doi: 10.20532/cit.2016.1002643

A New QoS-Aware Routing Protocol for MANET Using Artificial Neural Network

Prakash Srivastava and Rakesh Kumar

Department of Computer Science and Engineering, Madan Mohan Malaviya University of Technology, Gorakhpur, India

The explosive growth of Internet leads to increased diversity of applications. Multimedia applications, which are delay sensitive and include telephony, videoconferencing etc. can benefit from this approach to satisfy their QoS routing requirement. Existing route discovery involves estimation of delay at every mobile node which involves calculation of queueing delay as well as contention delay. However, it is a complex procedure but it is required in fulfilling QoS routing requirements in MANET. Artificial neural networks are emerging as a promising technology which can be applied in networking field, simulating human mind in learning from experiences.We have utilized the capability of artificial neural networks (ANN) for accurate prediction of end-to-end packet delay in MANET. Existing link failure strategy involves frequent route discoveries which incur high routing overhead and increased end-to-end delay. In this paper, efficient link failure recovery is also incorporated by utilizing multiple alternate paths in case of link failure, which is determined through signal intensity level and link expiration metric to provide link failure prediction before it breaks up.The proposed methodology has been implemented in a computer simulation. Results show that our approach has been found outperforming to existing approaches and has the potential to be applied in real world scenarios.

ACM CCS (2012) Classification: Networks → Network protocols → Network Layer Protocols → Routing Protocols

Networks → Network types → Ad hoc networks → Mobile ad hoc networks

Keywords: MANET, QoS routing, neural networks, link failure

1. Introduction

Mobile ad hoc Network (MANET) is a self organized, infrastructureless wireless network without any centralized administration [1]. The goal of QoS aware routing is to identify the optimal path that satisfies the stringent requirements of QoS parameters like delay, bandwidth, jitter etc. Multimedia applications such as audio and video have much more stringent QoS requirements. For a network to guarantee QoS deliveries, it has to reserve and control resources like minimal bandwidth requirement and delay gaurantees. The main issues and challenges of MANET routing protocols are to deal with link failures and route recovery in these situations. The efficiency of route recovery affects the overall performance of MANETs. AODV has two route repair strategies to deal with link failure. Routes are repaired by either reestablishing a new route starting from source node, or it can be locally repaired by the node that detects the link break along the end-to-end path [2]-[6].

The end-to-end delay in QoS routing consists of two types: contention delay and queueing delay. Contention delay consists of the latencies for data transmissions and retransmissions. On the other hand, queueing delay is the amount of time that the data packet waits until it gets the shared MAC layer interface. The estimation of end-to-end packet delay [7] in mobile ad hoc network environment is complex as it depends on significant variables such as path length from source to destination, average neighbours of intermediate hops, interference, medium access control protocol etc.

Source initiated route recovery will lead to better performance, but in other situations local repair will be the more appropriate choice. In either case, the route update process attempts to use flooding-search to obtain alternate path to the destination, it results in the decrease of the network throughput as well as in long delays. The problem gets worst when mobility is high. Our approach also focuses on estimation of link expiration metric (LET) and link failure prediction [8] of next node link using signal strength detection technique. If the signal strength detection or LET is low, predecessor node searches an alternate route utilizing a mesh structure and establishes a path after receiving acknowledgement from an alternate node to support effective link recovery strategy.

1.1. Contributions

Specific contribution of the paper focuses on designing a new QoS aware routing protocol for supporting delay sensitive applications. During route discovery process, end-to-end delay is predicted using artificial neural network by taking into account parameters, viz.: average number of neighbours and hop length, which helps in identifying routes. An artificial neural network is a powerful tool [9] which has the property to learn, adapt and predict the principle of learning from experiences. Besides this, efficient route recovery strategy is also incorporated by utilizing multiple alternate paths in case of link failure, which is determined through signal intensity level and link expiration metric to provide link failure prediction before it breaks up. The proposed efficient route recovery strategy ensures lesser routing overhead, hence results in optimized network performance.

Furthermore, comprehensive performance evaluation of the proposed approach with respect to varying scenarios of traffic load and mobility situations are analyzed. Comparison with existing state of the art related scheme is also performed to validate performance improvement over existing one.

1.2. Challenges in QoS Routing

QoS routing has to consider application requirements and the availability of network resources. Consequently, QoS routing in ad hoc networks exhibits great challenges which is reflected in Table 1.

Table 1. QoS routing challenges.

Dynamic topology	The topology of wireless ad hoc networks changes with time due to mobility scenario. It leads to imprecise network state information at the nodes, therefore it is a challenging task toprovide QoS guarantees
Shared bandwidth	The bandwidth in a wireless network is shared byother nodes in the network. The transmission from a node not only consumes localresources but also consumes the bandwidth of the neighbors within the contention range
Resource constraint	Mobile nodes have power limitations as they are battery operated. If the battery power is not efficiently used, mobile nodes will fail quickly which will affect the network availability andfunctionality
Lack of centralized control	The absence of any centralized control demands the routingprotocols to be self-creating and self-organizing. Further, the protocols must also bedistributed in nature

The delay sensitive applications along with their maximum delay tolerance value are given in Table 2. If the value exceeds beyond this, the performance will degrade. Hence, the objective of QoS routing will not be fulfilled.

Table 2. Delay sensitive applications with their requirements.

Type of Applications	One way delay tolerance value
Voice over IP (VOIP)	150 ms
Video Conferencing	150 ms
IPTV	100 ms
Video on demand	50 ms
Gaming	50 ms

The rest of the paper is organized as follows: Section 2 describes related work about QoS routing and link failure recovery schemes. Section 3 presents our proposed QoS routing protocol and neural network model. Section 4 remarks the analytical validation of our proposed approach. Section 5 reflects our simulated and predicted results using Network training tool in MATLAB. Finally, in Section 6 we conclude the paper with directions of future work.

2. Related Work

Kuppusamy *et al.* [10] compared the performance of Temporary-Ordered Routing Algorithm (TORA) and routing on-demand acyclic multipath algorithms, but these algorithms require additional control messages to construct and maintain alternate routes. TORA is a highly adaptive distributed routing algorithm which has been tailored for operation in a mobile networking environment. The basic underlying routing mechanism of TORA is neither a distance-vector nor a link-state algorithm, but it is one of a family of *link-reversal* algorithms. TORA routing protocol also degrades their performance because this protocol requires additional control message for alternate route setup. Therefore, this protocol performs very poorly in high load and frequent disconnection environments.

Zafar *et al.* [11] proposed a new capacity-constrained QoS-aware routing scheme referred to as the shortest multipath source: Q-SMS routing which allows node to obtain and then use estimation of the residual capacity to make appropriate admission control decisions. In the Q-SMS scheme, there is no provisioning of any predictive way to anticipate a route break, which causes performance degradation, particularly in mobile scenarios.

Singh *et al.* [12] proposed the prediction of end-to-end packet delay in Mobile Ad hoc Network using AODV, DSDV and DSR routing based on Generalized Regression Neural Network (GRNN) and Radial Basis function. However, there is no provisioning of link recovery.

Surjeet *et al.* [13] proposed a novel on demand QoS routing protocol MQAODV for bandwidth constrained delay sensitive applications in MANETs. It discovers routes based on bandwidth constrained path delay in addition to hop count. QoS is not guaranteed in case of route break or network partition, so this approach does not suit well for mobile topologies. The link failure prediction strategy is also not incorporated in this scheme.

Wang *et al.* [14] proposed a protocol that uses an alternate path only when data packets are not deliverable through primary route. This scheme however, is based on shortest path first algorithm for wireless networks.

Lee *et al.* [15] proposed AODV-BR Backup Routing, a modified protocol version of AODV which can be implemented in any demand unicast routing protocol to improve the reliable packet delivery in case of node movement and route breaks. The mess configuration provides multiple alternate routes and is constructed without extra overhead. This alternate route is utilized only when data packet cannot be delivered through primary route. Therefore, possibility of packet loss during handoff period can arise. This protocol also increased the number of route discovery processes because each time when a route fails, route discovery process is started by sending node. The performance of AODV-BR routing protocol decreases when the traffic load and mobility increase.

Gafur *et al.* [16] proposed an efficient local route repairing approach on the basis of TTL value as he considers that traditional route recovery technique does not always provide optimal path. In this paper they state that if there is any link breakage in the network, the alternative route to the destination can be derived with a guarantee that it is not a suboptimal route. This paper does not concern the calculation of TTL value from the point of breakage in actual networking scenario.

3. Proposed QoS Routing Protocol

The proposed protocol focuses on QoS aware route discovery that provides the guaranteed end-to-end delay prediction and effective route recovery strategy throughout the communication process. It will also focus on effective route maintenance strategy in case there is a link break or failure. The prediction of link failure is done before the actual failure occurs and alternate path is searched so that the data transmission is continued without any further delay. The prediction of link failure is done on the basis of signal strength detection and link expiration time (LET) metric. The protocol predicts end-to-end delay dynamically on the basis of artificial neural network (ANN) for the QoS aware routing path to support the stringent requirements of delay sensitive applications. For route selection, it considers only those routes which have end-to-end delay, less than or equal to that specified by the delay sensitive applications. In this section, we describe our proposed protocol, which includes training of Multilayer perceptron on the basis of average number of neighbours (N) and hop length (H). These parameters are chosen because they have a strong correlation and dependency on end-to-end delay. As average number of neighbours increases, there will be more collision and retransmission-leading to more delay. The increase in hop length results in accumulation of interface queuing delay. The predicted output is obtained as end-to-end delay. The accurate prediction given by neural network is used in QoS routing to identify optimal path that contains less end-to-end delay or, as per requirements, of delay sensitive applications. Hence, route discovery part is achieved with less routing overhead and complexity.

The Algorithm 1 for route discovery has been given below, along with neural network model which is depicted in Figure 1.

3.1. Neural Network Model

Our approach uses multilayer perceptron architecture of a neural network. The number of neurons in the input layer is enumerated as two as shown in Figure 1. The number of neurons in the hidden layer has a tremendous influence on the final predicted output. Using less number of neurons in the hidden layer will result in underfitting and too many neurons result in overfitting. Underfitting situation occurs when there are too few neurons in the hidden layers to adequately detect the signals in a complicated data set and overfitting situation occurs when the neural network has much information processing capacity that the limited amount of information contained in the training set is not enough to train all of the neurons in the hidden layers. Here, we choose to take two neurons in the hidden layer to avoid underfitting and overfitting situations. The neural network is trained through Back propagation learning algorithm which itself comes under the category of supervised learning strategy. In the input layer we

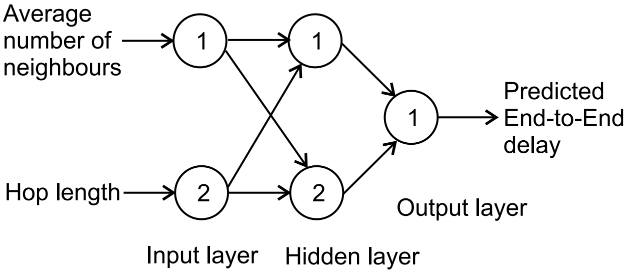

Figure 1. Multilayer Feed Forward Neural Network.

Algorithm 1. Route discovery.

```
Step 1: if Source node S has no route to the destination
            then broadcast a RREQ
Step 2: Training and testing data set for Neural Network is obtained on the basis of average number of
        neighbour nodes and hop length
Step 3: For each routes$_i$ available
        {
         predict end-to-end delay$_i$ on the basis of ANN
        }
Step 4: if (end-to-end delay$_i$ < max_delay)
            then buffer the route$_i$.
            else try after some time when the mobility changes
        Identified route$_i$ will be used for QoS routing.
Step 5: if destination node receives RERR packet due to local route repair fail
Step 6:     then pick up a fresh route, next better route, from buffer and unicasts RREP to the source
```

used the linear transfer function as the activation function considering $g = \tan\phi = 1$ and in the hidden and output layer we used sigmoidal function which behaves in a non linear activation fashion which is shown below:

$$O = \frac{1}{\left(1+e^{-\lambda I}\right)} \tag{3.1}$$

where,

$\lambda \rightarrow$ sigmoidal gain

$O \rightarrow$ Output at output layer

The sigmoidal gain λ provides a better method of coping with network paralysis and it can be applied to wide variety of applications so that faster training can be achieved.

3.2. Training Data Collection

The data sets used for training are obtained through simulated data using NS-2 tool. The random way point (RWP) movement pattern of mobile node is generated using BonnMotion [17] software. The maximum and minimum speed of a node is set to 10 m/s and 0.5 m/s respectively. The routing protocol used is Ad hoc on demand Distance Vector (AODV) routing. The IEEE 802.11 Distributed coordination function is used as MAC layer by every mobile node. The various scenarios of network are analyzed through simulation and their results are manipulated through trace file output using awk script. The end-to-end delay, average number of neighbors and hop length from source to destination are obtained using trace files. The hundreds of data sets are obtained through various scenarios of hop length and average number of neighbors for training artificial neural network. Seventy data sets were used for training the neural network, fifteen data sets were used for testing and remaining fifteen data sets were taken for validation.

3.3. Route Recovery Strategy

The link failure predicatibility is calculated on the basis of link expiration time (LET) and signal strength detection. If the route is unstable or likely to be broken then alternate path is selected for transfer of control packets to identify whether the nodes in the alternate route have the routing entry to the destination. Whenever the receiving node detects a weak signal strength then it sends a control message to transmitting node, the node starts local route discovery process by broadcasting the control message with one hop count. The availability of alternate route is acknowledged by alternate nodes, lying in radio range of transmitting node. Acknowledgement message contains information (either P_ACK or N_ACK) and this route status is forwarded to the receiving node. The searching and establishment of alternate paths are depicted in Figure 2.

Each mobile node keeps and maintains a Session ID table ⟨Src. Addr, Dest. Addr, SessionId⟩, Route table ⟨Dest. Addr, Dest. Seq. no., Session Id, Src. Addr, Next_Addr, Last_Addr, Time out⟩, Route Buffer table ⟨Session_ID, Alternate_route⟩. The Session ID table is used to record the current pair of source address, destination address and session ID. The route buffer table is used to store alternate routes available to each session. Route table keeps routing information for each session. Algorithm 2 refers to brief steps for effective route maintenance. The flow chart for route maintenance is depicted in Figure 3.

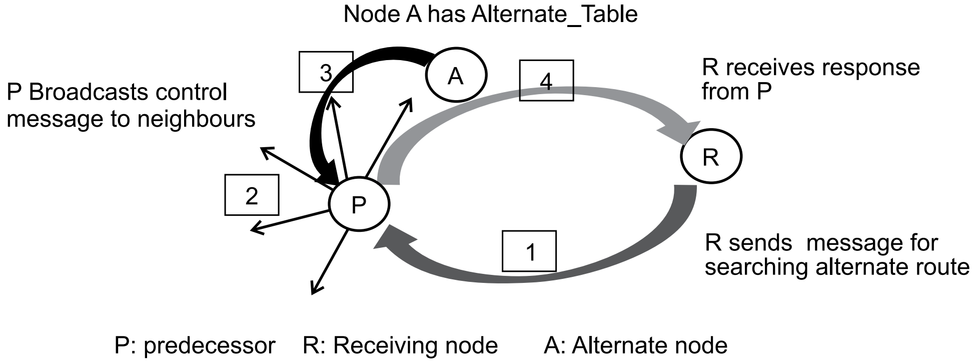
Node A has Alternate_Table
3
A
4
R receives response from P
P Broadcasts control message to neighbours
R
2
P
1
R sends message for searching alternate route
P: predecessor R: Receiving node A: Alternate node

Figure 2. Searching and establishment of alternate path.

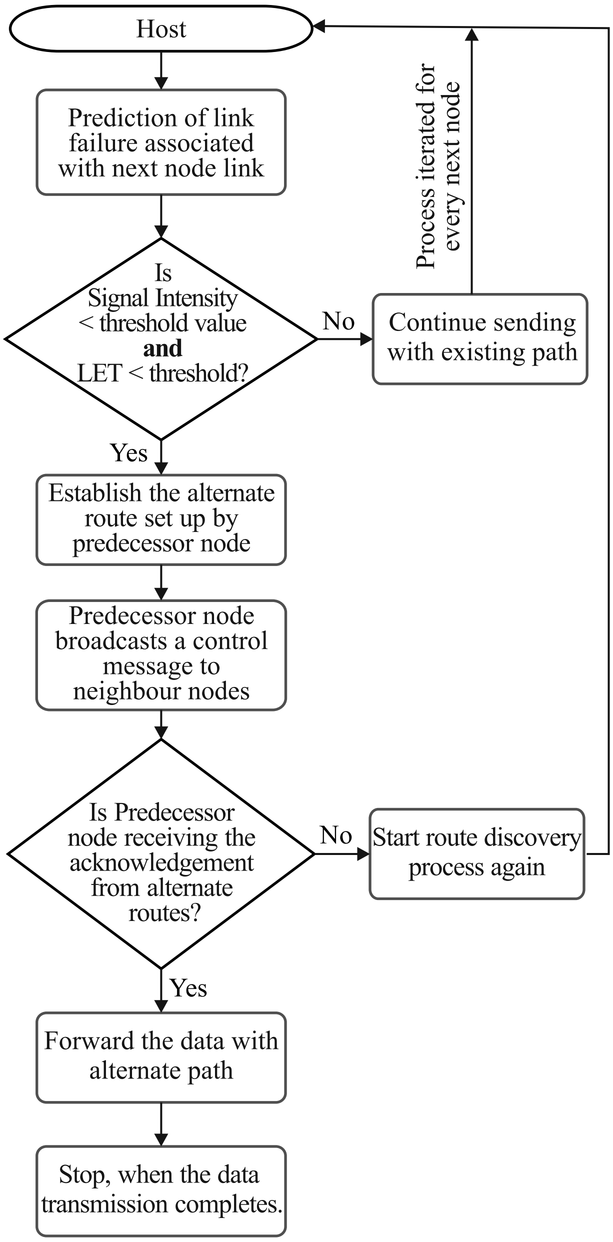
Host
Prediction of link failure associated with next node link
Is Signal Intensity < threshold value **and** LET < threshold?
No
Continue sending with existing path
Process iterated for every next node
Yes
Establish the alternate route set up by predecessor node
Predecessor node broadcasts a control message to neighbour nodes
Is Predecessor node receiving the acknowledgement from alternate routes?
No
Start route discovery process again
Yes
Forward the data with alternate path
Stop, when the data transmission completes.

Figure 3. Flow chart for route maintainenance.

3.3.1. Signal Strength Detection

The strength of a packet signal, which a node receives is obtained as per the following equation [18]. The threshold value considered here is 40 dBμ (dB-microvolts per meter).

$$p_r = \frac{p_t G_r G_t H_r^2 H_t^2}{d^4} \tag{3.2}$$

where $p_r \leftarrow$ strength of received signal

$p_t \leftarrow$strength of transmitted signal

$G_r \leftarrow$ antenna gain of receiver

$G_t \leftarrow$antenna gain of transmitter,

$H_r \leftarrow$ antenna altitude of receiver

$H_t \leftarrow$ antenna altitude of transmitter

$d \leftarrow$ distance between sending and receiving node is given as per equation 3.3:

$$d = \sqrt[4]{\frac{p_t G_r G_t H_r^2 H_t^2}{p_r}} \tag{3.3}$$

3.3.2 Link Expiration Time (LET)

Let i and j be two neighbouring wireless nodes in a route with (x_i, y_i) and (x_j, y_j) coordinates respectively. Also, let v_i and v_j represent the velocity of the node i and node j moving in the direction of θ_i and θ_j respectively. For the transmission range r of node i (as shown in Figure 4) for link i to j to be active, d must be less than r; if the value of d increases, the route breakage may occur and the link becomes unstable. Consequently, determination of path stability is relevant before data transmission achieves the desired level of quality of service. The link expiration time [19] is calculated with the help of equation 3.4:

$$LET = \frac{-(ab+cd)+\sqrt{(a^2+c^2)r^2-(ad-bc)^2}}{a^2+c^2} \tag{3.4}$$

where

$a = v_i \cos\theta_i - v_j \cos\theta_j$, $b = x_i - x_j$, $c = v_i \sin\theta_i - v_j \sin\theta_j$ and $d = y_i - y_j$.

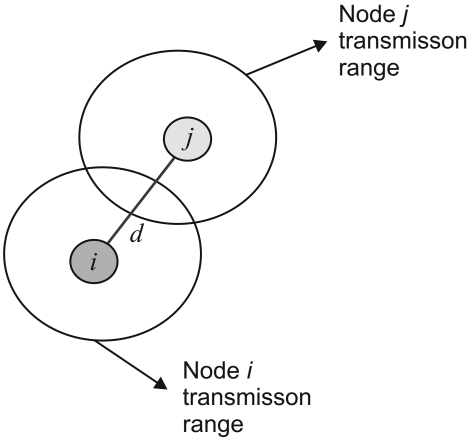

Figure 4. Link expiration range.

Algorithm 2. Route maintenance.

```
Step 1:  Calculate signal strength by each mobile and stationary node as per equation 3.2
Step 2:  if (computed signal strength <= threshold value) and (LET<=1)
           then
         Receiving node sends a message to the predecessor node.
Step 3:    else go to step 12
Step 4:  Predecessor node broadcasts control message to its neighbors
Step 5:  if Predecessor node receives acknowledgement from its neighbors
           then
Step 6:  if reply message includes P_ACK
           then Alt-Route: = P_ACK //P_ACK shows availability of alternate route
Step 7:    else if Alt-Route: = N_ACK //N_ACK shows unavailability of alternate route
Step 8:    else Alt-Route: = NULL
Step 9:  Predecessor node forwards Alt-Route status to receiving node.
Step 10: if linkfailureoccurs
           then
         if Alt-Route = = P-ACK
           then
         go to Step 12
Step 11: Send RERR message to source
Step 12: Continue data transmission with existing available path.
```

4. Analytical Model

The analytical model is based on IEEE 802.11 protocol for Wireless Local Area Networks (WLAN) and most Wireless LAN and MANETs use this standard. This standard operates in two modes: DCF (Distributed Coordination Function) and PCF (Point Coordination Function). A mobile node M transmits in DCF mode using RTS and CTS handshakes to avoid hidden terminal problem. The delay [20] at each node is calculated with the help of following equations:

$$\begin{aligned} &forwarding\_delay = \\ &\quad P_{idle}^{i}(DIFS) \times (DIFS + avg\_bt + DA(i)) \\ &\quad + (1 - P_{idle}^{i}(DIFS)) \times (SIFS{+}DB(i)){+}(L/R) \end{aligned} \quad (4.1)$$

where $P_{idle}^{i}(t)$ is the probability that mobile $node_i$ detects no other mobile node transmitting data during time interval $'t'$ and is given by $P_{idle}^{i}(t) = e^{-\lambda t}$, λ is the aggregate arrival rate including neighbour node at mobile node MN_i.

$DA(i)$ is expected delay encountered in the attempt state and is given by:

$$\begin{aligned} DA(i) = {} & P_{idle}^{i}(slot) \times (RTS + 2 \times SIFS + CTS) \\ & + (1 - P_{idle}^{i}(slot)) \times (RTS{+}2 \times SIFS{+}DB(i)) \end{aligned} \quad (4.2)$$

$DB(i)$ is expected delay encountered during back-off state and is given by:

$$\begin{aligned} DB(i) = {} & [1/\{P_{idle}^{i}(DIFS) \times P_{idle}^{i}(slot)\}] \\ & \times [P_{idle}^{i}(DIFS) \times (DIFS + avg\_bt \\ & {+}RTS{+}2 \times SIFS{+}P_{idle}^{i}(slot) \times CTS)] \\ & {+}[1 - P_{idle}^{i}(DIFS) \times X] \end{aligned} \quad (4.3)$$

avg_bt is a random back-off time interval before transmission and is given by:

$$\begin{aligned} &avg\_bt = \\ &\sum_{n=0}^{4} [P_{idle}^{i}(slot) \times (1 - P_{idle}^{i}(slot))^{n} \times 2^{n-1} \times W] \\ &\quad {+}(1 - P_{idle}^{i}(slot))^{5} \times 2^{4} \times W \end{aligned} \quad (4.4)$$

$$X = RTS + 3 \times SIFS + CTS + L + ACK \tag{4.5}$$

L and R are packet length and data rate respectively. W is contention window size whereas ACK is length of acknowledgement packet. In wireless links, the propagation delays are very small and almost equal for each hop along the path. So, here we assume that the propagation delay is negligible.

4.1. Network Model

We assume that the network consists of N nodes that are distributed uniformly and independently over a rectangular area. Each node in the network can act as a source, destination and/or relay of packets. The radio transmission range of each node is assumed to have an equal transmission range, denoted by d. Let d_{ij} denote the distance between node i and j. Nodes i and j are said to be neighbours if they can directly communicate with each other i.e., if $d_{ij} \leq d$. The transmission rate is R bits/seconds. In our model, we assume that average packet arrival rate at a mobile node is λ packets/sec, which includes packets generated by the mobile node itself as well as packets arrived from neighbour nodes given by the following equation:

$$\lambda = n \times \lambda_M \tag{4.6}$$

where λ_M is the number of packets generated by mobile node M itself and n consists of neighboring nodes including itself. Further, the size of each packet is assumed to be constant.

4.2. Average Hop Count

The average hop count is defined as the number of hops between an arbitrary source and destination in the network. The average number of hops traversed per packet n_{avg} in an ad hoc network is given by the following equation:

$$n_{avg} = \frac{E\{S\}}{d} \tag{4.7}$$

where $E\{S\}$ is expected distance between two nodes in a rectangular area and is given by [21] as follows:

$$\begin{aligned} E\{S\} = \frac{1}{15}\Bigg[& \frac{a^3}{b^2} + \frac{b^3}{a^2} \\ & + \sqrt{(a^2+b^2)} \times \left(3 - \frac{a^2}{b^2} - \frac{b^2}{a^2}\right)\Bigg] \\ & + \frac{1}{6}\Bigg[\frac{b^2}{a} \operatorname{arcosh} \frac{\sqrt{(a^2+b^2)}}{b} \\ & + \frac{a^2}{b} \operatorname{arcosh} \frac{\sqrt{(a^2+b^2)}}{a}\Bigg] \end{aligned} \tag{4.8}$$

where a and b denote its length and breadth respectively (in meters) for a terrain of network size $a \times b$. Here arcosh(x) is substituted in place of $\ln\left(x + \sqrt{(x^2-1)}\right)$.

4.3. End-to-End Delay Analysis

In mobile ad hoc network, end-to-end delay is the delay encountered by a packet, which is evaluated from the time the packet is generated to the time the source node receives an ACK indicating successful reception of the packet by the destination node. The end-to-end packet delay consists of MAC delay and transmission delay experienced at the source node as well as intermediate nodes. If D_{hop} is one hop delay and average number of hops traversed per packet between a source and a destination is given by n_{avg}, then the average end-to-end delay per packet i.e. D_{EtoE} can be obtained by the following equation:

$$D_{EtoE} = n_{avg} \times D_{hop} \tag{4.9}$$

where n_{avg} is obtained from equation 4.7.

4.4. Proposed Approach Analytical Validation

The comparison is done between the transmission delay obtained by analytical model with the results obtained by simulation analysis. The simulation parameters used are as given in Table 4. Table 3 shows average end-to-end delay per packet for different traffic load conditions obtained analytically using equation 4.9, and through simulation of our protocol. It is observed that there is only a slight deviation of results obtained through simulation and analytical approach.

Table 3. Analytical vs simulation results of end-to-end delay.

Traffic load in Mbps	End-to-end delay in milliseconds			
	Proposed Approach		MQAODV	
	Simulation	Analytical	Simulation	Analytical
0.1	228	222.23	257	265.48
0.2	262	251.23	289	299.89
0.3	346	336.13	357	357.56
0.4	365	362.72	379	365.56
0.5	389	387.15	398	379.78

5. Simulation Results and Discussion

The performance of our proposed approach is simulated using NS-2 simulator, for simulation purpose we have used the parameters given in Table 4. The network scenario consists of fixed number of nodes which is 50 in this case. The network topology consists of 1200 m × 800 m rectangular area. Nodes are generating constant bit rate (CBR) traffic with a packet of 512 bytes. The distributed coordination function IEEE 802.11 is used in MAC layer. The interface queue size is considered to be 50. The mobility model used was Random waypoint. All simulations were run for the duration of 700 seconds.

Table 4. Simulation parameters.

Parameters	Values
Number of mobile nodes	50
Number of corresponding hosts	6
Topology size	1200X800 m
Traffic type	CBR
Packet size	512 bytes
Wireless transmission range	250 m
Packet sending rate	5 packets/second
Mobility model	Random waypoint
Length of interface queue	50 packets
Link level layer	IEEE 802.11 DCF
Speed of a mobile node	10 m/s
Simulation time	700 seconds

5.1. Movement Model

The wireless channel used is based on two ray ground radio propagation model. The mobility model used in simulation is the Random Waypoint Model. Mobility model is generated using setdest utility. Setdest generates random positions of nodes in the network with specified mobility and pause time. The command used is:

/setdest –n ⟨num of nodes⟩ –p ⟨pause time⟩ –M ⟨max speed⟩ –t ⟨simu time⟩ –x ⟨max x⟩ –y ⟨max y⟩ > ⟨trace filename⟩

5.2. Traffic Model

All data packets are CBR (constant bit rate) and the size of each packet is 512 bytes. A packet-transmission rate of 20 Kbps is considered in this scenario. The connection pattern is generated using cbrgen. The command used is: ns cbrgen.tcl [–type cbr|tcp] [–nn nodes] [–seed seed] [–mc connections] [–rate rate]

5.3. Performance Metrics

The following performance metrics have been used to evaluate the performance of the proposed protocol

- *Packet Delivery Ratio* (*pdr*): It is the ratio between the received packets at the destination and number of packets generated by the sources.

$$pdr = \frac{\text{total number of received packets}}{\text{total number of sent packets by the source}} \times 100$$

- *Routing Overhead*: It is defined as the total number of control messages, including route discovery, sent out during the scenario.

$$R_{overhead} = \sum_{i=1}^{n} Overhead_i,$$

  where n is the number of nodes.

- *End-to-End Delay*: The time taken by the packet transmission from source node to a destination. This includes delays caused by buffering of data packets during route discovery, queuing at the interface queue, retransmission delays at the MAC propagation and transfer times.

5.4. Performance Evaluation of Delay Prediction using MATLAB

The neural network is realized using MATLAB software, back propagation training algorithm and random data division technique. The performance metric discussed in the following is used as a goodness measure to verify the correctness of neural network architecture used for our predicted results.

The correlation coefficient (r_c) given in equation 5.1 indicates how the predicted values vary from the target values. O_i represents the observed values for ithperiod during simulation taking the varying scenario as discussed in subsection 3.2. O_m represents the mean of the observed values during simulation. P_i and P_m indicate the predicted delay for ith period and mean of the predicted delay respectively obtained from the neural network. Table 4 reflects the goodness measure of accuracy of results obtained by proposed neural network model.

$$r_c = \frac{\sum_{i=1}^{n}(O_i - O_m)(P_i - P_m)}{\sqrt{\sum_{i=1}^{n}(O_i - O_m)^2 \sum_{i=1}^{n}(P_i - P_m)^2}} \tag{5.1}$$

The root mean squared error (rmse) in equation 5.2 reflects the deviation between the actual value observed from the simulation and predicted values obtained from neural network architecture. The significant feature of rmse is to uncover large errors rather than small errors.

$$\text{rmse} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(O_i - P_i)^2} \tag{5.2}$$

The mean absolute error in equation 5.3 indicates the absolute difference between the predicted and observed value. The error is considered in terms of absolute value of the error terms.

$$\text{mae} = \frac{1}{n}\sum_{i=1}^{n}|O_i - P_i| \tag{5.3}$$

The graph illustrating the mean squared error, see Figure 5, decreases as the number of iterations increases. When neural network is trained after several iterations it produces accurate delay prediction.

The error histogram shown in Figure 6 illustrates the stages of training, validation and testing along with the deviation between actual and target outputs. Once the artificial neural network is trained, the errors will be minimized. Accuracy of prediction is shown in Table 5, with the results obtained using 100 sets of data from NS-2 trace by varying simulation scenarios. Seventy

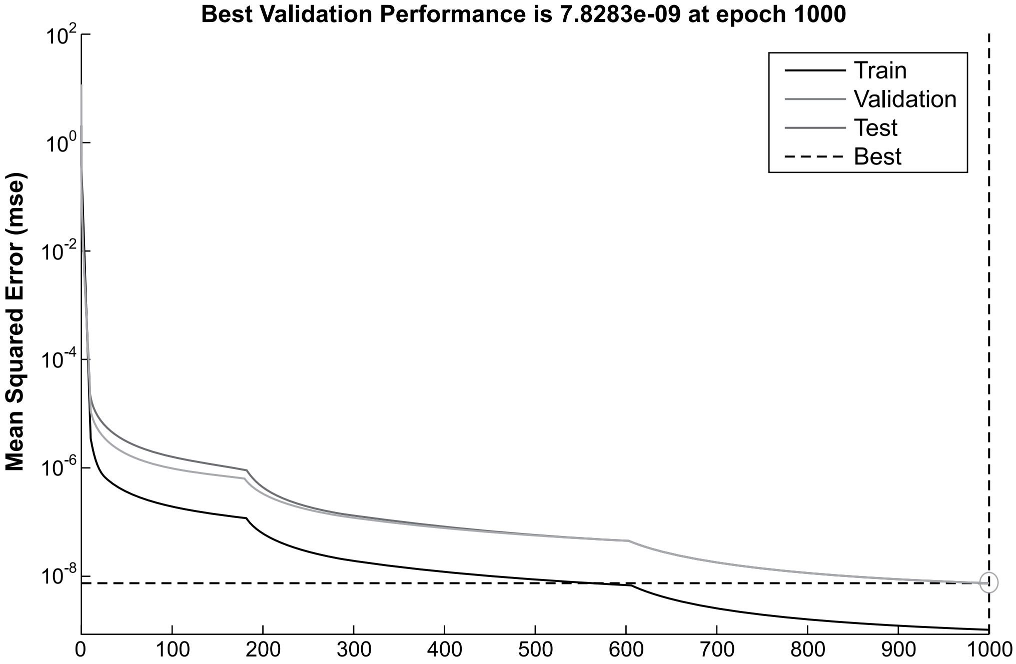

Figure 5. Mean squared error.

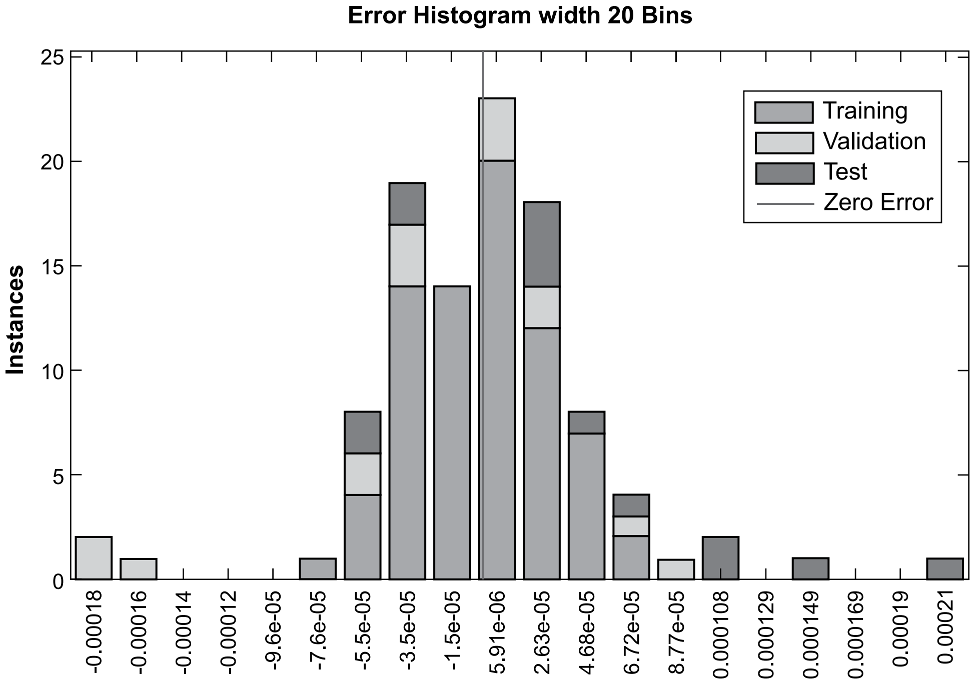

Figure 6. Error histogram reflecting deviation between targets and outputs.

sets of data are used for training multilayer feed forward neural network and remaining thirty sets of data are used for testing and validation.

Table 5. Goodness measure.

Performance Measures	
Root Mean Squared Error	0.0056045
Mean Absolute Error	0.0007854
Correlation Coefficient	0.9715465

The graph shown in Figure 7 illustrates the delay prediction obtained using neural network tool and actual delay realized through simulations. The predicted results do not show much deviation, hence our proposed neural network architecture is effectively trained and produces correct results. The comparison results are shown here to ensure that end-to-end delay predicted through multilayer feed forward neural network is similar to end-to-end delay results obtained through computer simulation. The predicted end-to-end delay using multilayer feed forward neural network architecture is used in QoS aware route discovery phase.

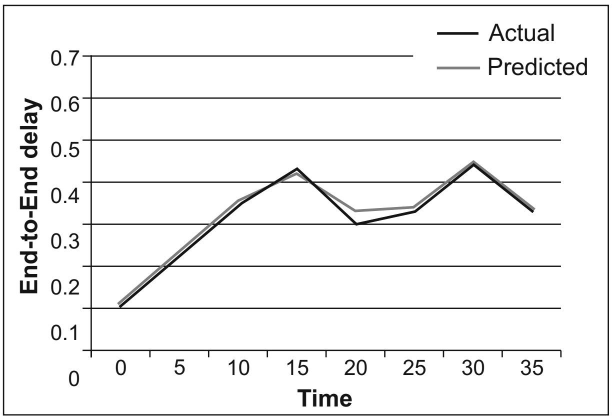

Figure 7. Actual vs predicted end-to-end delay.

5.4.1. Effect of Node Mobility

A simulation model consisting of 50 mobile nodes with 6 active sessions, each with 5 packets/second arrival rate and variable pause time was considered for simulation. To study the effect of mobility, pause time was varied from 0 to 700 seconds with steps of 100 seconds.

End-to-End Delay. With high mobility, the proposed approach has greater end-to-end delay compared to MQAODV [13] having pause time less than or equal to 300 due to high interference and frequent link breaks. As mobility decreases having pause time greater than 300 ms, end-to-end delay of our proposed approach also goes on decreasing, outperforming existing approaches. The average end-to-end delay with variable pause time is depicted in Figure 8.

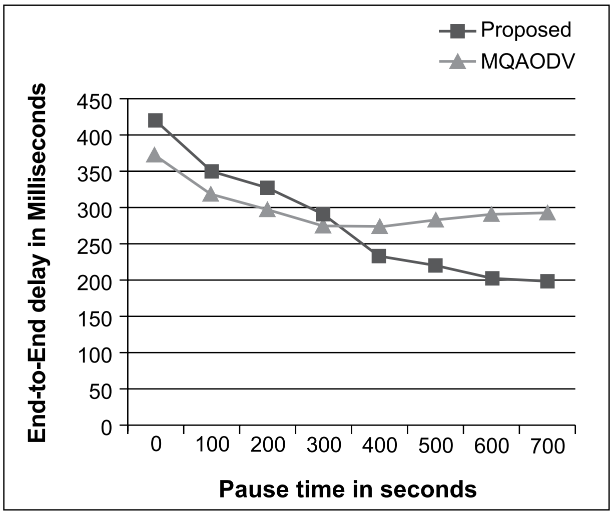

Figure 8. End-to-end delay vs pause time.

Routing Overhead. Routing overhead in our proposed approach is smaller than MQAODV, as the route maintenance procedure discussed in Algorithm 2 does not involve subsequent route error messages in case of link failure. Therefore the number of messages generated will be less. The routing overhead, with varying pause time of the mobile nodes, is depicted in Figure 9.

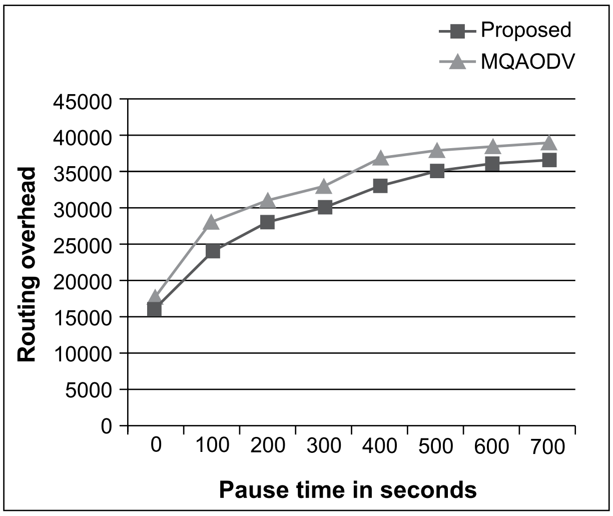

Figure 9. Routing overhead vs pause time.

Packet Delivery Ratio. With high mobility, the proposed approach reflects less packet delivery ratio as compared to MQAODV in the case of low pause time. With high mobility, interference from neighbouring nodes becomes high, and more collision and retransmission occur. As mobility decreases, pause time becomes greater than 300 seconds, and the proposed approach outperforms MQAODV because of its better route maintenance policy. At low mobility, few link breaks occur and the routes are balanced. The packet delivery ratio with variable pause time of the mobile nodes is shown in Figure 10.

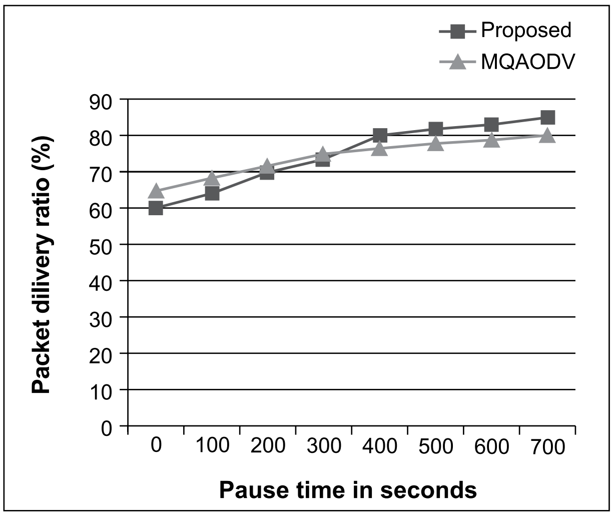

Figure 10. Packet delivery ratio vs pause time.

5.4.2 Effect of Traffic Load

In our model, different traffic loads were simulated varying from 0.1 to 0.6 Mbps. A simulation model consisting of 50 mobile nodes using random waypoint mobility model with 400 sec pause time and 5 packets/sec arrival rate were considered for simulation. This section discusses the comparative evaluation of our proposed approach and MQAODV.

End-to-End Delay. The end-to-end delay of our proposed approach is lesser than MQAODV in the situation of low and high traffic. The predicted end-to-end delay used in our scheme for identifying path having low delay latency. Furthermore, as the traffic load increases, interference with neighbour nodes increases, resulting in frequent route changes. Consequently, contention delay will be higher. However, our proposed approach performs better even in high traffic situations, since multilayer feed forward neural network architecture is trained with varying scenarios. The end-to-end delay with varying traffic load is shown in Figure 11.

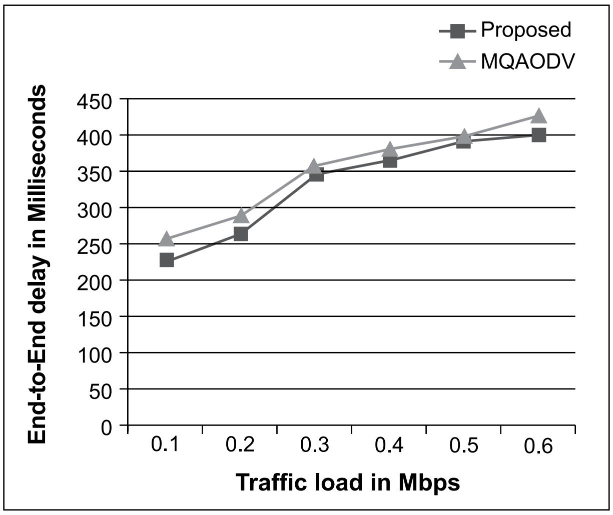

Figure 11. End-to-end delay vs traffic load.

Routing Overhead. The routing overhead in our proposed approach is lesser than MQAODV as it uses efficient link failure strategy to preserve the QoS requirements. At high traffic, as the load on the system is greater, the nodes become congested, which consequently increases the routing overhead. The routing overhead with varying traffic load is shown in Figure 12.

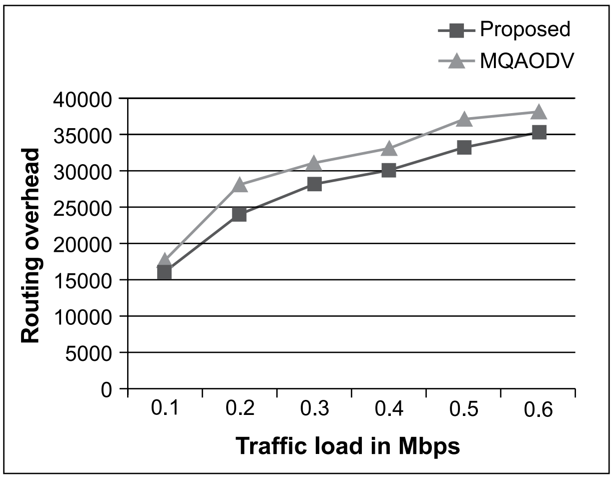

Figure 12. Routing overhead vs traffic load.

Packet Delivery Ratio. Figure 13 reflects the performance in our approach in terms of packet delivery ratio, the performance is shown better than MQAODV at high traffic.This approach has an efficient route recovery strategy so it results in less frequent route failure hence lesser packet drops, therefore enhanced throughput is achieved.

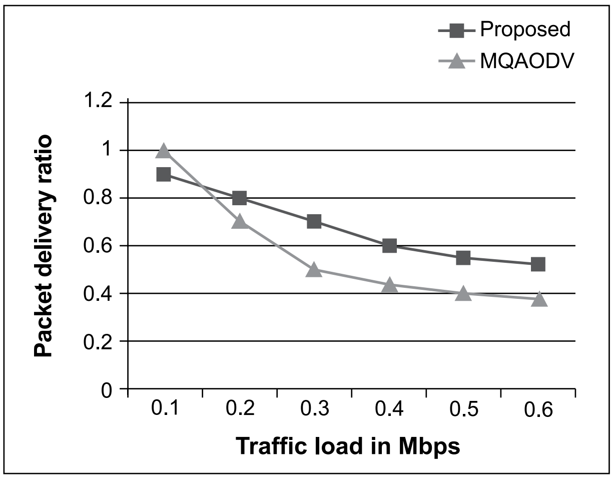

Figure 13. Packet delivery ratio vs traffic load.

6. Conclusion

The proposed approach is a novel QoS aware routing protocol which fulfils the stringent requirements of delay sensitive applications. This protocol improves the packet delivery ratio and throughput, which enhances the performance of the network. Our approach incorporates QoS enabled route discovery on the basis of prediction of end-to-end delay using artificial neural networks. Existing researches suggest calculation of individual node delay to identify the QoS aware routing path, which is a complex procedure as it involves estimation of contention delay and queueing delay of nodes associated with path discovery. The main aim of this paper is to provide an efficient QoS enabled routing protocol that fulfils the requirement of delay sensitive applications in wireless networks. Besides, link failure prediction mechanism is also integrated with routing protocol to quickly provide effective route maintenance strategy in the situation of node mobility.The stringent delay sensitive applications like multimedia applications can make use of this approach to fulfill their QoS requirements during transmission. The routing overhead is less in comparison with existing approaches, hence this proposed protocol provides scalability.

The future work may include prediction of the node mobility along with this existing protocol to provide stable routes. The enhancement of proposed approach requires consideration of the jitter factor for a particular traffic flow to support QoS requirements.

Acknowledgements

This project has been partially funded by the University Grant Commission (UGC), India under major research project vide letter no. 42-140/2013 dated 14th March 2013 and the Technical Education Quality Improvement Programme Phase II (TEQIP II).

References

[1] C. S. R. Murthy and B. S. Manoj, "Ad hoc Wireless Networks: Architectures and Protocols", Pearson Education, 2004.

[2] H. Qu *et al.* , "A new local repair scheme based on link breaks for ad hoc networks", Communication networks and Services research conference, Moncton, New Brunswick, Canada, 2009, pp. 364–371.

[3] P. Zhao *et al.*, "A new local repair scheme for mobile ad hoc network", *Proceedings of the 12th International Conference on Advanced Communication Technology*, Korea, 2010, pp. 1316–1320.

[4] A. Vere *et al.*, "Supporting service differentiation in wireless packet networks using distributed control", *IEEE journal on selected areas in communication*, vol. 19, pp. 2081–2093, 2012

[5] G. S. Campbell *et al.*, "Supporting service differentiation for real time and best effort traffic in stateless wireless ad hoc networks (SWAN)", *IEEE Transactions on Mobile Computing*, vol. 1, 2002, pp. 192–207.
http://dx.doi.org/10.1109/TMC.2002.1081755

[6] Li. Q. Liu and H. Jiang, "The Routing Protocol of AODV based on Link Failure Prediction", *Proceedings of the 9th IEEE International Conference on Signal Processing*, pp. 1993–1996, Beijing 2008.

[7] X. M. Zhang *et al.*, "Interference-Based Topology Control Algorithm for Delay-Constrained Mobile Ad Hoc Networks", IEEE Transactions on Mobile Computing, vol. 14, 2015, pp. 742–754.
http://dx.doi.org/10.1109/TMC.2014.2331966

[8] Li, Q. Liu, and H. Jiang, "The Routing Protocol of AODV based on Link Failure Prediction", *Proceedings of the 9th IEEE International Conference on Signal Processing*, pp. 1993–1996, Beijing, 2008.

[9] H. Kannichie and F. Kamoun, "Mobility Prediction in Wireless Ad Hoc Networks using Neural Networks", *Journal of Telecommunications*, vol. 2, pp. 95–101, 2010.

[10] P. Kuppusamy and K. Thirunavukkarasu, "A Study and Comparison of OLSR, AODV and TORA Routing Protocols in Ad Hoc Networks", *Proceedings of the IEEE 3rd International Conference on Electronics Computer Technology*, Kanyakumari, pp. 143–147, India, 2011
http://dx.doi.org/10.1109/icectech.2011.5941974

[11] H. Zafar *et al.*, "QoS-aware Multipath Routing Scheme for Mobile Ad Hoc Networks", *International Journal of Communication Networks and Information Security*, vol. 4, 2012, pp. 1–10.

[12] J. P. Singh, *et al.* , "Delay Prediction in Mobile Ad Hoc Network Using Artificial Neural Network", *Procedia Technology*, vol. 4, 2012, pp. 201–206, 2012.

[13] B. Surjeet *et al.*, "QoS Bandwidth Estimation Scheme for Delay Sensitive Applications in MANETs", *Journal of Communications and Network, Scientific Research*, vol. 5, pp. 1–8, 2013.

[14] J. Wang *et al.*, "QoS routing with mobility prediction in MANET", *Proceedings of the IEEE Pacific Rim Conference on Computers and Signal Processing*, Victoria, BC, Canada, pp. 357–360, 2001.

[15] S. Lee and M. Gerla, "AODV-BR: Backup Routing in Ad hoc Networks", Wireless adaptive mobility laboratory, Computer Science Department, University of California, pp. 1311–1316, 2000

[16] M. A Gafur *et al.*, "An Efficient Approach for Local Repairing in Mobile Ad Hoc Network", *Canadian Journal on Network and Information Security*, vol. 3, 2012, pp. 20–25.

[17] BonnMotion: A mobility scenario generation and analysis tool, University of Bonn, 2016. Available: http://www.sys.cs.uos.de/bonnmotion

[18] L. Qing *et al.*, "The Routing Protocol of AODV Based on Link Failure Prediction", *Proceedings of the 9th IEEE International Conference on Signal Processing*, Beijing, pp. 1993–1996, 2008.

[19] S. H. Bouk *et al.*, "Gateway Discovery Algorithm Based on Multiple QoS Path Parameters Between Mobile Node and Gateway Node", *Journal of communications and Networks*, vol. 14, 2012, pp. 434–442.

[20] P. Macharla *et al.*, "A QoS routing protocol for delay-sensitive applications in mobile ad hoc networks", *Proceedings of the 3rd International Conference on Communication Systems Software and Middleware and Workshops*, COMSWARE, pp. 720–727, Bangalore, 2008.
http://dx.doi.org/10.1109/comswa.2008.4554506

[21] C. Bettstetter and J. Eberspacher, "Hop Distances in Homogeneous Ad Hoc Networks", *Proceedings of the 57th IEEE Semiannual Vehicular Technology Conference*, Jeju pp. 2286–2290, South Korea, 2003.
http://dx.doi.org/10.1109/vetecs.2003.1208796

Received: February 2015
Revised: April 2016
Accepted: April 2016

Contact addresses:
Prakash Srivastava
Department of Computer Science and Engineering
Madan Mohan Malaviya University of Technology
Gorakhpur
Deoria Road, Gorakhpur-273 010 (U.P.)
India
e-mail: prakash_csrf@mmmut.ac.in

Dr. Rakesh Kumar
Department of Computer Science and Engineering
Madan Mohan Malaviya University of Technology
Gorakhpur
Deoria Road, Gorakhpur-273 010 (U.P.)
India
e-mail: rkcs@mmmut.ac.in

Prakash Srivastava received his B.Tech degree in Computer Science & Engineering in 2005 from I.E.R.T Allahabad (U.P), India and M.E degree in Computer Science & Engineering from NITTTR Chandigarh, India. Since July 2013, he has been a Research Scholar under TEQIP II in the Department of Computer Science & Engineering, MMM University of Technology, Gorakhpur. His areas of interests include computer networks, mobile ad hoc network, and MANET-Internet integration.

Dr. Rakesh Kumar is an Associate Professor in the Department of Computer Science Engineering at MMM University of Technology, Gorakhpur (U.P), India. He received his Ph.D. from Indian Institute of Technology, Roorkee, India in 2011. Currently, he is a principal investigator in a major research project sanctioned by the University Grant Commission, New Delhi, India. Dr. Kumar has supervised a large number of M. Tech Dissertations and guiding several Ph.D. students. He has published a large number of various research papers in international and national journals and conferences of high repute. He is a life member of CSI, ISTE and also a Fellow of IETE and I.E (India). His main interests lie in mobile ad hoc network, MANET- Internet integration, sensor network and network security.