159
97	18	0.0561	41.143
98	15	0.0562	41.126
99	20	0.0563	41.098
100	1	0.0551	41.016

Annex C

Table C-3 The prediction result of the final model

Target x	Predicted x	Target y	Predicted y
129.7156	107.228424	231.0979	220.1442
120.1699	129.330154	240.2457	229.1602
110.7698	120.305794	230.8672	238.6118
136.3183	111.496231	212.2988	228.9203
169.8127	135.609055	217.5237	209.9602
166.5246	167.68103	215.7803	215.2594
163.6189	164.5298	197.7517	213.482
156.4737	161.743744	197.9661	195.3978
169.7807	154.890884	169.0575	195.6078
164.4491	167.650208	179.6855	167.7419
165.5044	162.539764	154.2109	177.7965
135.8401	163.551575	176.3146	154.106
162.7199	135.153152	173.3911	174.5855
158.3512	160.881302	174.123	171.8138
176.6424	156.691452	170.8021	172.5016
175.84	174.217834	169.395	169.3742
156.929	173.450638	148.185	168.0524
126.8719	155.327484	149.1247	148.721
112.9182	126.634323	167.0078	149.5552
81.87854	113.501808	166.779	165.8281
64.97796	85.075562	141.5899	165.6168

Target x	Predicted x	Target y	Predicted y
65.14812	70.207657	140.9583	142.9303
61.62059	70.354774	116.0995	142.3769
64.85123	67.314583	110.3939	121.6165
96.99024	70.097878	106.8863	117.0847
77.21342	98.755966	85.75106	114.3493
78.49543	80.922539	113.502	98.57555
78.09761	82.060425	113.1833	119.5439
114.2324	81.707031	107.8142	119.2867
90.04467	114.730919	110.876	115.0711
80.32447	92.426933	139.7865	117.4664
75.51623	83.688049	148.2459	141.3662
73.77048	79.42099	149.8323	148.7719
64.69909	77.881088	143.9131	150.1802
80.92021	69.966545	164.2392	144.9546
113.2512	84.219643	146.3105	163.2607
96.1193	113.813095	179.3829	147.0654
75.44638	97.958939	190.6821	177.5116
82.97872	79.359337	229.2408	188.4417
92.60815	86.060059	228.0355	227.2507
92.72517	94.755333	226.4292	226.0058
67.89718	94.861855	213.4125	224.3546
48.65807	72.740074	209.7198	211.086
49.6661	56.351059	224.8924	207.3549

Target x	Predicted x	Target y	Predicted y
54.68765	57.191692	247.2709	222.7808
60.14778	61.409374	246.4741	245.905
78.74902	66.052628	261.6077	245.0733
43.26599	82.285995	269.2236	260.8248
43.02196	51.891014	268.6239	268.7519
55.85355	51.690758	297.8515	268.1212

CURRICULUM VITAE

PERSONAL INFORMATION

Title, name and surname	Eng. Nermin Makhlouf
Address	Skácelova 61, 612 00 Brno
Phone	(+420) 775 131 868
Email	nerminmakhlouf@hotmail.com
Date of birth	11 September 1986

EDUCATION AND TRAINING

Period	September 2011 – present
Institution	Brno University of Technology – Faculty of Electrical Engineering and Communication – Dept. of Telecommunications
Field of study	Electrical and Communication Technology
Degree	Ph.D. (in progress; expected completion: 2018)

Period	September 2004 – September 2009
Institution	Damascus University, Syria
Field of study	Electronics and Telecommunications Engineering
Degree	Diploma (M.A. equivalent); recognized

WORK EXPERIENCE

Period	September 2011 – present
Employer	Brno University of Technology – Faculty of Electrical Engineering and Communication – Dept. of Telecommunications
Field of work	Research
Position	Ph.D. student and junior researcher

Period	September 2009 – August 2011

Employer	Damascus University – Dept. of Electronics and Telecommunication Engineering
Field of work	Scientific research
Position	Lecturer; courses taught:

- Electronic Circuits
- Advanced Communications and Electronics Systems

PROJECT PARTICIPATION

Period	2014 - 2016
Project	Centre of Sensor, Information and Communication system (SIX)
Position	Investigator

SKILLS

Languages	Arabic – native tongue English – B2 Czech – B1
IT	Cisco Certified Network Associate MATLAB, NS-2 Website creation: JavaScript, HTML Programming languages: C++, OTCL MS Office: MS Word, Excel, Power Point, Visio

PUBLICATION ACTIVITIES:

Papers published in impact factor journals	1
Papers published in reviewed journals	4
Papers indexed in Scopus	4
Papers published in international conferences	4
Papers published in domestic conferences	1