FMPM: Futuristic Mobility Prediction Model for Mobile Adhoc Networks Using Auto-Regressive Integrated Moving Average

T. Prasannavenkatesan and T. MEnakadevi

Department of Electronics and Communication Engineering
Adhiyamaan College of Engineering Hosur
Anna University Chennai, India
E-mail: prasannait91@gmail.com

Abstract

The mobility of the node plays a crucial role in the route discovery process for Mobile Ad hoc networks (MANET). The high-speed node affects the routing process by packet relaying, packet delivery. This paper proposes a mobility prediction model FMPM using Auto-Regressive Integrated Moving Average (ARIMA) method to estimate the futuristic speed values of the nodes in MANET. It uses the autocorrelations of previous data by time series forecasting approach. The ARIMA model applied to mobile nodes for predicting its future speeds in the network. This mobility prediction supports the route discovery process to select the moderate mobility nodes for the reliable routing in the network. The nodes are trained by ARIMA model using neural network time-series tool of Matlab. Roughly, it trains each node with ten iterations. Simulation results show that the forecasted values almost match with the simulated node speed values. Performance analysis of the predicted mobility values exhibits the improved results compared to existing works on performance metrics such as mean square error, covariance, and computation overhead. The proposed FMPM produces more perfection in the speed prediction of a mobile node with a maximum of 80-100 % throughout the time series and has the lowest error values as 0.1085. But, the neural based prediction has the lowest MSE as 0.852. Therefore, the ARIMA prediction has approximately 0.75 reduced error values.

Keywords: ARIMA, Mobility prediction, MANET, Auto-regression, Random waypoint model

1. Introduction

A group of mobile nodes forms the self-organizing network called as MANET. It is the peer-to-peer network and does not require any pre-existing infrastructure. The nodes can act as end-host as well as routers. Each node has the capability to configure themselves in the network [1]. The nodes can establish the connection and communication with other nodes in two ways: single hop and/or multi-hop. If the nodes are within their transmission range, then it easily establishes the communication with others. In the case of multi-hop, it requires the involvement of many nodes helps to establish communication with another node. The extensive characteristics and nature of MANET create the applications in various areas including military, disaster, and emergency scenarios [2]. The MANET's critical characteristics such as unpredictable mobility, distributed multi-hop communication, limited resources, deviation of signals, and shared wireless medium severely disrupts the routing as well as applications. The dynamic behaviour of nodes causes the unexpected, rapid change of routing paths; as well as the mobility of nodes disturbs the communication links frequently [3].

Several papers present a prediction of bandwidth and energy while does not focuses on the effects of node mobility in the network. Numerous research works deals with discovering the shortest path among nodes by distance and delay [3-10]. In such shortest paths, the possibility of path breakup is higher; it can occur quickly because of node's mobility speed. It makes series disruption of routing and data loss. Thus, the stable paths are highly needed for routing; the prediction of future mobility of nodes can accomplish it [4, 5]. The exact prediction of the node mobility supports the effective route discovery process before the start of the routing. This paper presents an FMPM mobility prediction model to forecast the node speeds in ad hoc networks using the following approaches:

- The proposed FMPM model develops the generalized auto-regression technique and time series analysis to predict the futuristic mobility values.
- FMPM adopts the Auto-Regressive Integrated Moving Average (ARIMA) method for forecasting the speed values of nodes.
- The moving-average model resolves our work by differentiating and calculating auto-regression.

The remainder of the paper organized as follows. Section 2 explains the significant recent works on the existing mobility and other prediction models in MANET. Section 3 discusses the methodologies of this work. This part presents the approaches developed and incorporated for the prediction of node mobility. Section 4 evaluates the performance outcomes of the proposed methodology with the extensive performance metrics. Section 5 concludes the paper with future scope.

2. Literature Review and Related Works

This section deliberates several works on determining the future resources of the nodes. It discusses the existing model for the prediction of path length, energy, and reliability of the nodes.

Author A. Yassir et al., (2013) focused on mobility prediction of nodes in MANET, to save the air interference of the radio signals [12]. Based on the mobility patterns and past location of nodes, it designs the mobility models to the network. This information trained using ANN-feedback algorithm and Jorder, Elman, and hierarchical Elman network. Theoretical experiments show that it has better performance based on location prediction the mobility than the conventional methods. However, in highly scalable and dynamic network the proposed method may fail. This work simulates only 20 nodes and does not analyses any simulation/implementation.

The movement of the node in the MANET was predicted for routing by H. Kaaniche and F. Kamoun (2010). The stable path for the routing is established by mobility prediction using the ANN-BP algorithm [11]. It utilises onboard Global Positioning System (GPS) to learn the location of each node using the time series prediction technique. It selects a slower mobility node as the intermediate node for location prediction. The movement of the node should be in the direction of the source node and does not move away from the source node. It considers the recurrent neural network for the mobility prediction. The mathematical experiment shows that it works well on predicting the long-term mobility in the MANET using ANN-BP. However, this work discusses analytical experiments only. It does not have any significant implementation works for accurate prediction the mobility of a node in long-term time series; also it does not compares other existing works.

To forecast movement of the mobile nodes in MANET L. Ghouti (2016) were presented the neural network learning based resolutions. The mobility models such as the random waypoint, random walk, Manhattan grid, Gauss-Markov, vehicular, smooth random models are deeply discussed in this paper [13]. Location, velocity, and direction of change to time are predicted using mobility pattern, MLP, and ELM (Extreme Learning Machine). The power generalization capability intends to reduce the performance errors between training and the testing dataset using bias-variance dilemma. Author [15] introduced a scheme that increases and/or decreases the node traversal time (NTT), to increase the AODV performance in MANET. Although, NTT does not address the performance of the node in MANET. It does not discuss the operations of ANN in this work. This paper [14] indicates that, ELM has higher performances than MLP in many cases such as mobility prediction, MSE, and MAE. ELM based mobility prediction useful in predicting routing nodes with reduced battery and data exchange. K. N. Raju and S. P. Setty (2015) had improved the AODV performance by adopting the ANN approaches.

J. Prakash et al., (2012) presented an ANN based approach to forecast the delay of packet delivery in routing. It [17] designs a generalized regression neural network (GRNN) using RBF for the DSDV, AODV, and DSR models. It discusses the parameters average number of neighbour nodes for the training process and path length in ANN. However, the aptness of GRMM in dynamic environments like MANET is questionable. N. Maan and R. K. Purwar (2012) had presented a performance study on DSDV, AODV and Ant-based routing protocols based on swarm intelligence of ANN and Ant Colony Optimization (ACO). The swarm intelligence observes the decentralized and self-organized nodes in the system [18]. This paper concludes ant-based routing protocols perform better on many performance metrics. Nevertheless, it does not discusses indepth details about swarm intelligence and parameters of the swarm intelligence.

S. K. Shah and D. D. Vishwakarma (2010) were adopted an ANN method to improve the network performance of AODV protocol. This paper [16] discusses problems on periodically updating the routing information (HELLO update), which increases unnecessary network traffic and uses more bandwidth. ANN learning making decisions using computation process. Increasing HELLO beconning makes more opportunity to interfere with data. Decreasing HELLO message may, results link-breakages because of no updates on messages in that time-gap. The ANN enhances the time-period of HELLO messages by using the correlation of various topology. It uses the multilayer feedforward ANN and fuzzy AODV with a conventional back-propagation method to train the data.

Our contributions:

- Most of the papers concentrate on the estimation of link expiration, node's location prediction, path length prediction by many approaches in the MANET.
- To the best of our knowledge, none of the work focuses on the prediction of accurate node speed values using the ARIMA model in mobile networks.
- The proposed FMPM model adopts the ARIMA model to predict the futuristic mobility of the nodes using time series forecasting.
- Auto-regression facilitates the detection of cooperative nodes to provide an uninterrupted mobility based reliable routing using the node's previous speed history in the MANET.

3. Methodology

3.1 Proposed ARIMA Model

Our work primarily focusses on the precise forthcoming prediction of speed values of high speed or moderate speed mobile nodes. As mentioned earlier, it adopts ARIMA model and works by differencing moving average of values. This model was presented by the mathematicians Box and Jenkins (B-J model). The ARIMA is a more appropriate tool to predict the time series of data. ARIMA has a lower root mean square error, reduced processing time and works better than many prediction models [5]. The ARIMA (p, d, q) model contains a pool of models, that is autoregressive model AR (p), moving average model MA (q), smooth average moving model ARMA (p, q), integrated autoregressive moving average model ARIMA (p, d, q) with 'd' time. This works exploit the use ARIMA (p, d, q) model with the non-stationary sequences of a time series; where d is the smoother time difference of node speed values. Its aptness for MANET presented in the following subsections [6, 7].

3.2 Futuristic Mobility Prediction Model (FMPM)

The movement of node majorly disrupts the wireless connection and routing amongst the nodes. The repeated link-breakage in the connectivity between nodes is conceivable when the high-speed node considered for routing. The higher mobility (speed) node could move outside the node transmission zone while routing in progress [19]. Thus, the necessity to omit the higher speed node in the route discovery process emerges. It results discovering higher stable route, which should have reasonable mobility and routing overhead with lower link-breakages between the nodes. The proposed work, introduces the speed estimator factor to predict the low-speed mobile nodes. The ARIMA model computes the futuristic speed (mobility) (λ_{SE}) of nodes in MANET. The ARIMA model constitutes three tuples: p, d, q; where 'p' indicates the number of time lags; 'd' characterizes the number of subtracted data (speed) values form its past values in a time series; 'q' specifies the order of moving average method. Let consider a time series of data 'SV_t; $t>0$', the $ARIMA(p, d, q)$ model follows as in the equation (1):

$$SV_t - \alpha_1 SV_{t-1} - \alpha_2 SV_{t-2} - \ldots - \alpha_{p^{'}} SV_{t-p^{'}} = \varepsilon_t + \theta \lim_{\delta x \to 01} \varepsilon_{t-1} + \theta_2 \varepsilon_{t-2} + \ldots + \theta_q \varepsilon_{t-q}$$

$$(i.e.)\ \ SV_t(1 - \sum_{i=1}^{p^{'}} \alpha_i T^i) = \varepsilon_t (1 + \sum_{i=1}^{q} \theta_i T^i)\ \ \ldots\ldots\ldots (1)$$

Where SV_t is the several speed values in a time series; α_i represents the autoregressive part (subtraction of previous speed values of the mobile node); T^i is the number of time lag operations; θ_t gives the error factor, which is the normal distribution with zero mean; θ_i represents the moving average term of the speed values.

A factor of $(1 - T)$ and the multiplicity 'd' are the unit roots. It is presumed with the polynomial $(1 - \sum_{i=1}^{p^{'}} \alpha_i T^i)$ and p assumed as a polynomial factor of $p^{'} - d$. It gives as

$$(1 - \sum_{i=1}^{p^{'}} \alpha_i T^i) = (1-T)^d\ \ (1 - \sum_{i=1}^{p^{'}-d} \varphi_i T^i)$$

It makes the ARIMA (p, d, q) model by the drift rate δ. It modifies the final model as per the equation given in (5):

$$\therefore\ SV_t(1 - \sum_{i=1}^{p} \varphi_i T^i)(1-T)^d = \delta + \varepsilon_t (1 + \sum_{i=1}^{q} \theta_i T^i)\ \ldots (2)$$

The derived equation given in (2) estimates and models the futuristic speed value of a mobile node as speed estimator metric of the FMPM model. To calculate the speed estimator (λ_{SE}) value, this work considers the mobile adhoc network as the undirected weighted graph as presented in Fig. 1 and positive weights (speed) given to the nodes. At time 't', the speed (m/s) values of the nodes are illustrated in Fig.1. The weight given to the nodes from one to hundred (i.e.) S=> 35, 1=> 40, 2=> 50, 3=> 65, 4=> 50, 5=> 35, 6=> 45, D=> 10. The speed estimator (λ_{SE}) of a node is calculated based on the speed (mobility) of the nodes in various sessions. Consider, 'T' as the maximum time with the disjoint set of the several time value of 't' likewise given in the equation (3) and N as the number of speed samples.

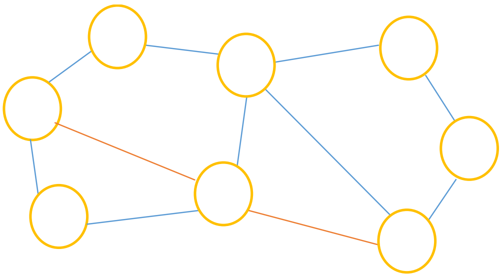


Figure 1. Speed of mobile nodes at Time 't'

$$T = \{t_1, t_2, t_3 \ldots\ldots, t_n\}\ ;\ \ \ t_i \le T\ \ \ \ldots\ldots(3)$$

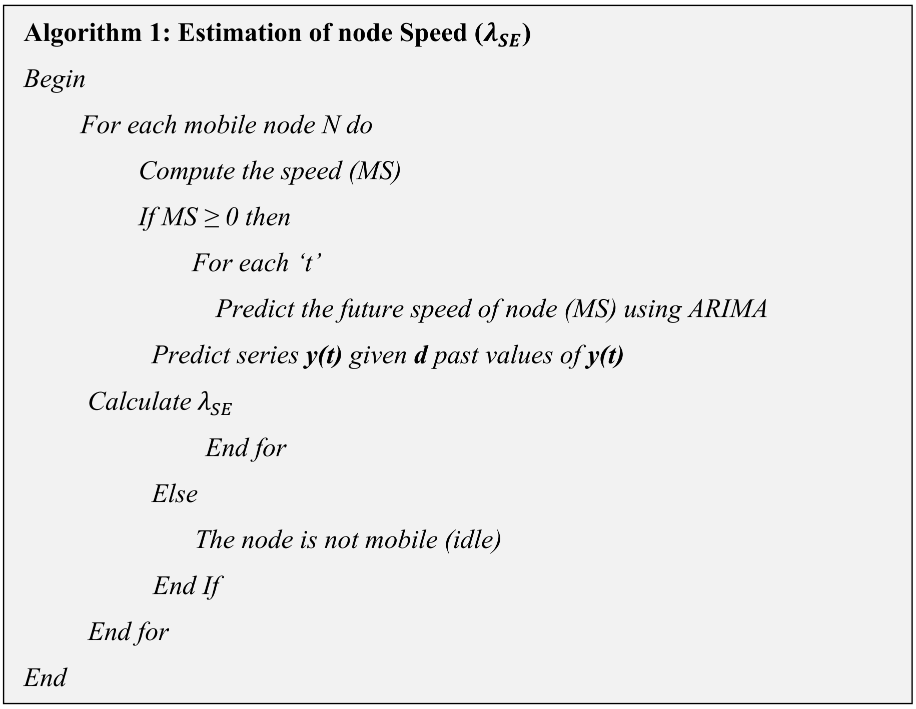
$$\lambda_{SE} = \frac{MS_{t1} + MS_{t2} + MS_{t3} + \ldots\ldots + MS_{tn}}{N};\ t_i \ge 1, n \le N$$

$$(i.e.)\ \ \ \lambda_{SE} = \sum_{ti=1}^{n} \frac{MS_{ti}}{N}\ \ \ \ldots\ldots\ldots(4)$$

The ns2 simulator extensively tests the predicted speed values and evaluates with the observed values. The algorithm 1, illustrates the pseudo code and steps involved in the calculation of speed estimator (λ_{SE}) of a node in the network.

3.3 Recurrent Neural Network Time series analysis

The proposed work uses the supervised-learning through discrete-time analysis to predict the upcoming speed values of the node. The supervised-learning examines the previous values, specifies as a sequence of input real-value to the input nodes. In this timestamp, each neuron determines its present activation



Algorithm 1: Estimation of node Speed (λ_{SE})

```
Begin
    For each mobile node N do
        Compute the speed (MS)
        If MS ≥ 0 then
            For each 't'
                Predict the future speed of node (MS) using ARIMA
          Predict series y(t) given d past values of y(t)
    Calculate $\lambda_{SE}$
              End for
          Else
              The node is not mobile (idle)
          End If
    End for
End
```

with the nonlinear function. It is the sum of weighted activations of all neurons. Recursively applying the same set of weights makes the recursive neural network over a differentiable structure. The structure of the recurrent neural resembles as a linear chain. The RNN aims to display the dynamic temporal behaviour of any data. The RNN processes the arbitrary sequences of inputs as feedback neural networks. Nevertheless, the Multilayer Perceptron (MLP) neural network only uses the feedforward neural networks. RNN also helps to understand and model the several un-segmented tasks such as handwriting recognition or speech recognition. The time-series analysis includes methods for evaluating the time-series data to extract the abstract statistics, other features of the data. Time-series model is more useful to forecast and predict the upcoming values based on perceived past data values. The regression-analysis employed to analyse the current values of independent time-series and approximates how these current values disturb other time series. The time series analysis compare the single, multiple time series and interrupted time series analysis at various points in time or on a single time series.

The neural time-series tool solves various classes of non-linear time series problems dynamically. It divides the dataset into training dataset, validation dataset, and testing dataset. It defines the network architecture and implements the training in the network. The neural time-series analysis tool can train any type of datasets. Afterwards, the network training evaluates the performance using regression analysis and mean squared error. The neural time-series analysis tool examines the outcomes using the error autocorrelation plot, a histogram of the errors. The test set evaluation also performed by the network analysis tool. The retain feature with modified settings is also can be adopted, if the results are not satisfied with a larger dataset. Neural net time series tool also generates the customized training process for the optimal results for solving several classification problems. Thus, the neural net time-series analysis tool solves the nonlinear time series problem by training a dynamic neural network.

3.4 Validation and analysis of network mobility of FMPM

The ns2 simulation and Weibull distribution validate the predicted speed values of the proposed FMPM model for the comparative analysis. Since, Weibull distribution function supports modelling the lifetime events and produces significant results for the reliability testing. The Weibull distribution adopted for testing the survivability of a mobile node's speed values in a time series and it is manipulated using the cumulative distribution function defined in (5).

$$F(x,k,\lambda)=1-e^{-(x/\lambda)^k};\quad x\geq 0, k>0, \lambda>0 \quad \ldots(5)$$

Where k represents the shape parameter; λ characterises the scale parameter of the distribution. It is a stretched exponential function. Weibull distribution provides distribution of failure rate in the power of time. The k and λ parameters of the problem are interpreted as follows; a value of $k < 1$ then the rate of failure decreases over time that means the failure rate of the system decreases over time. The $k = 1$, states that the failure rate of the system is constant. If $k > 1$, whenever time increases then the failure rate also increases, which means the system likely to fail as time goes on.

4. Simulation and Performance Evaluation

This part of the paper predicts the speed values of a node using ARIMA and RNN time series tool; then analyses and discusses its performances against ns2.

4.1 Speed prediction using Random waypoint model in ns2

The following strategy compares and analyses the predicted and actual speed values of a node in the mobile network. First, a MANET is constructed with the using the Network Simulator (NS-2.35) simulator. The NS2 is an event-driven simulation tool used to simulate the model for MANET [21]. NS2 a more suitable tool to simulate the dynamic nature of mobile nodes with several mobility model in various networks, especially for mobile ad-hoc networks [22]. The proposed FMPM model constructed for the 50 number nodes and that are placed arbitrarily within the 1000x1000m^2 area with an Omni-directional antenna. Each node has the radio propagation range of 150 meters in the network. Table.1 summarises the simulation parameters of the work. The random waypoint model adopted as the mobility model to generate node mobility. It creates movements to the nodes; when the simulation started, each node randomly moves towards the destination with varying speed from zero to 100, based on the random waypoint model.

Table 1. Simulation Parameters

S. No.	Specifications	Value
1.	Number of nodes	0-50
2.	Area size	1000 m X 1000m
3.	MAC	Wireless LAN (802.11)
4.	Transmission range	150 m
5.	Antenna Type	Omni-directional antenna
6.	Channel capacity	2 Mbps

Once the node reaches its destination, then it pauses for some time and then repeats the same process until the end of the simulation [19, 22]. The similar simulation environment simulates the nodes for 100 seconds. The speeds of the nodes are set to 0 m/s as minimum and 100 m/s as maximum values. In this

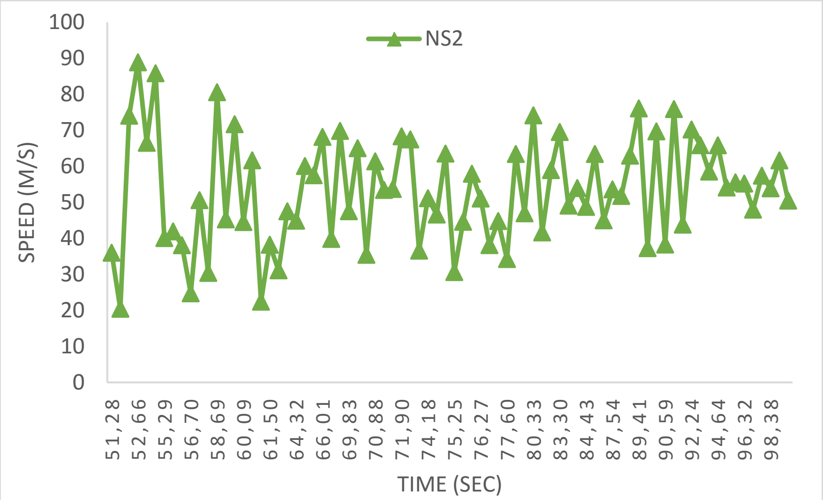


Figure 2. Speed predicted of a node using NS2

experiments, the movement of a node starts at the pause time of 0 sec, and the node becomes stationary at the pause time of 100 sec. The proposed algorithms evaluate the speed of nodes at various intervals of time. Movements are generated using mobility scenario generation and analysis tool such as BonnMotion tool [20]. The ARIMA and neural network time-series analysis take the first 50 seconds of the simulated data for training from the 100 seconds. The ARIMA and neural network time-series takes the next 50 seconds for the predicted speed values analysis. Fig. 2 illustrates the predicted speed values of a node for 51 to 100 seconds using the random waypoint model in ns2.

4.2 Speed prediction using ARIMA

The FMPM using ARIMA anticipates to forecast the accurate futuristic values by integrating moving average and autoregression of the assumed problem. Likewise, it considers the prediction of future speed values of a node as a nonlinear autoregressive problem. The ARIMA adapts time-series analysis in MATLAB-R2015a [20] to predict the future speed values. It forecasts series x(t) given ‘d’ previous values of x(t). This work takes the target time-series that is the speed values of a node as input to the ARIMA model. Each node takes 110-130 timestamps with their speed values as a matrix. The given data matrix (timestamp × speed) is randomly divided into two data matrices for training and validation as 70% and 30% timestamps respectively. The training dataset adjusts the network according to the errors. The validation dataset processes the network generalization and stops when it starts degrading [14]. The ARIMA time-series predicts the 10-15 step ahead speed values. Thus, the problem is defined as $x(t) = f(x(t-1), x(t-2), \dots, x(t-5))$.ARIMA time-series algorithm trains the given dataset and stops when the mean square error (MSE) increases. MSE is the average of the squared difference between outputs and targets. Fig. 3 depicts the speed prediction of a node using ARIMA of (p, d, q) model. Here, to predict the futuristic speed values of a node for 51 to 100 seconds, the first 50 seconds speed values are given as the input for the prediction. Based on the given input, ARIMA processes and produces the speed values for the next 50 seconds, as depicted in the Fig. 3. Table.2 presents the different (p, d, q) models and their MSE values for the speed value prediction of a node. It shows that (3, 2, 2) has lowest MSE value.

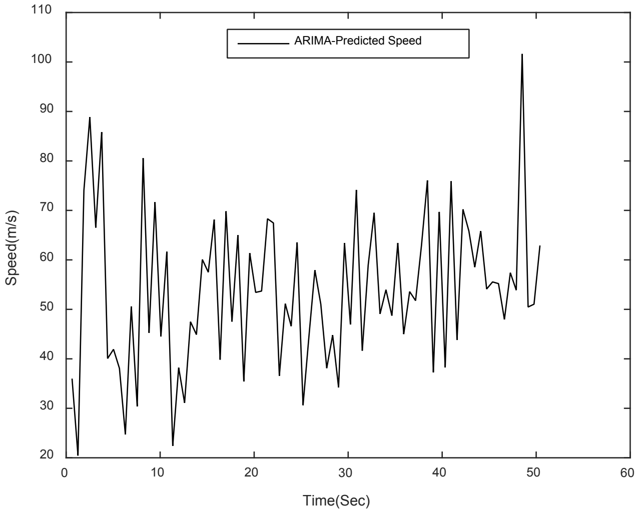


Figure 3. Predicted Speed using ARIMA

Table 2. ARIMA Performance Metrics

(p, d, q)	MSE
(3,2,2)	0.1085
(1,1,2)	0.2852
(1,2,1)	0.40788
(2,1,1)	9.9604

4.3 Speed prediction-using neural network time series

Similarly, to predict the speed values of a node using the neural network time series, this work uses the Nonlinear Autoregression (NAR) tool in MATLAB-R2015a. The neural network based prediction utilises the similar parameters used for the ARIMA based prediction. Levenberg-Marquardt training algorithm trains the given dataset and stops when the MSE increases. Approximately, each node has ten iterations. The dataset given to this prediction is similar to the one used in the ARIMA based prediction that is ns2 simulated (Random Waypoint model) 0-50 second speed values. The RNN time series tool trains

Table 3. MSE performances of RNN

Neurons	MSE
5	5.658
10	7.477
15	2.585
20	7.878
25	0.852
30	4.326
35	2.088
40	7.543
45	5.213
50	3.449

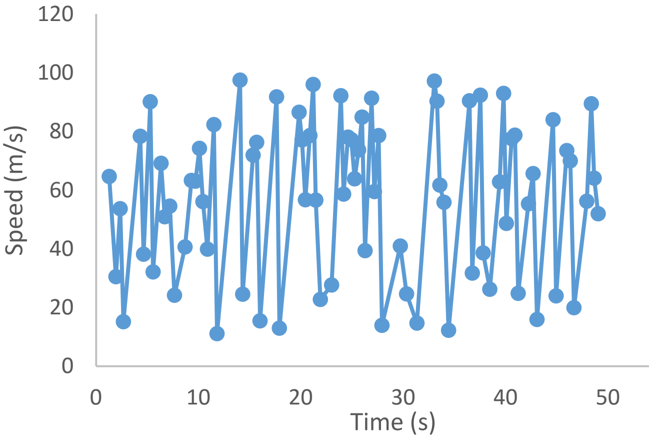


Figure 4. Predicted Speed using RNN-NAR

the training dataset and then tests the output samples. To discover the number of neurons for the RNN network, the numbers of neurons varies from 5 to 50 with the constant delay value of 5 for all training set. It produces the futuristic speed values for the 51 to 100 seconds for the dataset.

Fig. 4 depicts the predicted speed values using the time series neural network nonlinear autoregression tool; The RNN activation function determines the futuristic speed behaviour of a node as like shown in Fig. 4. Table.3 presents the predicted various MSE performances of the time series neural network for the given dataset. It evaluates the performance for a different number of neurons as depicted in Table.3. It also illustrates the effect regression values (R) values for the different number of neurons. The R-value is the relationship between the predicted output and the target values. The values nearing to one (1) specifies both targets and outputs have high relationships. The values closer to zero (0) indicates the relationship between target and output values is very less. According to Table.2, the neurons with 25 has the very good R-value as 0.97. Nevertheless, all the R values do not go below 0.5. Nevertheless, the lowest MSE value of ARIMA based prediction is 0.1085 for the (3, 2, 2) model, but the RNN based prediction has the lowest MSE as 0.852 for the neuron 25. It conforms that the ARIMA based model predicts precise futuristic speed values of a node.

4.4 Comparison of predicted speed values using ARIMA, RNN-NAR, and NS2

This section evaluates and discusses the comparison of predicted speed values with the targeted values. Up to this, we predicted the futuristic speed values of a node using ARIMA, RNN-NAR, and NS2 (random waypoint model). The following work compares the predicted speed values using ARIMA and RNN-NAR with the ns2 simulated speed values for 51 to 100 seconds; because this work takes the first 50 seconds of the ns2 predicted speed values as the input dataset for the training of both ARIMA and RNN-NAR. Fig. 5 illustrates the predicted speed values using ARIMA and NS2. Fig. 6 shows the predicted speed values using RNN-NAR and NS2. For all speed prediction, the speed values of a node are set to vary from 0 to 100 m/s as minimum and maximum values respectively.

From the Fig. 5, we can see that the predicted speed values using FMPM and simulated speed values using ns2 are majorly coincides with each other. It also shows that at periods of time 51 to 52, 66 to 70, 71 to 74, 76 to 83.5, 88.5 to 90, and 97 to 98.4 the speeds of predicted and actual speed values are more or less same as the overlapped line. Most of the speed range of the ARIMA and ns2 is similar to each other. On the other hand, in Fig. 6, the predicted speed values using RNN-NAR are out of range; and they are dissimilar with the

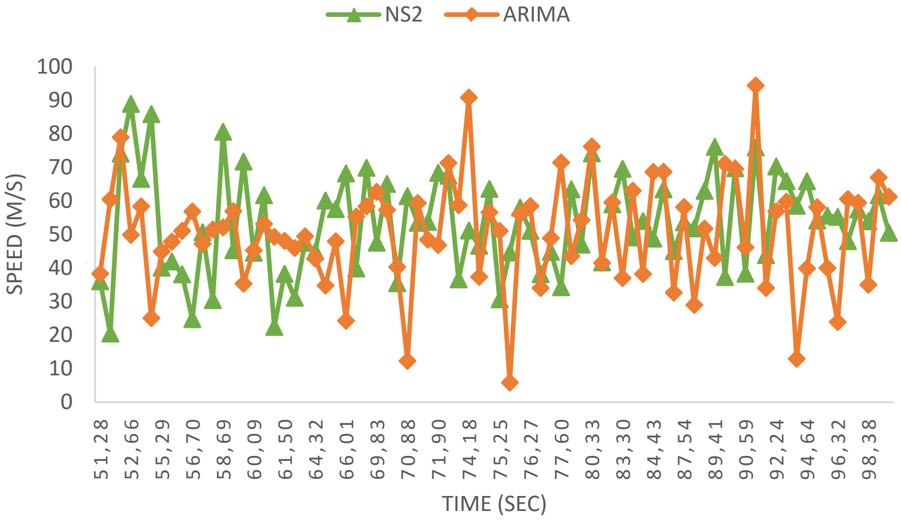


Figure 5. Comparison: Speed prediction- ARIMA vs. NS2

NS2 simulated results. As per Fig. 6, there is no similarity among RNN-predicted speed values and NS2 simulated values. The RNN predicted and NS2 simulated speed values coincide with each other at certain seconds only. Typically, a major period of predicted speed values in the time series is irrelevant to the observed values. At the time 52, 55, 59, 65, 74, 76, 78, 81, 85-87, 91 seconds both predicted and observed values match each other.

Thus, it shows that the overlapping of both values occurs in a very small period in the time series from 51 to 100. Fig. 7, shows the comparison of predicted speed values using ARIMA, RNN time series (NAR), and NS2 predicted speed values for the period 51-100 seconds. It illustrates that the predicted values using the ARIMA and ns2-simulated values majorly coincide with each other than time series neural network. As per the illustration in the Fig. 7, the ARIMA based prediction model exactly matches with the ns2 simulated values at most of the time, whereas the neural network based prediction coincides at a certain time (at 55, 76, 79 and 85 to 87) only. However, the predicted and observed speed values are equal to each other or undoubtedly within close range for the ARIMA algorithm based predictions at a maximum time. For example, at the time interval 66 to 70 seconds, the

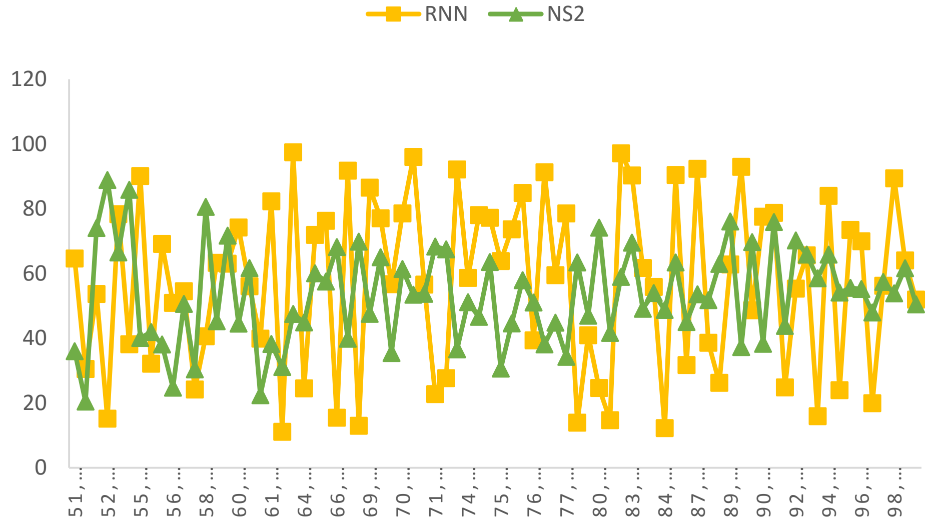


Figure 6. Comparison: Speed prediction- RNN-NAR vs. NS2

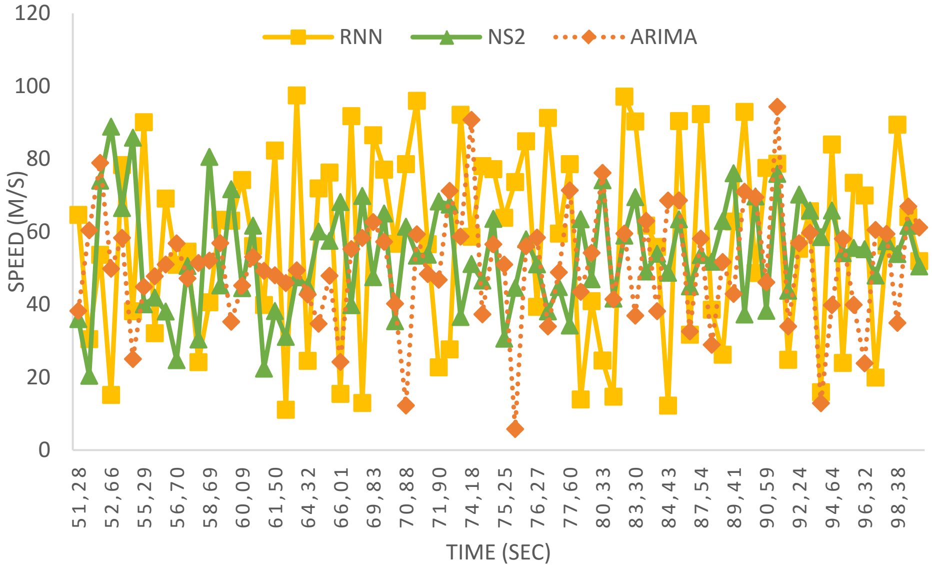


Figure 7. Comparison: Speed prediction-ARIMA, RNN-NAR, NS2

predicted speed is more coincided with an actual speed value of the node. Thus, it confirms that the FMPM using ARIMA effectively predicts the more or less accurate future speed values of a node. Fig. 7 gives the FMPM using ARIMA predicted speed values have more matches with the actual values than the RNN predicted values.

5. Conclusions

MANET is a very dynamic network with dissimilar speeds of nodes. It makes frequent link breakages and degrades the routing as well produces tedious overheads. This work predicts the future mobility values of nodes using the ARIMA model and RNN time series neural network; further, it analyses the accuracy of the predicted speed values of a node using the FMPM model. Simulated and predicted results show that the proposed FMPM using ARIMA accurately forecasts the mobility of nodes that is speed values. At the speed 65.5-69.5, the predicted speed coincides more for the ARIMA with an actual speed value of the node. Thus, the FMPM model accurately predicts the future speed values of a node using the ARIMA model. It enables more stable routing for the MANET. It has lowest error values as 0.1085 and neural based prediction has the lowest MSE as 0.852. Therefore, the ARIMA prediction has approximately 0.75 reduced error values. The other resource constraints of the nodes (energy, channel availability, etc.) should be predicted for more precise routing. These developments will be incorporated into the routing protocol as the future enhancements of this work. These prediction enhancements help to support the reliable routing.

References

1. S. S.Chaudhari, and R. C. Biradar, "Traffic and mobility aware resource prediction using a cognitive agent in mobile ad hoc networks", *Journal of Network and Computer Applications*, *72*, 87–103, 2016.
2. T. Prasannavenkatesan, and T. Menakadevi, "Significance of Scalability for On-Demand Routing Protocols in MANETs", *Proceedings of IEEE Conference on Emerging Devices & Smart Systems,* Namakkal, March 4-5, pp. 76-82, 2016.
3. A. Pal, J. P. Singh, and P. Dutta, "Path length prediction in MANET under AODV routing: Comparative analysis of ARIMA and MLP model," *Egyptian Informatics Journal*, 103–111, 2014.
4. H. Kaaniche, F. Kamoun, "Mobility Prediction in Wireless Ad Hoc Networks using Neural Networks", *Journal of Telecommunications, 2*(1), 95–101, 2010.
5. R. K. Yadav, M. Balakrishnan, "Comparative evaluation of ARIMA and ANFIS for modelling of wireless network traffic time series", *EURASIP Journal of Wireless Communications and Networking*, vol. 1, pp. 1-15, *2014*.
6. Y. Cao, T. Cao, H. Ye, Y. Yan, J. Chu, "ARIMA Prediction Model-based Cluster Algorithm in Ad Hoc Networks", *Proceedings of the 2nd International*

Conference on Computer Application and System Modeling, 132–136, 2012.

7. X. Li, N. Mitton, D. Simplot-ryl, "Mobility Prediction Based Neighborhood Discovery in Mobile Ad Hoc Networks."*Research Report, INRIA, RR-7469,* 2011.
8. X. M. Zhang, F. F. Zou, E. B. Wang, D. K. Sung, "Exploring the dynamic nature of mobile nodes for predicting route lifetime in mobile Ad hoc networks," *IEEE Transactions on Vehicular Technology*, *59*(3), 1567–1572, 2010.
9. Y. Cong, X. Zhou, R. A. Kennedy, R. A. "Interference prediction in mobile ad hoc networks with a general mobility model," *IEEE Transactions on Wireless Communications*, vol. 14, no.8, 4277–4290, 2015.
10. E. Y. Hua, Z. J. Haas, "An algorithm for prediction of link lifetime in MANET based on Unscented Kalman Filter," *IEEE Communications Letters*, *13*(10), 782–784, 2009.
11. H. Kaaniche and F. Kamoun, "Mobility Prediction in Wireless Ad Hoc Networks using Neural Networks," *Journal of Telecommunication,* vol.2(1), pp. 95–101, 2010.
12. A. Yassir, G. A. Nasir, and P. Roy, "Mobile Ad-hoc Networks Location Prediction by using Artificial Neural Networks : Considerations and Future Directions," *Int.J.Computer Technology and Applications,* vol. 4, no. 1, pp. 120–125, 2013.
13. L. Ghouti, "Simulation Modelling Practice and Theory Mobility prediction in mobile ad hoc networks using neural learning machines," *Simulation Modelling Practice and Theory,* vol. 66, pp. 104–121, 2016.
14. L. Ghouti, T. R. Sheltami, and K. S. Alutaibi, "Mobility Prediction in Mobile Ad Hoc Networks Using Extreme Learning Machines," *Procedia - Procedia Comput. Sci.*, vol. 19, no. Ant, pp. 305–312, 2013.
15. K. N. Raju and S. P. Setty, "Artificial Neural Network Based Decision on Parameter Values in AODV to Enhance the Performance of Mobile Ad Hoc Networks," *International Journal of Computer Science and Information Technologies*, vol. 6, no. 5, pp. 4375–4377, 2015.
16. S. K. Shah and D. D. Vishwakarma, "Development and Simulation of Artificial Neural Network based decision on parametric values for Performance Optimization of Reactive Routing Protocol for MANET using Qualnet," *IEEE Conference on Computational Intelligence and Communication Networks,* pp. 167-171, 2010.
17. J. Prakash, P. Dutta, and A. Pal, "Delay Prediction in Mobile Ad Hoc Network using Artificial Neural Network," *Procedia Technol.,* vol. 4, pp. 201–206, 2012.
18. N. Maan and R. K. Purwar, "Role of Artificial intelligence in MANET," *International Journal of Advanced Research in Computer Engineering and Technology*, vol. 1, no. 4, pp. 102–104, 2012.
19. T. Prasannavenkatesan, R. Raja, and P. Ganeshkumar, "PDA-Misbehaving Node Detection & Prevention for MANETs," *Proceedings of International Conference on Communication and Signal Processing,* Melmaruvathur, pp. 1808–1812, 2014.
20. Matlab NTS-NAR Tool. Retrieved from: https://www.mathworks.com/
21. NS2 simulator. Retrieved from: http://www.isi.edu/nsnam/ns/
22. T. Prasannavenkatesan, P. Rajakumar, and A. Pitchaikkannu, "Overview of Proactive Routing Protocols in MANET," IEEE Proceedings of International Conference on Communication Systems & Network Technologies, Bhopal, pp. 173-177, 2014.