A Survey on Load Management Schemes in Routing Algorithms of Multi-Hop, Multi-Gateway Wireless Mesh Networks

Aditya Saxena
Research Scholar, Electronics & Communications Engineering Department, Maulana Azad National Institute of Technology (MANIT), Bhopal, 462003, India

Jyoti Singhai, PhD
Professor, Electronics & Communication Engineering Department, Maulana Azad National Institute of Technology (MANIT), Bhopal, 462003, India

Deepak Raghuvanshi, PhD
Assistant Professor,Electronics & Communication Engineering Department, Maulana Azad National Institute of Technology (MANIT), Bhopal, 462003, India

ABSTRACT

Wireless Mesh Networks (WMN) is emerging as a promising technology platform for future generation wireless networking. It has distinct technical advantages over other available wireless technologies. However, the increasing requirement of smart users drives heavy data traffic in WMNs and hence their load management becomes crucial for efficient network operation. Smarter load management also leads to higher WMN reliability, enhanced throughput, scalability and network availability. This paper provides a detailed investigation of various load management techniques for WMNs proposed so far that will enable us to better understand load balancing and associated challenges.

Keywords
Wireless Mesh Networks; Load Management; Routing Algorithm; Load Balancing.

1. INTRODUCTION

A wireless mesh network consists of interconnected nodes in the form of a cluster and a dedicated gateway per cluster. These clusters in turn are connected with each other and form a complete, big mesh network. The prime purpose of this network is to communicate between various clients by transfer of data packets. The data packets can be sent to the outside world/other clients which are hooked to the other networks via gateway, which has internet accessibility. The nodes in WMN are the data packet router which facilitates the movement/forwarding of packets in multi-hop fashion so as to reach the final receiver.

The WMN is an innovative technology which can provide internet access to a very wide population in a very cost-effective way. However, to enable a WMN to work efficiently, there are various criteria which a WMN needs to fulfill to make it practical. In WMN, nodes communicate in a multi-hop way. The data traffic needs to be directed in such a way that not only does the data reach the destination fast but it also takes into account all the possible alternative connected links. At times, user loads the network with heavy data such as hooking to a video conference and sending big data files like videos, pictures or other big data files. These conditions tend to load the network heavily and hence load management becomes highly critical. A good network load management serves this peak requirement well and avoids overloading of node/ gateway node. A WMN also needs to be self-healing in

case of node failures by identifying and establishing alternative routes and bounce back to original routing when node is up and running again. The WMN should also be able to merge different networks and work in expanded fashion and have quick scalability.

Routing is a very challenging issue in WMNs. It is a difficult task for a good routing algorithm to successfully meet various criteria simultaneously. Not only should it have self-organizing and self-healing properties but also needs to adapt itself to bandwidth limitation, manage power efficiently and utilize the multi-hop properties for better load management (especially peak traffic handling).

The routing in WMNs often leads to problems like large area of flooding (duplicate packets), empty set of neighbors (in case of Greedy Forwarding methods), high power consumption, interference due to external factors and load-balancing issues.

The WMN application locations are carefully studied at the time of network setup to identify the issues and accordingly the appropriate routing algorithm category is chosen for optimum network operation.

This paper presents critical review and analysis of load management schemes in routing algorithms applicable to multi-hop, multi-gateway WMNs.

In this paper, section 2 presents load management and load balancing routing algorithms and in section 3, the performance summary of the available algorithms and summary of conclusion is presented.

2. ROUTING ALGORITHMS FOR MULTI-HOP, MULTI-GATEWAY WMNS

In the past, the routing algorithms for WMNs were classified according to centralized/ distributed and proactive/ reactive categories. However, in recent years, the WMNs have much wider applications. Hence, the routing algorithms in literature[1] have evolved to cover the unique challenges they are facing and hence introduced many new routing categories namely Geographical, Hierarchal, Multi-path, Power-aware , Hybrid routing algorithms, etc.

The Geographical routing exploits the Global Positioning System (GPS) which provides the location of the node. The

sender uses the destination's geographical location for packet delivery. In this case, the sender need not be fully aware of the network topology, but the hardware complexity is increased at nodes.

In Hierarchal routing, the nodes are divided into groups called clusters. Each cluster has one or more cluster head (node) called gateway. The gateway is the access point between two clusters. The other nodes which are not gateways can only communicate within their cluster. Thus congestion is reduced but at the cost of energy depletion of gateways.

In Multi-path routing, multiple paths are used to route the data packets from source to destination. This increases the resource utilization and provides redundancy and robustness to the network.

In Power-aware routing, the prime issue is power management. The nodes are either located remotely so that battery replacement is not easy or mobility of the node is such that it requires long battery life. The low battery power restricts the transmission range of the node as energy E is proportional to d^α (where d is the transmission distance and α is the attenuation factor).

Hybrid routing algorithm adopts both proactive and reactive approach based on the network conditions for packet delivery. Hence, it combines the advantages of both proactive and reactive routing.

This section discusses the different aspects of available routing algorithms of this category with advantages and limitations of each algorithm.

2.1 Fixed partition load sharing (FPLS) in WMN

In the FPLS algorithm,[2] each gateway (GW) caters to a unique set of mesh routers (MRs). Then, the WMN is partitioned into fixed clusters around the GWs on fair share basis. This operation happens only once at the time of network boot up (offline). This has an advantage over Legacy system of partitioning in which the partition is reviewed and changed again and again and MRs are reallocated at the time of mesh overload during network operation. The Legacy system of partitioning is atomic (network operation is halted at the time of reallocation) while fixed partitioning is non-atomic. Due to this reason, the Legacy partition method is slow and causes delays in packet transmission.

Once the network operation begins in FPLS system, the load on the GWs is monitored continuously and if the GW is overloaded, the MR is transferred to the neighbor GW for load management. Traffic halts only for a duration of a sending GW (which is overloaded) transferring an MR to a receiving GW (which is less loaded neighbor).

In case all the GWs are overloaded and transfer of MR to its neighboring GW is not possible, the MR is transferred to wired network.

The load management scheme functions by adopting the technique of representing the whole WMN in the matrix form of graph G. Then, the Adjacency matrix A, Incidence matrix B and Cycle matrix C of graph G are generated. The WMN is thus partitioned and based on this partition, matrices A', B' and C' of graph G' are generated.

The WMN is configured as per the partitions and the network starts functioning by booting up. Once, the network is up and functioning, the load is monitored continuously. The network

operation is stable if the condition $\| R_i \|^2 \le \sum_{i=1}^{k} L_i U_i$ is met, where R_i is the real time performance of ith partition, U_i be the upper performance limit and L_i be the lower performance limit of ith partition.

If this condition is violated, then there is a requirement of load sharing. The identified node is then transferred to the neighboring gateway cluster and partitioning is done again. This process is repeated whenever the stability condition is violated.

In case the entire WMN is overloaded, then the further transfer of node to the neighboring gateway cluster is not possible. This situation is identified by checking the condition of non-bipratiteness of supergraph of WMN G^2.

If the supergraph of WMN G^2 is not bipartite between the overloaded and less loaded partitions, the WMN needs to share load with the wired network.

The FPLS load management scheme has a distinct advantage of being much faster than the Legacy system of partitioning. This is because of the fact that fixed partitioning is done only at the time of creating clusters and does not add to delay in packet transfer and hence this operation is offline and non-atomic. The transformation of whole WMN into clusters and their computation using matrix operation ensures that the integrity of the WMN is intact despite shifting of MRs from one cluster to another. This is governed by satisfying the condition of $BC^T = 0 \bmod 2$. This load management scheme also ensures that the network will not crash under extreme overloading/ traffic conditions as the MRs will be shifted to wired networks in that situation.

However, the delay due to online computation in FPLS system is significant as compared to Legacy system only for denser networks with more number of MRs per cluster GW. Also, the performance of this scheme is dependent on the shifting of number of MRs on wired network under overloading conditions.

2.2 Dynamic load-aware routing (DLAR)

In DLAR protocol,[3] the load of intermediate nodes is considered as the main route selection logic and the congestion of active routes is continuously monitored and the active path is redefined in case of overloading. In this algorithm, the number of packets buffered at the interface is considered as primary route selection criteria. The traffic load is balanced and distributed to the network hosts using the least loaded routes.

DLAR works on building routes "on-demand". In case the route is needed but the destination information is not known, the ROUTE REQUEST packet is sent by the source in order to discover a route. When a non-duplicate ROUTE REQUEST is received by nodes which is other than the destination, they build the route entry for the source and destination nodes as well as record the previous hop entry (which is needed later to send the ROUTE REPLY to the source). Then, nodes attach their load information and broadcast the ROUTE REQUEST packet. The destination node accepts all the duplicate ROUTE REQUEST packets from the previous nodes and based on the load information, it chooses the least loaded route and sends the ROUTE REPLY on the selected route to the source. The DLAR algorithm presents three route selection schemes.

In DLAR scheme 1, the algorithm adds the routing load of each intermediate node and the route with the least sum is

selected. In case of a tie, the route with the shortest hop distance is selected. If the load and hop distance is same, the route in which the packet was received earliest by the destination is chosen.

DLAR scheme 2 is similar to scheme 1. Instead of utilizing the sum of number of queued packets in intermediate nodes, scheme 2 utilizes the average number of packets being buffered along the path. The tie condition of scheme 2 is similar to that of scheme 1.

DLAR scheme 3 chooses the selected route as the one which has least number of congested nodes. A threshold load value τ is decided and the load above that threshold is considered as congested node.

The DLAR algorithm has the advantage of being a simple logic and delivering a quick decision. This algorithm uses the most up-to-date route information at the time of route discovery by not allowing intermediate nodes to reply from cache. The congestion status of active data sessions is monitored continuously and the congested routes are reconfigured dynamically. The use of least-loaded routes helps in load balancing and efficient use of network resources. However, it leads to increase in overall traffic due to frequent shutting of ROUTE REQUEST and ROUTE REPLY packets and slows down the entire network under heavy traffic conditions.

2.3 Congestion aware load balancing (CALB) through multiple gateways

The CALB scheme of load balancing[4] is divided into two phases namely Gateway Discovery Protocol and Load Migration Procedure. In the Gateway Discovery phase, all the nodes of the network discover their primary gateways. The gateways broadcast their presence by sending signals (beacons) on periodic basis. The node registers itself to the gateway on receiving the beacon signal in case it has not already registered or the new gateway is nearer than the already registered gateway. The GW_REQ message is sent by the node for allocation request of the GW and is accepted by the GW by sending back the GW_REP packet. Once allocated, each node sends the periodic HELLO packet to other nodes to notify its allocated GW. When the node changes its gateway, it saves the previous allocated gateway as secondary gateway which can be utilized for load balancing at a later time. Initial selection of gateways is based on hop-count as other node-specific information (like queue length, etc.) is not available. The second phase is Load Migration Procedure in which each GW monitors its queue length during a time window. If the average queue length is rises above a certain threshold value in that time, it is the indication of upcoming congestion at the GW. In this case, the GW identifies the high traffic nodes and sends the notification to these nodes by sending CONGEST_NOTIFY message intimating them to look for alternative GWs that are relatively less congested.

The CALB scheme has the advantage over the path load balancing schemes because load balancing across gateways is much more effective than the load balancing along the paths. However, this scheme creates additional traffic at the time of node shifting to the alternative gateway.

2.4 Load balanced ad hoc routing (LBAR) protocol

The LBAR protocol[5] primarily finds a path which reflects the least traffic so that data packets can be sent with least delay.

The route discovery procedure begins when a source node would like to communicate with the node which does not have an identified route. The process is twofold – forward and backward. The forward stage begins at the source node by broadcasting the setup messages (carrying the cost information from source to the current node) to its neighboring nodes. A node which receives the setup message updates its cost information in the same way and forwards to its neighbors. To prevent looping, the setup messages contain the route record and the list of all node IDs used in establishing the path from source node to the current node. The destination node collects all the arriving setup messages within a route-select waiting period and selects the best-cost route. The backward process is the sending of ACK message from destination towards the source along the selected path. Upon receiving the ACK message, the source begins the packet transmission. In case of link failure, the destination sends the ACK message to an alternate route and then the packet transmission resumes.

The Activity Ai is the number of active paths through node i (where the greater the value of Activity is, the more traffic passing through node i would be), the Traffic Interference $TI_i = \sum_{\forall j} A_j^i$ is the sum of activity of neighboring nodes of node i, and j is the neighboring node of i, then the Cost C_k of route k is defined by

$$C_k = \sum_{i \in k} (A_i + TI_i) = \sum_{i \in k} (A_i + \sum_{\forall j} A_j^i) \qquad (1)$$

where i is the node on path k other than the source and destination and j is the neighboring node of i.

LBAR has lesser end-to-end delay timing as compared to the traditional DSR[10] and AODV[11] protocols. In addition to that, LBAR offers quick response to link failures thereby enhancing the data transmission reliability. However, this higher reliability comes at the cost of higher number of ACK messages floating on the network and it slows down the network in case of frequent link failures.

2.5 Neighborhood load routing (NLR) protocol

In the Neighborhood Load Routing (NLR) protocol,[6] the load balancing is achieved by concentrating on the load of the center node. The prime reason for center node loading is due to the fact that the typical routing algorithm chooses shortest path for packing transmission thereby causing excessive loading of center nodes/ nodes near to GW. Hence, in order to balance the load, the loads over different nodes must be kept comparable or relatively equal. In this protocol, the center load balancing problem is presented as a weighted directed graph. A WMN is represented as $G = (V, E)$ where V denotes a set of nodes in the network and E is a set of links between each pair of nodes. The load balancing in this network $G = (V, E)$ is achieved, if and only if $LB_i \to \overline{LB}(i \in V)$ where LB_i denotes the traffic load of node i and $\overline{LB}$ denotes the average load of the network. This means that the traffic load of each node of the network should be approximated to the average load of the network for achieving the load balancing.

When the traffic load of a node is high, it diverts the packets to the neighborhood nodes. Hence, the average load of each neighborhood node is measured and the aim is to bypass not only the busy node but also the busy neighborhood nodes. Therefore, in case the neighborhood nodes are also busy/ heavily loaded, the packet transmission is further deferred

from these busy nodes. This means that in the heavily loaded neighborhood, the extra traffic in one node generates interference in all its neighboring nodes and this results in building up of a waiting queue. To resolve this issue, NLR is developed to check the average value of the neighborhood load of a link by the formula

$$NLR = \sum_{i=1}^{k} \frac{load_i^n}{b_i^n} \quad \text{and} \quad n = \frac{tr}{d_{avg}} \qquad (2)$$

where n is the interference radius of neighborhood in the hop number; tr denotes the transmission range ; d_{avg} is the average distance between the two one-hop nodes $load_i^n$; denotes the average load of a neighborhood of node i with radius n hops and b_i^n is the average transmission rate of this neighborhood.

The NLR has a significant performance advantage in grid topology WMNs in the areas of average packet delivery ratio and average throughput and it achieves lowest average end-to-end delay and average jitter. However, in random topology networks, the nodes are distributed unevenly. This causes heavy interference in high node density neighborhoods which lowers the overall communication quality in high node density neighborhoods and degrades the overall network performance.

2.6 Probing-based any path forwarding routing (PAFR) algorithms

In the probing-based any path forwarding routing (PAFR),[7] the authors studied the delay-optimal routing using a probing-based any path forwarding approach, and specific consideration had been given to transient link uncertainties. The simple two-state model is developed in which each link either fails or works at full rate, the connection between WMN routing and a stochastic recoverable version of the classic Canadian Traveler Problem (SRCTP) is established. The online routing algorithm for delay minimization is developed. In this, a node checks (online) the neighbor's availability upon each packet transmission, and accordingly decides its next hop while being transmitted. The Canadian Traveler Problem (CTP) is an online optimization problem which has incomplete path information. This is similar to the traveler in Canada who wants to travel in the country in winter. He has all the necessary information like maps and directions but some of the roads are blocked due to snowfall and severe weather conditions. However, the path blockage information is available to him only when he reaches near the particular path. The problem has to devise a strategy with minimum expected travel time. The SRCTP-based routing protocol includes three parts: a DSDV[18]-like message exchange mechanism, an optimal forwarder set calculation algorithm and an online probing and forwarding procedure. A DSDV, being a distance-vector protocol, maintains a routing table for each destination, which are updated regularly by updates generated by neighbors. The SRCTP-based protocol uses a revised version of DSDV routing message exchange mechanism. The node n_i has k neighbor nodes $\{n_i^1, \dots, n_i^k\}$, among which h neighbors are chosen as a candidate set for the next hop dynamically through a probing process. Through probing, packet will be quickly delivered by instantaneously available link rather than insisting on the temporal bad link.

This protocol is highly efficient in the networks where link reliability is a big issue or in the networks setup in high noise and interference areas. However, the delay performance of this protocol closely depends on the probing order of the neighbors. The size of candidate set h also plays an important

role in delay minimization. Increasing h offers a better chance of finding a working link though it increases the probing cost.

2.7 A Cross-Layer Enhanced and Adaptive Routing Framework (CLEAR)

The Cross-Layer Enhanced and Adaptive Routing Framework (CLEAR)[8] includes a bio-inspired routing protocol called Birds Migration Routing (BMR) protocol that adopts a cross-layer based routing metric called Multi-Level Routing (MLR) metric, a mobility tracking module embedded in BMR and a three dimensional load balancing module. These modules work together in order to perform efficient routing in WMNs. The MLR distributes the account of Quality of Service (QoS) parameters that are needed to assess the status of wireless medium among multiple levels or stages. This approach simplifies the routing metric and doesn't need one expression to combine all the parameters. The MLR is concerned with load balancing among Wireless Mesh Routers (WMRs) by accounting for the queue length and the usage of wireless medium. The second module in CLEAR is routing module (BMR). This is inspired from the biological behavior of bird's flocks and their migration process. This behavior is utilized to balance the load among the gateways within the network. In BMR, the network backbone is divided into flocks (regions) where the traffic of each flock migrates to the least congested gateway. The functionality of BMR is complemented by the usage of simple association protocol, defined in CLEAR, and it is part of the mobility tracking module, run by the Wireless Mesh Clients (WMCs) to define their existence with respect to the WMRs around them.

The CLEAR outperforms other routing schemes in terms of network throughput, end-to-end delays and interference reduction. In this framework, the boundaries between the physical, data link and routing layer are broken to achieve extra performance.

2.8 Load-balanced multicast tree routing (LMTR) algorithm

The Load-balanced Multicast Tree Routing (LMTR)[9] algorithm focuses on the problem of multicast load balancing in Multi-channel Multi-radio WMNs (MCMR-WMN). An innovative load-aware dynamic cost function is introduced to weigh the links of the network. This cost function considers the benefits of Wireless Broadcast Advantage (WBA) as well as the problem of load balancing. The LMTR algorithm provides balanced multicast trees using this cost function. With the latest usage of multicast applications like IP-TV, video conference, distant education, online games, etc., this algorithm efficiently balances the network in such environments. Consider a WMN represented by graph $G = (V, E)$ where $V = \{v_1, v_2, v_3, \dots, v_n\}$ is the set of vertices and E denotes the matrix of communication links between nodes and in which a node x is directly connected to node y, then

$$E = \{e_{xy}\}_{nxn}; \quad x, y = 1, 2, \dots, n \qquad (3)$$

Where e_{xy} indicates the assigned channel (e.g. channel k) of link (x, y) if x is directly connected to y, else it is zero. In other words, if K is the number of available non-overlapping channels in the network

$$e_{xy} = \begin{cases} k \in \{1, 2, \dots, K\} & \text{if } x \text{ is directly connected to } y \\ 0 & \text{else} \end{cases} \qquad (4)$$

If link (x, y) is bound to channel k, then link (y, x) also uses channel k and $e_{xy} = e_{yx}$.

LMTR finds the multicast routing tree for an arrived multicast session request by assigning a load-aware dynamic cost to the links. The cost of links will be updated based on the "WBA" and "Load Balancing" factors. In LMTR, the initial tree of each session consists only of the source node. The destinations are then added to the tree, step by step. At the end of each step, the "covered destination", which is the one that is located on the multicast tree, is defined.

For each new multicast session request, LMTR constructs the bandwidth-guaranteed multicast tree step-by-step. Initially, there isn't any covered link; therefore, the cost function $CF (x, y)$ will be determined by the traffic load of both mesh routers x and y. The initial tree consists only of the source node. The destinations and corresponding paths then will be efficiently added to the tree step-by-step. In each step, LMTR first calculates the link costs; it then finds the feasible minimum-cost paths connecting source node to uncovered destinations. Among these minimum- cost paths, the path with the least cost will be selected and added to the tree. Also, the corresponding forwarding nodes will be added to the set of forwarding nodes at multicast tree. In each step of the algorithm, by adding the optimum path to the tree, it is necessary to update the list of uncovered destinations and other parameters required to recalculate the link costs in next step. The above process will be repeated until the multicast routing tree spans all destination nodes.

The algorithm is very effective in the scenarios where multicast traffic dominates. In the mixed traffic scenario, it is recommended to use another suitable algorithm.

3. PERFORMANCE ANALYSIS
3.1 FPLS
The objective of FPLS[2] is to utilize network efficiently. They have evaluated the performance of their proposed algorithm on the basis of percentage utilization and average response time for varying conditions of MRs and GWs.

For their simulation scenario, the simulation results of average response time and percentage utilization for 5 GW with respect to increasing number of MRs up to 500 were determined. Similarly, the average response time and percentage utilization of 100 MRs were determined with respect to increasing number of GWs up to 10. The comparison shows that for fixed number of GWs, the average response time and percentage utilization of network increases almost linearly (5 times increase in response time and 4 times increase in percentage utilization with 5 times increase of MRs). However, the results are reversed in the scenario of fixed MRs and increasing number of GWs.

They concluded that increasing the number of MRs per GW provides better percentage utilization and average response time because of reduction in throughput of legacy system due to reduction in GWs.

3.2 DLAR
DLAR[3] focusses on improvement of delay caused due to congestion and queue at intermediate nodes. The simulation results were compared with DSR[10] which considers the shortest route. They evaluated the performance of the algorithm on the basis of packet delivery ratio, hop-count and end-to-end delay with varying pause time. The simulation results show that packet delivery ratio increases up to 30%

with DLAR schemes in comparison with DSR with 20 sources sending 8 packets/second. This increase steps up to 40% with 40 sources sending 4 packets/second. The hop-count for DLAR schemes goes down 20% and end-to-end delay reduces by 35% versus DSR.

They concluded that DLAR schemes outperform DSR which uses the shortest path and does not consider the routing load. DLAR protocols delivered more fractions of data packets and yielded shorter end-to-end delays.

3.3 CALB
In CALB[4] scheme, the evaluation was done on the basis of network instantaneous throughput, packet delivery ratio and average end-to-end delay for CALB load balancing scheme and compared it with default scheme which does not employ the load balancing. The simulation was done with 3 MRs which generated the traffic load *f1*, *f2* and *f3* at the rate of 1200, 300 and 700 kbps. It was noted that the throughput changed drastically after 50 seconds with the throughput of *f1* at 400 kbps, *f2* at 150 kbps and *f3* at around 10 kbps in case of default scheme. However, in CALB scheme, the throughput of *f1*, *f2* and *f3* balanced around 200 kbps. Also, the packet delivery ratio in case of CALB scheme almost doubled and average delay reduced by almost 50% in comparison to default scheme.

They concluded that their scheme provides elegant load balancing mechanism so that the traffic load is distributed among multiple gateways. This way all the available gateways are utilized for balancing the traffic load and mitigate congestion at some gateways.

3.4 LBAR
In the LBAR[5] scheme, three performance metrics were evaluated namely packet delivery ratio, average end-to-end delay and normalized routing load. These parameters were simulated for various numbers of sources and pause times and compared with AODV[11] and DSR.[10] The packet delivery fractions were similar for 10 and 0 sources for all the above three schemes but with 30 and 40 sources, the packet delivery fraction increased by 30% and 40% respectively in case of LBAR versus both AODV and DSR. Similarly, average delay time reduced to 4 times and normalized routing load increased 4 times for LBAR in case of 30 and 40 sources versus AODV and DSR.

The conclusion is the better performance of LBAR versus AODV and DSR is mainly because of redundant route information that is stored in destination node to provide aid in routing, which eliminates the necessity of source reinitiating of route discovery.

3.5 NLR
In NLR[6] protocol, they compared NLR with hop-count,[16-18] Expected Transmission Count (ETX),[14] Success Probability Product (SPP),[12] Load-count[4,13] and Weighted Cumulative Expected Transmission Time (WCETT-LB)[19] metric in average packet delivery ratio, average throughput, average end-to-end delay and average jitter in grid topology. The simulation results showed that the average throughput is 1.08 times the normalized value in NLR than Load-count, hop-count, ETX and SPP. Also, average packet delivery ratio is 1.35 times the normalized value in NLR than others listed above. Similarly, average end-to-end delay is about 1.1 times the normalized value.

This clearly concludes that NLR has distinctly better performance in grid topology WMNs compared to other

routing metrics. The important point to be considered is the fact that in grid topology WMNs, the average distance between each pair of two one-hop nodes is the same; the interference range of each node is set to be approximately equal as all mesh devices (802.11 enabled) have similar transmission powers. Also, NLR in grid topology is more accurate and precise in the selection of a communication path.

3.6 PAFR

PAFR[7] studied the simulation of stochastic recoverable version of the classic Canadian Traveler Problem (SRCTP) and Stopping Theory (ST) routing algorithms and compared with traditional deterministic routing such as ETX,[15] DSR[10] and AODV.[21] For the latter, the first node in the candidate set built by the algorithms, was selected. The performance was compared against SRCTP and ST in terms of end-to-end routing delay and packet delivery ratio.

Through extensive simulations, the effectiveness of SRCTP and ST under various network configurations was demonstrated. In particular, considerable reduction on end-to-end delay (51.15% − 73.02% for two-state links, 5.16% for multi-rate links) and improvement in packet delivery ratio (99.76% for two state links, 94.44% for multi-rate links) was observed, as compared to traditional deterministic WMN routing.

3.7 CLEAR

CLEAR[8] protocol adopts the performance metrics of average network throughput, average end-to-end delay, average path length and average routing overhead. The exhibited results were divided into two parts: performance analysis of Multi-level Routing (MLR) metric and performance analysis of complete CLEAR framework and comparison was done with traditional routing hop-count, ETX[15] and WCETT.[25] The results showed that MLR can deliver up to three times the data delivered by the hop-count metric, and up to two times the data delivered by both ETX and WCETT metrics especially when the network is heavily loaded, i.e. number of simultaneous connections is larger than 4. Also, MLR tends to use longer paths than other metrics as the traffic load increases. This is because MLR tries to avoid passing through the congested regions and looks for higher throughput paths even if they are longer. However, as the load increases, i.e. more than 6 simultaneous connections, MLR exhibits the highest end-to-end delay followed by WCETT and then ETX while the hop-count metric exhibits the least delay. A distinguishing feature of MLR over other metrics is the low amount of the generated routing overhead.

3.8 LMTR

The performance of LMTR[9] was compared with conventional Shortest Path Tree (SPT)[24] and Wireless Closest Terminal Branching (WCTB)[23] algorithms in grid and random topology networks. The traditional algorithms use hop-count as routing metric without considering the problem of load balancing. Different from existing algorithms, a novel load-aware cost function that considers both the benefits of Wireless Broadcast Advantage and Load Balancing are defined. The LMTR algorithm not only minimizes the number of transmissions, but also distributes the traffic among all mesh routers fairly. As a result, it implicitly reduces the amount of interference in the network. This scheme can also control the trade-off between load balancing and network delay. The simulation results show that LMTR algorithm outperforms the previous multicast routing algorithms such as SPT and WCTB in terms of load balancing on mesh routers and uses network resources efficiently.

4. CONCLUSION

The existing routing algorithms used for multi-hop, multi-gateway WMNs evaluate their performance on the basis of efficiency of delivering packets and time elapsed in this process. It is desirable that packet delivery ratio be maximum and average end-to-end delay be minimum. The different algorithms discussed in section 2 show improvements in packet delivery ratio and end-to-end delay for a given scenario by modifying existing algorithms. Section 3 presented comparative improvement achieved by these algorithms for the parameters such as packet delivery ratio, hop-count, end to end delay and average throughput in comparison to existing algorithms.

5. REFERENCES

[1] E. Alotaibi and B. Mukherjee, A Survey on Routing Algorithms for Wireless Ad-Hoc and Mesh Networks, Elsevier Computer Networks 56 (2012) 940–965, 15 November 2011.

[2] S. Pande, V. Pande, G. Kadambi and Y. Varshinin, Managing the Integrity of Wireless Mesh Networks for Load Sharing and Internetworking, IEEE/ACM Transactions on Networking, 10 May 2013.

[3] S. Lee and M. Gerla, Dynamic Load-Aware Routing in Ad Hoc Networks, Proc. ICC '01.

[4] D. Nandiraju, L. Santhanam, N. Nandiraju and D. P. Agrawal, Achieving Load Balancing in Wireless Mesh Networks Through Multiple Gateways, Proc. 2006 IEEE International Conference on Mobile Ad-Hoc and Sensor Systems (MASS), 2006.

[5] H. Hassanein and A. Zhou, Routing with Load Balancing in Wireless Ad Hoc Networks, Proc. ACM MSWiM '01.

[6] L. Zhao, A. Y. Al-Dubai and G. Min, An Efficient Neighbourhood Load Routing Metric for Wireless Mesh Networks, Simulation Modelling Practice and Theory 19 (2011) 1415–1426, 28 October 2010.

[7] F. Chen, Y. Wang, J. Liu and Z. Li, Probing-Based Anypath Forwarding Routing Algorithms in Wireless Mesh Networks, Ad Hoc Networks 13 (2014) 222–233, 23 April 2011.

[8] G. A. Al-Mashaqbeh, J. N. Al-Karaki and S. M. Bataineh, CLEAR: A Cross-Layer Enhanced and Adaptive Routing Framework for Wireless Mesh Networks, Published online: 2 July 2009 © Springer Science+Business Media, LLC. 2009

[9] A. Avokh and G. Mirjalily, Load-Balanced Multicast Tree Routing in Multi Channel Multi Radio Wireless Mesh Networks Using a New Cost Function, Published online: 20 March 2012 © Springer Science+Business Media, LLC. 2012

[10] D. B. Johnson and D. A. Maltz, Dynamic Source Routing in Ad Hoc Wireless Networks, In Mobile Computing, eds. T. Imielinski and H. Korth, (Chapter 5, Kluwer Academic Publishers, 1996), pp. 152- 181.

[11] C. W. Perkins, E. M. Royer and S. R. Das, Ad Hoc On-Demand Distance Vector Routing, IETF Draft, October 1999.

[12] S. Roy, D. Koutsonikolas, S. Das and Y. C. Hu, High-Throughput Multicast Routing Metrics in Wireless Mesh Networks, Proc. IEEE International Conference on Distributed Computing Systems, 2006.

[13] A. Pirzada, R. Wishart, M. Portmann and J. Indulska, ALARM: An Adaptive Load-Aware Routing Metric for Hybrid Wireless Mesh Networks, Proc. 32nd Australasian Computer Science Conference, 2009, pp. 25–34.

[14] S. Keshav, A Control-Theoretic Approach to Flow Control, Proc. Conference on Communications Architecture & Protocols, 1993, pp. 3–15.

[15] D. Couto, D. Aguayo, J. Bicket and R. Morris, A High-Throughput Path Metric for Multi-Hop Wireless Routing, Proc. 9th Annual International Conference on Mobile Computing and Networking, 2003, pp. 134–146.

[16] I. D. Chakeres and E. M. Belding-Royer, AODV Routing Protocol Implementation Design, Proc. International Workshop on Wireless Ad Hoc Networking (WWAN), Tokyo, Japan, March 2004.

[17] D. B. Johnson and D. A. Maltz, Dynamic Source Routing in Ad Hoc Wireless Networks, Mobile Computing 353 (1996) 153–181.

[18] C. Perkins and P. Bhagwat, Highly Dynamic Destination-Sequence Distance Vector, Routing (DSDV) for Mobile Computers, Proc. ACM SIGCOMM Computer Communication Review, October 1994, pp. 234–244.

[19] L. Ma and M. Denko, A Routing Metric for Load-Balancing in Wireless Mesh Networks, Proc. 21st International Conference on Advanced Information Networking and Applications Workshops, Vol. 2, 2007, pp. 409–414.

[20] Y. Bejerano, Efficient Integration of Multi-Hop Wireless and Wired Networks with QoS Constraints, ACM, Mobicom, 2002

[21] C. E. Perkins and E. M. Royer, Ad-Hoc On-Demand Distance Vector Routing, Proc. Second IEEE Workshop on Mobile Computing Systems and Applications, 1999, pp. 90–100. doi:10.1109/ MCSA.1999.749281.

[22] H. S. Chiu, K. L. Yeung and K. Lui, Maximizing Broadcast Load in Multi-Channel Multi-Interface Wireless Mesh Networks, Proc. GLOBECOM, 2008, pp. 533–537.

[23] T. Liu and W. Liao, Multicast Routing in Multi-Radio Multi-Channel Wireless Mesh Networks, IEEE Transactions on Wireless Communications 9(10) 2010, 3031–3039.

[24] U. T. Nguyen and J. Xu, Multicast Routing in Wireless Mesh Networks: Minimum Cost Trees or Shortest Path Trees?, IEEE Communication Magazine 45(11) 2007, 72–77.

[25] R. Draves, J. Padhye and B. Zill, Routing in Multi-Radio, Multi-Hop Wireless Mesh Networks, Proc. ACM Mobicom, 2004