SLBQT-DSR: SOURCE-BASED LOAD BALANCING ROUTING ALGORITHM UNDER CONSTRAINTS OF QUALITY OF TRANSMISION FOR MANET[*]

LE HUU BINH[1,2,3,a], VO THANH TU[4], NGUYEN VAN TAM[1,2]

[1]*Institute of Information Technology, Vietnam Academy of Science and Technology*

[2]*Graduate University of Science and Technology, Vietnam Academy of Science and Tech*

[3]*Faculty of Information Technology, Hue Industrial College*

[4]*Faculty of IT, College of Sciences, Hue University*

[a]*binh.lehuu@hueic.edu.vn*

Crossref
Similarity Check
Powered by iThenticate

Abstract. The routing technique under the constraints of the quality of transmision (QoT) in mobile ad hoc networks (MANET) has been studied widely recently. For this routing technique, QoT of the data transmission routes is improved. However, for the network models with mesh topologies such as MANET, The routing technique under the constraints of QoT can increase the bottlenecks due to the unbalanced traffic load. In this paper, we improve the DSR protocol by using the source-based load balancing in combined with the QoT constraint. The simulation results have shown that, the proposed algorithm outperforms the original algorithms in terms of the signal to noise ratio, bit error rate, blocking probability of the data packets and throughput.

Keywords. MANET; Load balancing routing; QoT aware routing; DSR.

1.   INTRODUCTION

With the trends in the development of communication network technologies, the wireless communications is one of the decisive solutions for the transmission technology of the telecommunications network in general and the computer network in particular. In the era of the fifth generation (5G) wireless network and Internet of things (IoT), there are several wireless network models to provide the practical applications such as MANET, wireless sensor networks (WSN), wireless mesh networks (WMN), and hybrid wireless networks [25]. Among these types, MANET is a network model that operates according to the principle of peer-to-peer networks, it does not depend on a preexisting infrastructure. A network model can be deployed easily and flexibly. Thus MANET is becoming more and more widely used in many fields, such as community network, enterprise network, home network, emergency response network, vehicle network, sensor network [18].

In order to improve the performance of the MANET, several published works have been focused on the control protocols for the data transmission from source to destination, in

---

[*]This paper is selected from the reports presented at the 11th National Conference on Fundamental and Applied Information Technology Research (FAIR'11), Thang Long University, 09 - 10/08/2018.
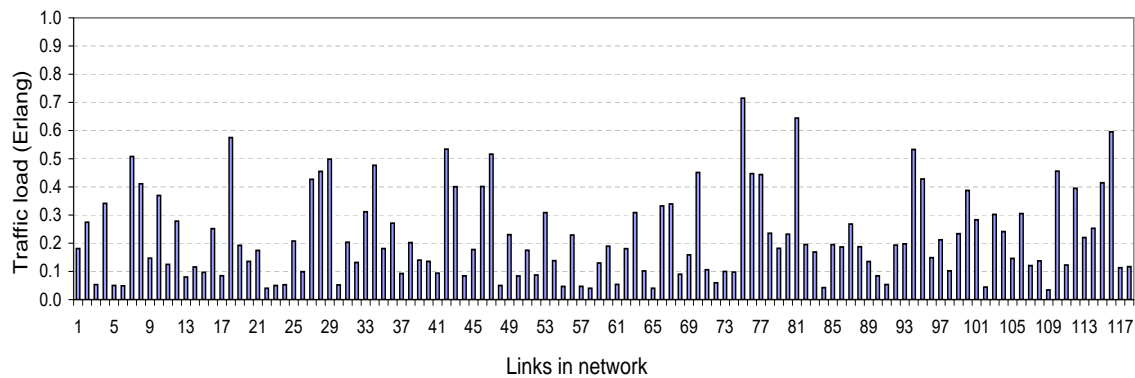
Figure 1. Traffic load distribute over all links in a MANET network topology with 60 nodes using the DSR routing protocol

which the routing protocols are the most studied. Most of published works related to routing protocols dedicate to improve the routing algorithms in order to decrease the probability of congestion, end-to-end delay, and increase the throughput of network [2, 11, 12]. In order to ensure the QoT of the data transmission routes, several works have proposed the routing algorithms that take into account the constraints of some QoT parameters [1, 7, 19, 24]. The proposed algorithms in these works attempt to find the best QoT route. This can reduce the blocking probability of data packets due to the unguaranteed QoT. However, for the network models with mesh topologies such as MANET, The routing technique under the constraints of QoT can increase the bottlenecks due to the unbalanced traffic load. In Figure 1, we analyze the traffic load that distribute to all connections using the DSR routing protocol for the case that the network size of 60 nodes. We can observe that, the traffic load unbalanceable distributes to all connections. There are some connections that its traffic load is greater than 0.7 Erlang, but there are also several connections that its traffic load only is less than 0.1 Erlang. This shows that the network resources are not used effectively.

In order to improve the efficiency of the network resources, the load balancing routing is one of the effective solutions that is used for MANET [6, 8, 10, 14, 15, 20, 21]. However, in the case of MANET with the wide area and high node density, the load balancing routing techniques can decrease the QoT due to the fact that the data transmission routes can pass through multiple hops. Thus, one problem to consider is how to combine harmony between QoT constraint routing and load balancing routing techniques, in order to find a set of routes
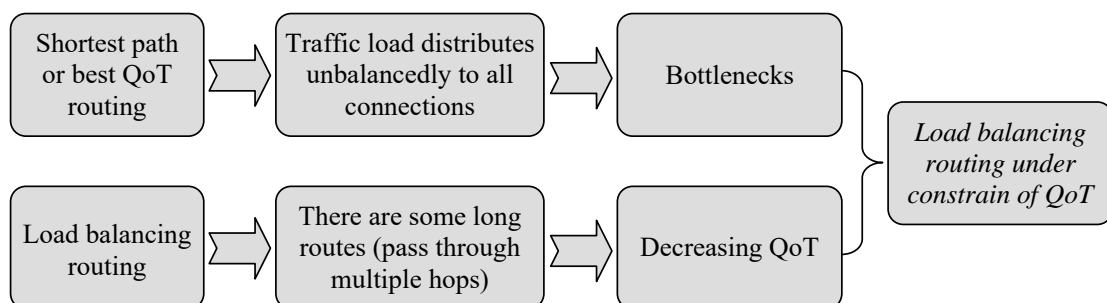

Figure 2. The idea of proposing load balancing routing in combined with QoT constraints

that load traffic load distribute balancedly for all connections in the network while satisfying the constraint conditions of QoT as shown in Figure 1. This is the research objective of this paper. We propose a load balancing routing algorithm that takes into account the constraints of QoT for the MANET network. The proposed algorithm was improved from the route discovery algorithm of DSR protocol, called SLBQT-DSR (Source-based Load Balancing and Quality of Transmission aware DSR). This research work is an extension to our previous work [3], where we introduced the LBQT-AODV algorithm that is the load balancing and QoT aware routing based on AODV in MANET.

The rest of this paper is organized as follows. Section 2 presents the basics of the load balancing routing technique in MANET. Section 3 presents our proposed algorithm. Section 4 presents the simulation results and discussions. Finally, concluding remarks and promising future work items are given in Section 5.

2. LOAD BALANCING ROUTING AND RELATED RESEARCH

The load balancing routing is a routing technique in which its object is to distribute balancedly the load traffic to all connections in the networks. For routing technique, the traffic bottleneck can be resolved both on nodes and links. Nowadays, the load balancing routing techniques are commonly used in wired and wireless networks in order to improve the efficiency of network resources utilization, especially for the mesh network models that MANET is a typical example.

In MANET network, there are two methods for implementing load balancing techniques which are the single-path load balancing and multiple-path load balancing. For the method of the single-path load balancing, the route cache of each node only stores the information of one route to the destination node. This is the unique route used for transmitting data. Thus, the load balancing must be done during route discovery. This method is usually done by setting the traffic load aware weight functions on the connections. Then, the routing algorithm selects the load balancing route based on this weight function. For the method of the multiple-path load balancing, the routing algorithm will find K smallest weight routes between each pair of source-destination nodes. Then, select one of these K routes to transmit the data. Depending on the route selection method for data transmission, the multiple-path load balancing method is classified into different categories, which are to select random paths, select sequential routes and select the route according to a given rule.

The load balancing routing technique in MANET has been implemented by several research groups recently. The authors of [15] have proposed a load balancing routing protocol for MANET namely LMP-DSR (Load balanced Multi-Path Dynamic Source Routing). LMP-DSR protocol is modified from original DSR protocol by using multiple paths routing instead of single path routing. The authors introduced two data structure, route cache and load table. The source node maintains up to five paths in its route cache, these paths are received from result of route discovery process. While choosing the route for data transmission, it keeps track of load balancing by checking the count of packets transmitted on a given route which is maintained in load table. By simulation method, the authors have proved that LMP-DSR protocol improved the network performance in terms of average delay, packet delivery ratio and throughput compared with original DSR protocol. In [6], a multi-level routing algorithm (MRA) has been proposed to balance the traffic load in wireless ad hoc network. MRA uses

an efficient method of selecting the intermediate nodes which have the enough resources and capability to reach the destination node. Simulation results have proved that the average end to end delay is reduced and connection establishment is very responsive. In [10], the authors have proposed a routing protocol called LBCAR (load balanced congestion adaptive routing). LBCAR protocol uses two metrics, traffic load density and link cost associated with a routing path in order to determine the congestion status, the route with low traffic load density and maximum life time will be selected for data transmission. The performance of the network using LBCAR protocol has been compared with ad hoc on-demand distance vector (AODV) routing protocol [16] and congestion adaptive routing protocol (CRP) [22] by simulation method. Simulation results have proved that, LBCAR protocol outperformed the AODV in terms of packet delivery ratio, average end-to-end delay, and normalized routing overhead. Compared with CRP, the packet delivery ratio and average end-to-end delay are almost the same in both routing protocols.

In [20], the authors have proposed a load balancing protocol for mobile ad hoc networks called FMLB (Fibonacci multipath load balancing), which uses the Fibonacci sequence for selecting the packet transmission routes. Mathematically, Fibonacci sequence is the sequence of numbers that starts with 0, and 1, and each number is the sum of the previous two numbers. The mathematical formula of the Fibonacci sequence is defined by [21]

$$f(n) = \begin{cases} 0 & \text{if } n = 0, \\ 1 & \text{if } n = 1, \\ f(n-2) + f(n-1) & \text{if } n \geq 2. \end{cases} \tag{1}$$

Suppose that there are k possible routes between source and destination nodes which are arranged in descending order according to their number of hops, thus, FMLB protocol will assign the weights of $f(1)$ to $f(k)$ for routes from 1 to k, respectively. The number of distributed packets for the routes is its corresponding Fibonacci value. The authors of [20] have set k to seven. The simulation results illustrated that the performance of the network using FMLB protocol is improved in terms of packet delivery ratio and end-to-end delay as compared to other well known protocols.

Another load balancing routing algorithm has been deployed in [14], where the authors have proposed a multi-path load balancing technique for congestion control (MLBCC) to efficiently balance the traffic load in MANET. In MLBCC protocol, the authors have proposed a congestion control mechanism and a load balancing mechanism. The congestion control mechanism detects the congestion in candidate node by using the arrival rate and the outgoing rate. The load balancing mechanism selects a gateway node by using path cost and link cost. Simulation results have proved that MLBCC protocol improves the performance of network in terms of control overhead, packet delivery ratio, average delay and packet drop ratio compared with FMLB protocol [20], stable backbone based multi-path routing protocol (SBMRP) [13].

Based on several published works as described above, we could comment that, the load balancing routing technique have attracted significant research interests and have been deployed in several different methods. The main objectives of proposed routing algorithms are to distribute balancedly the traffic load to all connections, thence reducing the blocking probability of data pakets in networks. However, in the case of MANET with the wide area and high node density, the load balancing routing techniques can decrease the QoT

due to the fact that the data transmission routes can pass through multiple hops. Therefore, the consideration of the constraints of QoT in load balancing routing algorithms is very essential. We propose a load balancing routing algorithm that takes into account the constraints of QoT for the MANET network, called SLBQT-DSR. SLBQT-DSR algorithm was improved from the route discovery algorithm of DSR protocol by using the principle of the source-based load balancing. The constraints of QoT are determined based on the cross-layer model in combined with the agent technology that we implemented in [5]. The details of the SLBQT-DSR algorithm are presented in the following sections.

3. SLBQT-DSR ALGORITHM

3.1. The idea of the proposed algorithm

The basic feature of the DSR protocol is that the route cache of each node stores the detailed information of each route from source to destination. Thus each node can determine the traffic load from it distributed to all connections in the network based on routing information in its route cache. Thence, when the source node receives the RREP packets for route discovery results, based on the routing information in its route cache, the source node can select a route so that the traffic load that distributes to all connections is most balanced. This is the idea of selecting the load balancing route of the SLBQT-DSR algorithm. In addition, in order to ensure that the selected routes satisfy the constraints of QoT, The RREQ packet processing process at the nodes in [5] is applied for the SLBQT-DSR algorithm to determine QoT constraints during route discovery.

3.2. Analytical model for SLBQT-DSR

In order to formulate SLBQT-DSR algorithm, the following symbols and notations are used.

Let $N_{sx} = \left[n_{ij}^{(sx)} \right]_{n \times n}$ be a matrix denoting the links of the route from node S to node X (r_{sx}), where each element $n_{ij}^{(sx)}$ is determined by

$$n_{ij}^{(sx)} = \begin{cases} 1 & \text{if } r_{sx} \text{ passes through connection } c_{ij}, \\ 0 & \text{otherwise.} \end{cases} \tag{2}$$

Let ρ_{sx} be the traffic load offers from node S to node X, $F_s = \left[f_{ij}^{(s)} \right]_{n \times n}$ be a matrix denoting the traffic load from node S distributes to all connections in the network. Thus F_s is determined by

$$F_s = \left[f_{ij}^{(s)} \right]_{n \times n} = \sum_{x=1}^{m|x \neq s} \rho_{sx} N_{sx}. \tag{3}$$

Consider the case when the node S wants to discover a new route to the node D. The SLBQT-DSR algorithm will broadcast the RREQ packet to discover the K routes satisfying the constraints of QoT and end-to-end delay (EED). K found routes which are denoted by a matrix $N_{sd}^{(k)} = \left[n_{ij}^{(sdk)} \right]_{n \times n}$, where each element $n_{ij}^{(sdk)}$ is determined according to (2).

In order to express the load balancing route which is selected in the K available routes, we define the variable $x_{sd}^{(k)}$ as follows

$$x_{sd}^{(k)} = \begin{cases} 1 & \text{if the route } k^{th} \text{ is selected,} \\ 0 & \text{otherwise.} \end{cases} \tag{4}$$

Then the matrix that denotes the traffic load from the node S to all connections in the network is transformed into

$$F_s' = \left[f_{ij}'^{(s)}\right]_{n \times n} = F_s + \rho_{sd} \sum_{k=1}^{K} x_{sd}^{(k)} N_{sd}^{(k)}. \tag{5}$$

From (5) we have

$$f_{ij}'^{(s)} = f_{ij}^{(s)} + \rho_{sd} \sum_{k=1}^{K} x_{sd}^{(k)} n_{sd}^{(k)}. \tag{6}$$

After determining the F_s' matrix, SLBQT-DSR algorithm is formulated as the following linear integer progeamming (ILP) problem

$$\min \left(\max_{\forall f_{ij}'^{(s)} \in F_s'} f_{ij}'^{(s)} \right) \tag{7}$$

subject to the following constraints

$$x_{sd}^{(k)}(x_{sd}^{(k)} - 1) = 0, \tag{8}$$

$$\sum_{k=1}^{K} x_{sd}^{(k)} = 1, \tag{9}$$

where (8) is the binary and integer constraint according to the definition of the variable $x_{sd}^{(k)}$ as in (4), (9) is the constraint of the route selection according to (4) which allows only to select one of the K available routes. By solving the ILP problem (7) with the constraints (8) and (9), we obtain the solution for $\{x_{sd}^{(k)}\}$, i.e. the load balancing route for data transmission that satisfies the constraints of QoT and EED.

To see more clearly the principle of the load balancing route discovery as the analytical model above, we consider an example as shown in Figure 3. At the moment the route cache of node 1 contains 4 records corresponding to 4 routes to destination nodes 2, 4, 5 and 6. These routes are structured in detail as $1 \to 4 \to 2, 1 \to 2, 1 \to 4 \to 2 \to 5$ and $1 \to 3 \to 5 \to 6$. According to definition (2), these routes are denoted by the matrices $N_{sx}(s = 1, \ x = 2, 4, 5, 6)$ as follows

$$N_{12} = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}, \qquad N_{14} = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}, \tag{10}$$

$$N_{15} = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}, \qquad N_{16} = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}. \qquad (11)$$

For simplicity in the calculation, consider the case where the traffic load from node 1 offers to each destination node is 1 Erlang ($\rho_{1x} = 1$). According to (3) we have the matrix denoting the traffic load from node 1 offers to all connections in the network at the moment as follows

$$F_1 = \left[f_{ij}^{(1)} \right]_{7 \times 7} = \rho_{12} N_{12} + \rho_{14} N_{14} + \rho_{15} N_{15} + \rho_{16} N_{16} = \begin{bmatrix} 0 & 0 & 1 & 3 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}. \qquad (12)$$

Assuming at the moment, node 1 wants to discovery a new route to the node 7. The SLBQT-DSR algorithm will broadcast the RREQ packet to discovery the K routes satisfying the constraints of QoT and end-to-end delay (EED). Consider the case of $K = 2$. After the broadcast of the RREQ packet, node 1 receives two RREQ packets corresponding to two routes that can be used for the data transmission. The structures of these two routes are $1 \to 4 \to 2 \to 7$ and $1 \to 3 \to 5 \to 2 \to 7$. These two routes are denoted by matrixs $N_{sd}^{(k)}$ as
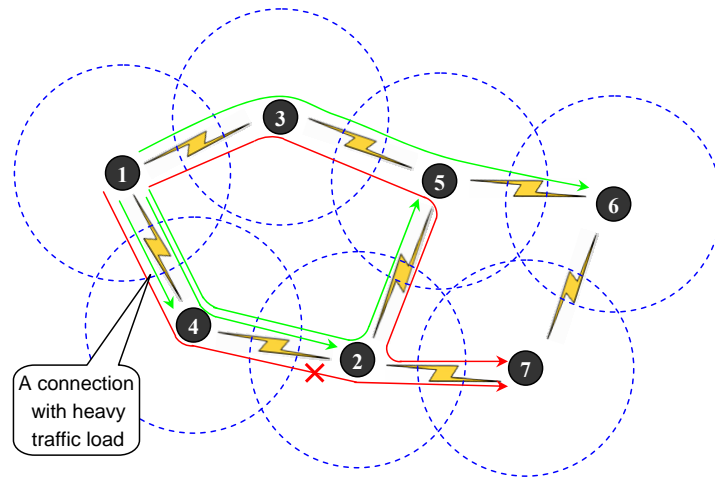
Figure 3. An example of discovering a balanced route using SLBQT-DSR algorithm

follows

$$N_{17}^{(1)} = \left[n_{ij}^{(171)}\right]_{n \times n} = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}, \tag{13}$$

$$N_{17}^{(2)} = \left[n_{ij}^{(172)}\right]_{n \times n} = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}. \tag{14}$$

The next task of the SLBQT-DSR algorithm is to select one of these two routes to transmit data. According to (7) we determine the objective function of the SLBQT-DSR algorithm as follows

$$\min \left(\max_{\forall f_{ij}'^{(1)} \in F_1'} f_{ij}'^{(1)} \right). \tag{15}$$

From (6), $f_{ij}'^{(1)}$ is determined by

$$f_{ij}'^{(1)} = f_{ij}^{(1)} + \rho_{17} \sum_{k=1}^{2} x_{17}^{(k)} n_{ij}^{(17k)} = f_{ij}^{(1)} + x_{17}^{(1)} n_{ij}^{(171)} + x_{17}^{(2)} n_{ij}^{(172)} \quad (\text{due to } \rho_{17} = 1). \tag{16}$$

From (16), (13), (14) and (12) we determine the components of the objective function (15) as follows

$$f_{13}'^{(1)} = 1 + x_{17}^{(2)}, \tag{17}$$

$$f_{14}'^{(1)} = 3 + x_{17}^{(1)}, \tag{18}$$

$$f_{25}'^{(1)} = 1, \tag{19}$$

$$f_{27}'^{(1)} = x_{17}^{(1)} + x_{17}^{(2)}, \tag{20}$$

$$f_{35}'^{(1)} = 1 + x_{17}^{(2)}, \tag{21}$$

$$f_{42}'^{(1)} = 2 + x_{17}^{(1)}, \tag{22}$$

$$f_{52}'^{(1)} = x_{17}^{(2)}, \tag{23}$$

$$f_{56}'^{(1)} = 1. \tag{24}$$

All remaining values of $f_{ij}'^{(1)}$ are equal to 0. According to (8) and (9), the constraints of the SLBQT-DSR algorithm in this case are determined by

$$\begin{cases} x_{17}^{(1)}(x_{17}^{(1)} - 1) = 0 \\ x_{17}^{(2)}(x_{17}^{(2)} - 1) = 0, \end{cases} \tag{25}$$

$$x_{17}^{(1)} + x_{17}^{(2)} = 1. \tag{26}$$

By solving the ILP problem (15) with the constraints (25) and (26), we obtain $x_{17}^{(1)} = 0$ and $x_{17}^{(2)} = 1$. Therefore, the value of the objective function (15) is

$$\min \Big(\max_{\forall f_{ij}^{\prime(1)} \in F_1^\prime} f_{ij}^{\prime(1)} \Big) = f_{14}^{\prime(1)} = 3 + x_{17}^{(1)} = 3. \tag{27}$$

For the above results, the route $1 \to 3 \to 5 \to 2 \to 7$ is chosen. Then the maximum load traffic on all connections in the network is 3 (connection from node 1 to node 4). For the topology as shown in Figure 3, we can observe that if the route $1 \to 4 \to 2 \to 7$ is chosen, the maximum load traffic on all connections in the network is 4 (also on the connection from node 1 to node 4). Thus the SLBQT-DSR algorithm has found the load balancing route.

3.3. Cross-layer model for SLBQT-DSR algorithm

To be able to perform the SLBQT-DSR algorithm according to the objective function (7) with the constraints (8) and (9), the network layer must be able to directly access to the information of the physical layer. This can only be performed by using cross-layer model [17, 24, 26]. In SLBQT-DSR algorithm, the cross-layer model is proposed as shown in Fig. 4, in which an agent is used for the exchange of the information of QoT between physical and network layers. This agent is called stationary agent (SA) which resides in every node and performs its tasks during route discovery as well as data transmission. The tasks if the SA include: (i) updating traffic load for the connections, and (ii) predicting the performance parameters which include the SNR and EED of a route. The model of the prediction of the performance parameters is illustrated as shown in Figure 5.

3.4. SLBQT-DSR algorithm

The principle of the route discovery of SLBQT-DSR is performed according to the Algorithm 1. There are two main differences between SLBQT-DSR and DSR algorithms, which are the processing to forward a RREQ packet at the nodes and the selecting of the load balancing route at source node.
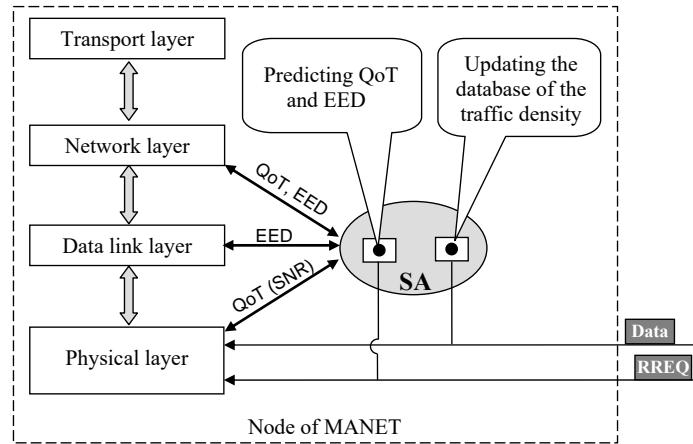

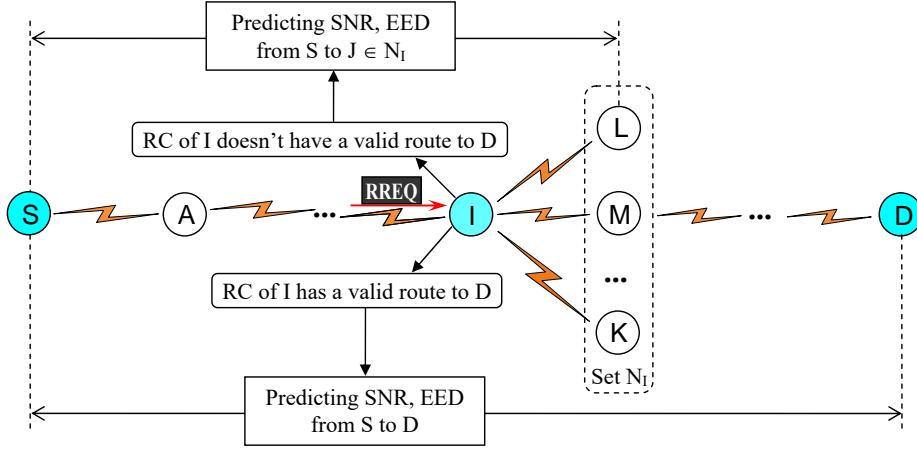*Figure 4.* Cross-layer model uses for SLBQT-DSR algorithm

Figure 5. Cross-layer model uses for SLBQT-DSR algorithm

3.4.1. Processing to forward the RREQ packet

Consider the case in which node I wants to forward a RREQ packet. According to the principle of the DSR algorithm, node I will broadcast the RREQ packet to all its neighbors. For the SLBQT-DSR algorithm, RREQ packet is only broadcasts to the nodes in the set Q_I, which is the set of the neighbors of node I satisfying the constraint conditions of QoT and EED. The set Q_I is determined according to Algorithm 2. When node I receives RREQ packet, SA at I first reads the information of SNR and EED from source node (S) to I ($\beta_{si}^{(r)}$ and $\tau_{si}^{(r)}$) stored in RREQ packet. Then, for each J is the neighbor of I, SA collects and predicts the information SNR and EED of the hop from I to J ($\beta_{ij}^{(h)}$ and $\tau_{ij}^{(h)}$), $\beta_{ij}^{(h)}$ is determined by the ratio of the signal power at node J to the noise power of hop h_{ij}, $\tau_{ij}^{(h)}$ is predicted based on the principle of the queuing at the nodes that we deployed in [5]. Thence $\beta_{si}^{(r)}$ and $\tau_{si}^{(r)}$ are determined as follows [4, 5]

$$\tau_{sj}^{(r)} = \tau_{si}^{(r)} + \tau_{ij}^{(h)}, \tag{28}$$

$$\beta_{sj}^{(r)} = \begin{cases} \min(\beta_{si}^{(r)}, \beta_{ij}^{(h)}) & \text{if relay type is decode and forward} \\ \left(\frac{1}{\beta_{si}^{(r)}} + \frac{1}{\beta_{ij}^{(h)}}\right)^{-1} & \text{if relay type is amplify and forward.} \end{cases} \tag{29}$$

After SA at node I predicted $\beta_{sj}^{(r)}$ and $\tau_{sj}^{(r)}$ according to (29) and (28) respectively, the constraint conditions of QoT and EED will be checked in order to determine whether the node J is included in the set Q_I.

3.4.2. Selecting of the load balancing route at source node

As discussed in Section 3.1 on the idea of the proposing SLBQT-DSR algorithm, the selection of the load balancing route is performed at the source node based on its route cache. After finding the K available route that satisfies the constraint conditions of QoT end EED according to Algorithm 1 above, the source node selects one of these K routes to transmit data so that the load traffic is distributed balancedly over the connections in the network. The process of selecting the load balancing route is performed according to Algorithm 3.

Algorithm 1 SLBQT-DSR algorithm

Input: A MANET topology; A new route discovery request from source node (S) to destination node (D).

Output: A load balancing route from S to D satisfies the constraint conditions of QoT and EED.

Method:

1: S creates the RREQ packet;
2: I ← S; $N_{RREP} = 0$;
3: **repeat**
4:     Determine set Q_I according to the Algorithm 2;
5:     I broadcasts RREQ packet to all nodes J∈ Q_I;
       {*At the nodes receive RREQ packet (node J)*}
6:     **if** (*J already received this RREQ*) **then**
7:         Discard RREQ;
8:     **else**
9:         **if** (*J is not the destination*) **then**
10:             **if** (*RC of I has a valid route to D*) **then**
11:                 SA at J predicts SNR and EED from S to D;
12:                 **if** (*SNR and EED satisfy the constraint of QoT*) **then**
13:                     Creating the RREP packet and send back to S;
14:                     $N_{RREP} \leftarrow N_{RREP} + 1$;
15:                 **else**
16:                     Update the reverse route to S into the RC of J;
17:                     Update the route from S to J into RREQ packet;
18:                     I←J;
19:                 **end if**
20:             **else**
21:                 Update the reverse route to S into the RC of J;
22:                 Update the route from S to J into RREQ packet;
23:                 I ←J;
24:             **end if**
25:         **else**
26:             Creating the RREP packet and send back to S;
27:             $N_{RREP} \leftarrow N_{RREP} + 1$;
28:         **end if**
29:     **end if**
30: **until** ($N_{RREP} = K$) **or** ($T_{wait} > timeout$)
31: **if** ($N_{RREP} > 0$) **then**
32:     S selects a load balancing route according to Algorithm 3;
33: **else**
34:     Refuse to request a new route discovery from S to D;
35: **end if**

Algorithm 2 Finding the set of the neighbors of I satisfying the constraints of QoT (Set Q_I) by SA

Input: A MANET topology; A RREQ packet arrives intermediate node I.

Output: Set Q_I.

Method:

 1: Read the information of $\beta_{si}^{(r)}$ and $\tau_{si}^{(r)}$ in RREQ packet;

 2: $Q_i \leftarrow \varnothing$;

 3: **for** (*each node J is the neighbor of node I*) **do**

 4:     Collect and predict the information of $\beta_{ij}^{(h)}$ and $\tau_{ij}^{(h)}$;

 5:     Predict $\tau_{sj}^{(r)}$ according to (28);

 6:     Predict $\beta_{sj}^{(r)}$ according to (29);

 7:     **if** $(\beta_{sj}^{(r)} > \beta_{req})$ **and** $(\tau_{sj}^{(r)} \leq \tau_{limit})$ **then**

 8:         $Q_i \leftarrow Q_i \cup J$;

 9:     **end if**

10: **end for**

Algorithm 3 Source node selects a load balancing route in the K available routes from S to D using SLBQT-DSR algorithm

---

Input: A MANET topology; K available routes from S to D satisfy the constraints of QoT and EED.

Output: A load balancing route from S to D.

Method:

 1: Based on the information in the route cache of S, construct the traffic distribution matrix $F_s = \left[f_{ij}^{(s)} \right]_{n \times n}$ according to (3);

 2: Based on the information of K available routes, construct the traffic distribution matrix $F_s' = \left[f_{ij}'^{(s)} \right]_{n \times n}$ according to (5);

 3: Construct the ILP problem according to the objective function(7) subject to the constraints of (8) and (9);

 4: Solving the ILP problem;

 5: Select the load balancing route based on the results of the ILP problem solving;

 6: Update the information of the found route into the route cache of S;

---

4. SIMULATION RESULTS AND DISCUSSION

4.1. Simulation scenarios

In order to evaluate the performance of SLBQT-DSR algorithm, we use the simulation method based on OMNeT++ discrete event simulator [23]. LBQT-DSR algorithm is compared with DSR [9] and QTA-DSR algorithms [5] in terms of the SNR, BER, blocking probability of the data packet, and network throughput. The simulation assumptions are presented in Table 1. In addition, for each simulation scenario, the number of source nodes

is set to 30% of the the number of nodes, the remaining nodes are the destination nodes. For example, for the scenarios of 30 nodes, there are 10 source nodes and 20 destination nodes.

Table 1. Simulation parameters

Parameters	Setting	Parameters	Setting
Network Size	1000 m × 1000 m	BER threshold	10^{-6}
Modulation technique	256-QAM	Minimum Required SNR	23.5 dB
MAC protocol	802.11ac	Noise model	Thermal noise
Number of nodes	From 20 to 50	Temperature	300^0K
Transmit Power	19.5 dBm	Mobility model	Random - Waypoint
Receiver Sensitivity	-68 dBm	Speed of nodes	5 - 20 m/s
Transmission Range	250 m	Time simulation	40 minutes

4.2. Performance analysis

4.2.1. Evaluation of SNR and BER

The obtained results in Figure 6 show the SNR of all routes for the cases that DSR and SLBQT-DSR algorithms are used. In this case, the networks size is 50 nodes, the channel bandwidth is 40 MHz and the average mobility speed is 20 m/s. According to the simulation scenarios presented in Section 4.1, the minimum required SNR for ensuring QoT is 23.5 dB. From the curves in Figure 6a, we can observe that for DSR algorithm, there are many routes do not satify the constraint conditions of QoT due to the fact that its SNR is less than required SNR. These routes account for 1.58%. This is the cause of the increasing the packet blocking probability due to the unsatisfactory constraint conditions of QoT. For SLBQT-DSR algorithm, SNR values improved significantly compared with DSR algorithm. This is more clearly visible from Figure 6b, where we analyzed the SNR of all routes when SLBQT-DSR is used. Most of the routes satify the constraint conditions of QoT due to the fact that the QoT constraints were considered during the route discovery.

Since SNR decreases in case of the SLBQT-DSR is used, BER increases as shown in Figure 7, where, we plot the BER of the routes as a function of the simulation times. In
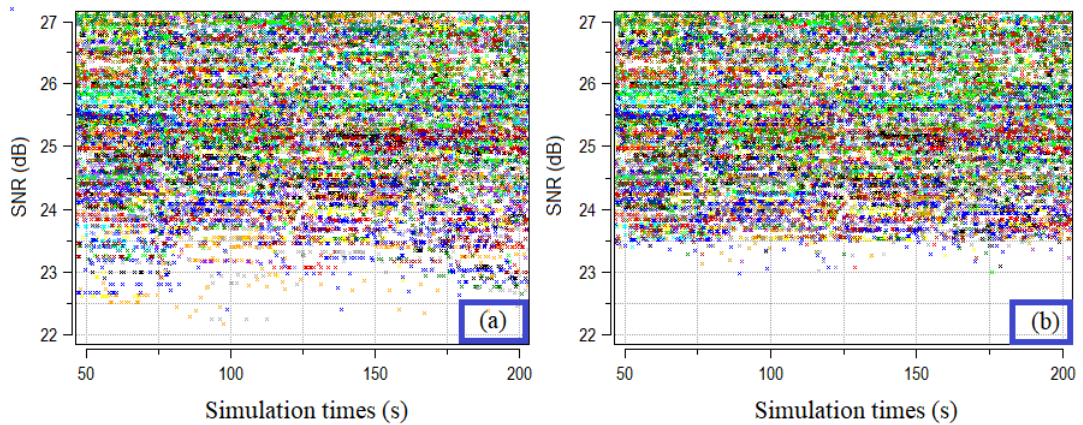

Figure 6. The comparison the of SNR of (a) DSR and (b) SLBQT-DSR algorithms
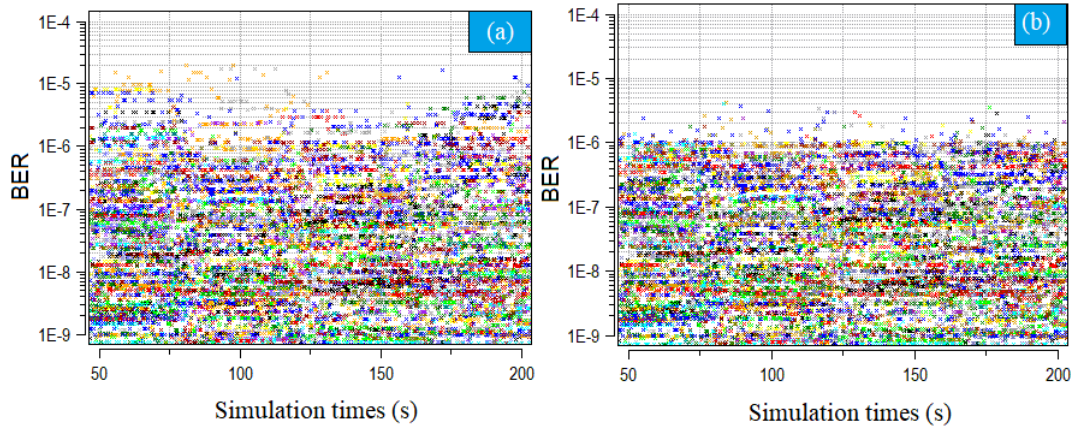
Figure 7. The comparison of the BER of (a) DSR and (b) SLBQT-DSR algorithms

this case, the network size is 50 nodes, the channel bandwidth is 40 MHz and the average mobility speed is 20 m/s. We can observe that, for DSR algorithm (Figure 7a), there are many routes that its BER is greater than 1e-6. These routes do not satify the constraint conditions of QoT. For SLBQT-DSR algorithm (Figure 7a), As the constraint conditions of QoT ware considered during route discovery, most of the routes are that its BER is within allowable limits, i.e it is less than 1e-6.

4.2.2. Evaluation of blocking probability

The blocking probability of data packets (BPD) is an important parameter with regard to the network performance. In our context, BPD is given by

$$BPD \;=\; \frac{N_b}{N_g}, \tag{30}$$

where N_g and N_b are the number of data packets are generated and the number of data packets are blocked in the overall network respectively. N_b includes two components, blocking due to the congestion of the traffic load and blocking due to unsatisfactory constraint conditions of QoT.

The obtained results in Figure 8 show the difference in the BPD of DSR, QTA-DSR and SLBQT-DSR, where, we plot the BPD as a function of the traffic load in Erlang. These results are simulated for the case that the number of node is 30, the average mobility speed of the nodes is 20 m/s and the channel bandwidth is 40 MHz. The curves in Figure 8 show that, if the traffic load is less than 0.8 Erlang, BPD of both QTA-DSR and SLBQT-DSR are almost similar. In case of heavy traffic load, i.e from 0.8 to 1 Erlang, BPD of LBQT-DSR algorithm reduced significantly compared with that of QTA-DSR algorithm. Considering the case of the traffic load is 1 Erlang, BPD of QTA-DSR and LBQT-DSR algorithms are 0.031 and 0.025, respectively. Thus BPD of LBQT-DSR algorithm reduced by 0.0053 compared with that of QTA-DSR algorithm.

Next, we analyze the case of the variable network size. The results obtained as shown in Figure 9, where, we compared the BPD of the algorithms DSR, QTA-DSR and SLBQT-DSR. These results are implemented for the case that the average mobility spend of the nodes is
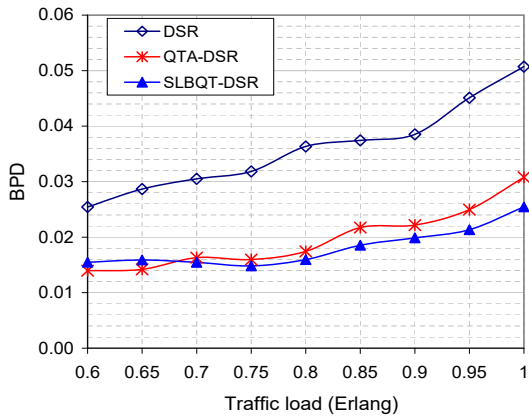
Figure 8. The comparison BPD versus the traffic load of DSR, QTA-DSR and SLBQT-DSR algorithms
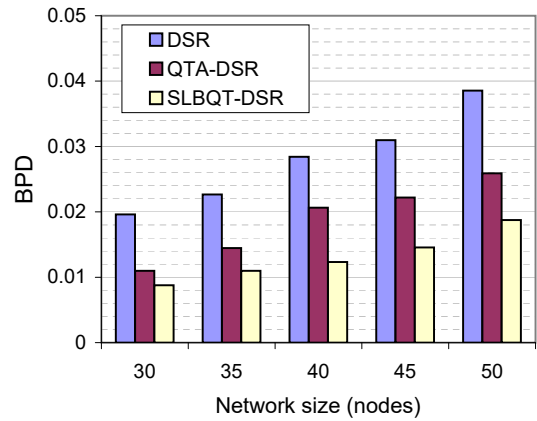
Figure 9. The comparison BPD versus the network size of DSR, QTA-DSR and SLBQT-DSR algorithms

5m/s, the channel bandwidth is 40 MHz, and the traffic load is 0.95 Erlang. We can observe that, for all algorithms, the larger the network size, the higher the PBD, due to the fact that QoT decreases according to the network size. However, BPD of SLBQT-DSR algorithm is always less than the other two algorithms. The average PBD reduction rate by 53.54% and 29.26% compared with DSR and QTA-DSR algorithms, respectively.

For the cases of the variable mobility speed, PBD of all three algorithms increases according to the mobility speed of the nodes. This is more clearly visible from Figure 10. These results are implemented for the case that the networks size and traffic load are 40 nodes and 0.95 Erlang, respectively. We can observe that, PBD reduced significantly in the cases of QTA-DSR and SLBQT-DSR algorithms is used. In particular, the SLBQT-DSR algorithm has the lowest BPD value, with a mean decrease of 46.72% compared to the DSR algorithm.
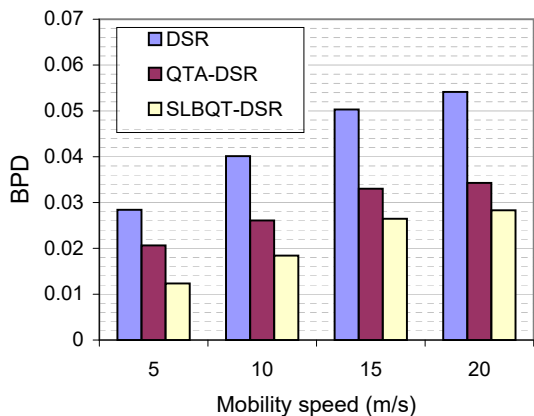

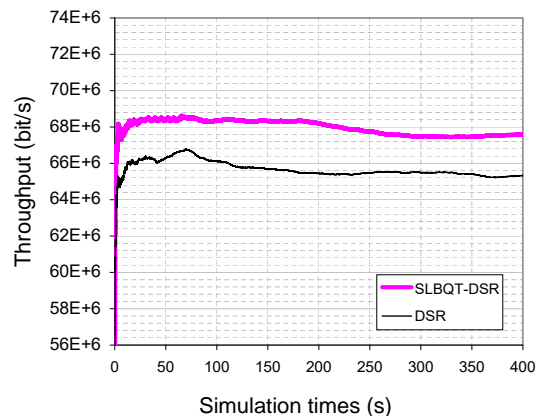*Figure 10.* The comparison BPD versus the mobility speed of DSR, QTA-DSR and SLBQT-DSR algorithms

Figure 11. The comparison throughput versus the simulation times of DSR and SLBQT-DSR algorithms

4.2.3. Evaluation of throughput

In this section, we analyze the average receiving throughput at each node. In Figure 11, we plot the average throughput of each node as a function of simulation times for the case that the network size is 50 nodes, the average mobility spend of the nodes is 5 m/s, and the traffic load is 0.9 Erlang. The curves in Figure 11 show that, the throughput of SLBQT-DSR algorithm increases significantly compared with DSR algorithm. The average throughput of SLBQT-DSR and DSR algorithms are 67.385 and 65.482 Mbps, respectively. Thus, average throughput of SLBQT-DSR increases 1.903 Mbps compared with DSR algorithm.

5. CONCLUSION

The load balancing routing is one of the optimal routing techniques to improve the network performance of MANET network. With load balancing routing techniques, the local congestion at some connections as well as intermediate nodes is minimized due to the fact that traffic is distributed evenly for all connections in the network. However, for the wide area and high node density MANET, the load balancing routing techniques can decrease the QoT and increase the EED due to the fact that the data transmission routes can pass through multiple hops. In this paper, we proposed a load balancing routing algorithm for MANET taking into account the constraint of QoT and EED. The proposed algorithm is called SLBQT-DSR which is improved from the route discovery algorithm of the DSR protocol. The performance of SLBQT-DSR algorithm is demonstrated by the simulation method using OMNeT++. The simulation results have shown that the proposed algorithm can improve the network performance in terms of SNR, BER, blocking probability and throughput compared with DSR and QTA-DSR algorithms.

In the near future, we continue to study the load balancing routing techniques taking into account the QoT with respect to the other routing protocols of MANET, specially for multi-channel MANET networks.

REFERENCES

[1] F. Alnajjar, "SNR/RP aware routing model for MANETs," *Journal of Selected Areas in Tele-communications (JSAT)*, pp. 40–48, 2011.

[2] H. Bakhsh and M. Abdullah, "ARPM: Agent-based routing protocol for MANET," *International Journal of Internet Protocol Technology*, vol. 3, no. 2, pp. 136–146, 2008.

[3] L. H. Binh, V. T. Tu, and N. V. Tam, "A load balancing routing algorithm under constraint of quality of transmission in MANET," in *Proceedings of the 11th National Conference on Fundamental and Applied Information Technology Research (FAIR'11)*, 2018.

[4] L. H. Binh and V. T. Tu, "QTA-AODV: An improved routing algorithm to guarantee quality of transmission for mobile Ad Hoc networks using cross-layer model," *Journal of Communications*, vol. 13, no. 7, pp. 338–349, 2018.

[5] L. H. Binh, V. T. Tu, and N. V. Tam, "Quality of transmission aware routing in adhoc networks based on cross-layer model combined with the static agent," *Journal of Computer Science and Cybernetics*, vol. 32, no. 4, pp. 351–366, 2016.

[6] K. J. Dsouza and S. M, "MRA: Multi-level routing algorithm to balance the traffic load in wireless Ad hoc network," in *Proceedings of National Conference on Parallel Computing Technologies (PARCOMPTECH)*, Feb 2015, pp. 1–5.

[7] M. Elshaikh, M. F. M. Fadzil, N. Kamel, and C. M. N. C. Isa, "Weighted signal-to-noise ratio average routing metric for dynamic sequence distance vector routing protocol in mobile Ad-Hoc networks," in *Proceedings of IEEE* 8th *International Colloquium on Signal Processing and its Applications (CSPA)*, 2012, pp. 329–334.

[8] S. G. and R. A., "Efficient and secure routing protocol for wireless sensor networks through SNR based dynamic clustering mechanisms," *Journal of Communications and Networks*, vol. 15, no. 4, pp. 422–429, 2013.

[9] D. Johnson, Y. Hu, and D. Maltz, "The dynamic source routing protocol (DSR) for mobile Ad Hoc networks for IPv4," *RFC4728*, [Online]. Available: http://www.rfc-editor.org/rfc/rfc4728.txt. [Online]. Available: http://www.rfc-editor.org/rfc/rfc4728.txt

[10] J. Y. Kim, G. S. Tomar, L. Shrivastava, S. S. Bhadauria, and W. H. Lee, "Load balanced congestion adaptive routing for mobile Ad Hoc networks," *International Journal of Distributed Sensor Networks*, vol. 2014, 2014, [Online]. Available: http://dx.doi.org/10.1155/2014/532043.

[11] E. Kulla, M. Ikeda, L. Barolli, F. Xhafa, M. Younas, and M. Takizawa, "Investigation of AODV throughput considering RREQ, RREP and RERR Packets," in *Proceedings of* 27th *International Conference on Advanced Information Networking and Applications*, Barceona, Spain, Mar 2013, pp. 169–174.

[12] H. Li, C. Dan, W. Min, and L. Shurong, "Mobile agent based congestion control AODV routing protocol," in *Proceedings of the* 4th *International Conference on Wireless Communications, Networking and Mobile Computing*, 2008, pp. 1–4.

[13] S. Mallapur and S. R. Patil, "Route stability based on demand multipath routing protocol for mobile Ad Hoc networls," in *Proceedings of Interational Conference on Communication and Signal Processing*, India, April 2014, pp. 1859–1863.

[14] S. V. Mallapur, S. R. Patil, and J. V. Agarkhed, "Load balancing technique for congestion control multipath routing protocol in MANETs," *Wireless Personal Communications*, vol. 92, no. 2, pp. 749–770, 2017.

[15] L. K. Malviya and D. Tiwari, "LMP-DSR: Load balanced multi-path dynamic source routing protocol for mobile Ad-Hoc network," in *Proceedings of Fourth International Conference on Computing, Communications and Networking Technologies (ICCCNT)*, July 2013, pp. 1–5.

[16] C. Perkins, E. B. Royer, and S. Das, "Ad hoc on-demand distance vector (AODV) routing," *RFC 3561*, [Online]. Available: https://www.ietf.org/rfc/rfc3561.txt.

[17] V. R, M. V, and S. V, "Cross-layer approach among physical, MAC and routing layer in a shadowing environment," *International Journal of Engineering and Technology (IJET)*, vol. 5, no. 2, p. 1, 2013.

[18] S. K. Sarkar, T. G. Basavaraju, and C. Puttamadappa, *Ad Hoc Mobile Wireless Networks - Principles, Protocols, and Applications*.   Taylor & Francis Group, LLC, 2008.

[19] S. Srivastava and A. K. Daniel, "An efficient routing protocol under noisy environment for mobile ad hoc networks using fuzzy logic," *International Journal of Advanced Research in Artificial Intelligence*, vol. 2, no. 6, pp. 34–39, 2013.

[20] Y. Tashtoush, O. Darwish, and M. Hayajneh, "Fibonacci sequence based multipath load balancing approach for mobile Ad Hoc networks," *Ad Hoc Networks*, vol. 16, pp. 237–246, 2014.

[21] Y. M. Tashtoush and O. A. Darwish, "A novel multipath load balancing approach using fibonacci series for mobile Ad Hoc networks," *International Journal of Computer Theory and Engineering*, vol. 4, no. 2, pp. 220–225, 2012.

[22] D. A. Tran and H. Raghavendra, "Congestion adaptive routing in mobile Ad Hoc networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 17, no. 11, pp. 1294–1305, 2006.

[23] A. Varga, *OMNeT++ Discrete Event Simulation System, Release 4.6*, 2015, [Online]. Available: http://www.omnetpp.org.

[24] A. Yadav and T. Sharma, "Cross-layer approach for communication in MANET," *International Journal of Computer Science and Mobile Computing*, vol. 4, no. 3, pp. 285–292, March 2015.

[25] Y. Zhang, J. Luo, and H. Hu, *Wireless Mesh Networking - Architectures, Protocols and Standards*. Taylor & Francis Group, LLC, 2007.

[26] M. M. zoulikha and B. Amal, "Cross-layer approach among physical, MAC and routing layer in a shadowing environment," *Ad-Hoc and Sensor Wireless Networks*, vol. 21, no. 1-2, pp. 101–119, 2014.