

# Node Stability-Based Routing in Wireless Mesh Networks

Mustapha Boushaba, Abdelhakim Hafid  
Network Research Laboratory University of Montreal  
Montréal, Canada  
{boushamu, ahafid}@iro.umontreal.ca

Michel Gendreau  
CIRRELT and MAGI, École Polytechnique of Montreal,  
Montréal, Canada  
michel.gendreau@cirrelt.ca

**Abstract-** Network stability is a key performance metric in supporting real time communication over wireless networks. Because of high bandwidth demand and dynamic traffic variation, several paths in Wireless Mesh Networks (WMNs) are expected to be unstable. High levels of network instability can lead to interferences, packet losses and high delays.

In this paper, we address the stability problem of WMNs; instability in these networks is caused mainly by link quality fluctuations and frequent route flapping. First, we present the key factors that may cause network instability; then, we propose a new technique, called Node Stability-based Routing (NSR), using the entropy function (known as a measure of the uncertainty and the disorder in a system) to define a node stability and a probability function to select an appropriate gateway. Simulation results show that NSR can significantly improve the overall network performance compared to techniques using interference and channel switching (MIC), Expected Transmission count (ETX) or load at gateways as a routing metric, Reinforcement learning-based best path to best gateway (RLBDR), and nearest gateway (i.e., shortest path to gateway).

**Index Terms-** Multi-channel, Routing, Wireless Mesh Networks, Network Stability

## 1. INTRODUCTION

In the last several years, wireless communication has emerged as promising technology for next-generation networks. Among those networks, Wireless Mesh Networks (WMNs) [1] have become quite popular allowing communication to the Internet, anytime and anywhere. WMNs have the potential to eliminate cables and thus dramatically reduce installation and maintenance costs.

A WMN can be seen as a multi-hop Mobile Ad-hoc Network (MANET) with extended connectivity; the key difference is that WMNs are characterized by a relatively static architecture and low mobility. WMNs combine wired and wireless networks with wireless Mesh Routers (MRs) as backbone and mobile stations as users. The role of MRs is to relay information from one MR to another via multi-hop communications. Usually, MRs send traffic to a gateway (GW) that connects nodes to the Internet. In the case of a WMN with a single gateway, the gateway selection problem becomes simple; indeed, all upstream/downstream traffic flows traverse the same GW to the Internet; thus, the GW is more likely to become the bottleneck/single point of failure in the network [2]. To mitigate this problem, multiple gateways are installed to distribute load and improve performance. However, increasing the number of GWs does not necessarily increase the capacity of WMNs. Indeed, network capacity is closely related to the network connectivity and the placement of GWs; these issues are out of scope of this paper.

WMN performance is highly impacted by interferences and GWs congestion causing considerable packet losses and higher delays. To improve WMN performance, many schemes using directional and smart antennas [3], multiple input multiple output (MIMO) systems [4, 5], multiple radios and multiples channels [6, 7] and cooperative communications [xxxxxx] have been proposed. Particularly, routing schemes can play a major role impacting (positively or negatively) the network performance; indeed, the main goal of routing is to find better routes, according to specific routing metrics, from sources to destinations. To achieve a good network performance, a routing metric should (a) not impact negatively the network stability; (b) capture the characteristics of mesh networks; (c) compute paths in polynomial time, and (d) avoid forwarding loops [8]. Network stability is considered one of the most determinant network performance metrics which has not been considered by most routing protocols. Network instability occurs whenever the frequency of rerouting increases considerably; this increase is generally caused by one or more links that exhibit considerable quality fluctuations. In fact, an exaggerated link quality variation creates an inappropriate flooding of updates which may cause route flaps degrading the network performance.

Network instability causes degradation of the network performance impacting negatively applications that require a certain Quality of service (e.g., multimedia applications). Indeed, instability (frequent route flapping) may cause out of order delivery, high jitter, packet losses and/or high delays which are unacceptable for multimedia applications. For better understanding, let us consider the network configuration shown in Fig. 1 where nodes  $R1, R2, R3, R4, R5$  and  $R6$  are MRs, nodes  $R1$  and  $R2$  are source routers optionally equipped with traffic aggregation devices and nodes  $G1$  and  $G2$  are gateways. In this example, we consider delay as a routing metric; thus, traffic from  $R1$  to  $G1$  follows the path  $R1-R4-G1$ . Now, let us assume that the estimated value of the metrics of links  $(R6-G1)$  and  $(R3-R6)$  change to 0.5 ms; thus, the delay in sending packets from  $R3$  to  $G1$  through  $R6$  is now 1 ms; in this case, the routing protocol selects the path  $R1-R3-R6-G1$  to send traffic from  $R1$  to  $G1$ . Hence, some packets (of the same flow) rerouted on  $R1-R3-R6-G1$  will reach their destination  $d$  in the Internet before the packets already transmitted on  $R1-R4-G1$ ; consequently, many packets may be out-of-order at the destination  $d$  [27]. The problem becomes very critical when the route flapping is consistent. The destination will be overloaded with handling the out-of-order packets and therefore delay and jitter will increase making the quality of some multimedia applications, such as video-conferencing, unacceptable.

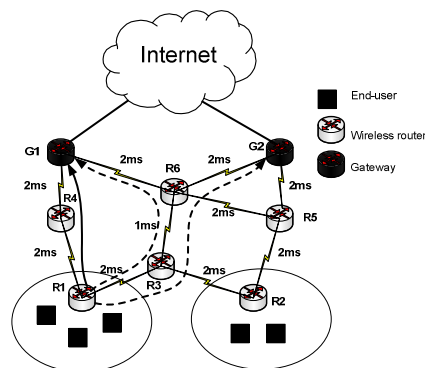


Fig. 1 Route flap scenario

Wireless Mesh Networks rely on routing protocols to find optimal paths. Generally, these protocols do not consider the instability of the network which may occur when the volume of traffic and demand for wireless capacity increase. Unlike wired links, wireless links often have frequent bandwidth fluctuations; factors, such as interference, fading and shadowing impact considerably the state of the link. Therefore, they increase the possibility of route breaks, during data transmission, which impact the network stability. In WMNs, generally multiple users intend to communicate to the Internet through gateways. Since the transmission coverage of wireless communication is limited, traffic need to be forwarded in multi-hop manner to gateways (generally, WMNs consist of many MRs and only one or a few gateways). This architecture (many-to-one and hop-by-hop network) makes the gateway a bottleneck and causes a kind of congestion, called funneling effect [41, 42], near the gateway and thus influences delay and reliability

This paper extends our contribution in [22] and proposes a new routing technique, in WMNs, that takes into account network stability with the assumption that all traffic go through gateways; indeed, the objective of the proposed scheme is not to impact negatively the network stability. The proposed paper extends our previous contribution by (1) reconsidering the stability index that quantifies the stability of Mesh routers; (2) improving/extending the routing protocol by adding a new Loop Free Forwarding Algorithm; (3) proposing a new technique based on probabilities to select gateways, which improves and reduces the number of routes flapping. Using NSR, upon receipt of a packet, a MR selects a most stable mesh router as the next hop, towards a selected gateway; and (4) computing the Gini index for different scenarios to study the fairness between gateways. Simulations show that our proposed Node Stability-based routing outperforms existing routing protocols.

The remainder of the paper is organized as follows. In Section 2, we present related work. Section 3 presents some factors leading to unstable networks. Section 4 presents the entropy function and its usefulness. In Section 5, we present the details of the proposed Node Stability-based Routing scheme. Section 6 evaluates the performance of our proposal. Section 7 concludes the paper and presents future work.

## **2. BACKGROUND**

### **2.1 Factors leading to unstable networks**

High levels of network instability can lead to packet losses and increase network latency. There are several factors that lead to network instability; these factors include: (1) Interferences: they are one of the major factors that impact network performance by creating instability. In wireless networks, channels are shared among nodes; thus, when nodes, sharing the same channel, transmit at the same time, they can cause inter-flow interferences and intra-flow interferences; (2) Routing protocols: particularly, in link-state protocols, the inappropriate number of advertisement messages to update routes may consume an excessive amount of resources in the network and cause route flaps. Consequently, it degrades the performance and creates an unstable network; (3) Network topology and traffic patterns: they have an impact on the behavior of routing and thus network

stability. In particular, the shortest path routing protocol, which has seen enormous success in wired networks, is inadequate in WMNs. In fact, the shortest path routing, using congestion as the link cost, is very prone to instability under heavy loads and bursty traffic; (4) Congestion: it is the result of MAC queue buildup at intermediate nodes; it can be observed by a forwarding node when packets arrival rate is bigger than its forwarding rate. There are many factors that contribute to congestion, namely network topology, number of flows, traffic characteristics of the flows and their routes, as well as channel capacity and the available transmission rate at the physical layer. The presence of congestion in the network impacts the network throughput, causes peer failures in routers and leads to routing failures and thus network instability; (5) route flapping: it occurs when a MR alternately advertises a destination network first via one route then via another (e.g., unavailable route and then available again). This phenomenon is undesirable in WMNs; it often forces MR to compute a new preferred route to a destination which causes network delay and requires extra CPU cycles. Whenever Route Flapping occurs, it causes unnecessary delay in packets forwarding; and (6) Effects of exploration: One of the major difficulties in the development of reinforcement learning algorithms is managing the trade-off between the execution of the best known policy, the exploitation and the exploration. In a multi-agent environment, exploration can contribute to instability since the agents switch their respective actions from time to time. The problem is even greater when the number of nodes that contribute to exploration is high.

## **2.2 Entropy utility**

Entropy was first introduced in [29] as a thermodynamic property; it measures the reversible change in thermal energy concerning the absolute temperature. It was also defined by Boltzmann [30] in statistical mechanics as a combination of microstates statistic entropy to characterize the order and the self-organization of a system. Later, Shannon and Weaver [31] defined information entropy; it is used to quantify the uncertainty of the information and to quantify the capacity of a transmission channel. Since, the concept of entropy has been generalized, and extended to many fields, such as for example data communications, medicine and the ecological environment.

## **3. RELATED WORK**

In the context of telecommunications networks, several contributions [32, 33, 34, 35, 36, 37, 38, 49, 50] extended/adapted the entropy definitions in different domains. Lu et al. [32] apply the notion of entropy to ad hoc wireless networks; they use it to quantify the internal organization state generated by different self-organization schemes. This organization exhibits two levels: (1) microscopic level (based on logic links between nodes) and (2) macroscopic level (formation of flexible structure). The authors make use of entropy because (a) wireless networks and thermal dynamic systems are similar in terms of organization; (b) it describes the degree of order in the network; and (c) it exhibits the behavior of self-organization protocols compared to changes of inherent parameters in the network, such as link and node reliability. The authors define the entropy

of a link as  $-p(u,v) \log_2(p(u,v))$ , where  $p(u, v)$  is the probability that a link exists between nodes  $u$  and  $v$ ; the entropy of the network is then defined as the sum of the entropies of its links.

Sneppen et al. [33] define two entropies, namely road entropy and target entropy. The road entropy quantifies the predictability or the order/disorder of traffic around a node whereas the target entropy measures the predictability of traffic to a specified node. To improve the network performance, the authors propose a fast and accurate technique to classify connection types [34]; they classify the type of an access link into Ethernet, wireless LAN and low-bandwidth wired connection by using the median and the entropy of packet-pair inter-arrival times (i.e., inter-arrival times between back-to-back packets that travel from source to destination). The rationale behind their use of entropy instead of variance is that entropy is much better in capturing the randomness of a random variable than variance.

In the context of wireless sensor networks and MANET, a number of contributions propose to use entropy to determine stability of paths. Chen et al. [35] propose the EQMGA algorithm to construct a new metric-entropy in order to select long-life paths; the goal is to reduce, via the use of entropy, the number of route reconstructions (i.e., rerouting) to provide QoS guarantees. An et al. [36] propose an Entropy-based Routing Protocol using Mobility (ERPM) in Mobile Ad-hoc Wireless Sensor Networks; the objective of ERPM is to select the most stable route between any source and destination based on mobility of sensor nodes to increase the route lifetime. Gui et al. [37] propose a novel Entropy-based Long-life Multipath Routing algorithm called ELMR while Lian et al. [38] propose a QoS-Aware routing protocol; both proposals were designed for MANETs. They use the same formulation to compute the entropy and therefore the stability of a node. In their calculation of entropy, the authors use the relative mobility between any pair of nodes  $(m, n)$  represented as a probability. This formulation cannot be used in WMNs since nodes are static..

For WMNs, many routing schemes have been proposed in the literature [9, 10, 54]. These schemes could be roughly divided into three categories: (1) proactive routing protocols (e.g., Optimized Link State Routing OLSR [11]); (2) reactive routing protocols (e.g., Ad hoc On-demand Distance Vector AODV [12]); and (3) hybrid protocols (e.g., Temporally-ordered routing algorithm TORA [13]). Based on the aforementioned routing techniques, several routing metrics have emerged. In general, a routing metric is used, by a routing protocol, to select a path having the highest throughput, the lowest delay and/or the lowest packet loss ratio. However, most routing protocols do not take into consideration network stability. We believe that network stability has a considerable impact on network performance; indeed, a stable network outperforms an unstable network.

Stability is hard to define; it has been used in different domains and networks to indicate a state or quality of the system under consideration. In the open literature, several contributions [15, 16, 17, 28, 39, 40, 51] have been made to study stability in different networks. Gaertner et al. [15], in the context of MANETS, define a stable link as a link where at least a percentage  $p$  of packets is received in a time period  $T_s$ . They choose  $T_s$  to be (a) small enough to reflect rapid changes in link states; and

(b) large enough to tolerate some effects of fading and to prevent unnecessary route discoveries; they choose  $p$  to be (a) small enough to meet the link quality requirements; and (b) large enough to tolerate fast signal fading.

Routing, service and resource discovery capabilities are needed to enable applications and services over large scale WMNs. In this context, Aydin et al. [16] propose a layered architecture to autonomously select a set of backbone nodes to create a quorum system. They propose an algorithm called ADB (Adaptive Dynamic Backbone) that consists of three components (a) neighbor discovery; (b) backbone selection; and (c) backbone connection. ADB explores stability information, maintained by each mesh node, that consists of two metrics: (1) *nodeStability*, is the estimated stability of the area surrounding a node. It consists of a set of probabilities (the cardinality of this set is equal to the number of the node's neighbors); an element of this set represents an approximation of the probability  $P_i$  that a link from the node (let us denote it by  $node_i$ ) to its neighbor will not break within the next predefined time window; and (2) *pathStability*: is the estimated path stability from the node (denoted by  $node_i$ ) to its current backbone. This metric measures the probability that a path will still exist within the next time window; it is computed as follows:  $pathStability_i = pathStability_{(parent\ of\ i)} * P_i$ . This idea is important in an environment with high mobility (not the case of WMNs). However, the metric used to capture the stability of a link is not sufficient since it considers only the probability that it will be still usable within the next specific time window without considering/worrying about quality of links (interference, loss, load etc ).

For Ad-hoc networks [39, 40], Associativity-Based Routing (ABR) [39] is considered as one of the earliest contributions whose fundamental objective is to find longer-lived routes. Periodically, each node broadcasts a hello message to signify its existence and to compute the lifetime of links even if there is no traffic to send. ABR tries to choose stable links over transient links; a link between two neighboring nodes is defined as stable if one node continuously receives signals from its neighboring node and the number of hello messages exceeds a threshold. Otherwise, the link is unstable. Thus, the shortest and stable route is selected as the optimal route by the destination. Signal Stability Adaptive Routing (SSA) [40] estimates link stability based on signal strength. Each mobile node measures the signal strength to its neighbors. The authors distinguish between strongly connected links and weakly connected links; a link is considered to be strongly connected if the signal strength exceeds a certain predefined threshold; this is typically the case when two nodes are close to each other. Thus, the authors define a stable link as the link which exhibits the strongest signal for the maximum amount of time. SSA tries to find a path using only stable links; however, if no stable path is found, the protocol tries to find a path by using all possible links.

Ramachandran et al. [17] reported, after experiments on UCSB Meshnet [18] and MIT Roofnet [19], that the main reason of network instability is the fluctuating quality of links caused by the effect of multipath fading, interference and weather conditions. Their analysis of routing stability are based on link quality information and use three route-level characteristics [17]: (1) *prevalence*: represents the probability or the number of times that a path has been met in the past; (2) *persistence*:

represents the probability that a route remains unchanged over a long time period; and (3) *route flapping*: represents the variation of the paths during a time period. Their findings can be summarized as follows: (1) By analyzing the persistence and prevalence, the authors found that routes in wireless mesh networks are inherently unstable; indeed, they are weakly dominated by a single route; and (2) a routing protocol that always flaps routes will likely achieve only minimal gains (i.e., marginal improvement in throughput) in a large number of instances.

In order to improve the routing stability, the authors recommend to use a route flap dampening, such as using a route which offers 10% throughput improvement over the old one or using an alternate route which is consistently stable for a specified amount of time.

mETX (modified ETX) and ENT (Effective Number of Transmissions) are two metrics proposed in [52] and derived from ETX [20]. The authors were particularly interested in improving the measurements' accuracy of packet loss rates by considering the standard deviation and average links quality; the main objective is to reflect physical layer changes on routing metrics. Like ETX, mETX actively measures the link quality by using probe packets through computing the bit error probability in probe packets. On the other hand, ENT takes into account only the number of successive retransmissions of probe packets per link considering their variance. Both mETX and ENT improve the packet loss rate compared to ETX and take into account the probe packets size, which makes the use of data rate trivial; yet they suffer from the same problems as ETX (e.g., interference)

#### **4. PROPOSAL DESCRIPTION**

One of the key symptoms of network instability is the disappearance and the reappearance of routes in the routing table in an intermittent manner. Thus, the number of these disappearances/reappearances of routes may characterize the intensity of the perturbation in the network.

##### **4.1 Network Model and assumptions**

We consider multi-hop infrastructure WMNs with two sets of nodes: MRs and GWs. MRs form a multi-hop wireless backbone to relay traffic between users and the Internet. To reach the Internet, traffic passes through GWs which are MRs with more functionality (e.g., more buffer size and wired and wireless interfaces). Each MR may be equipped with multiple wireless interfaces and multiple channels on each interface. We assume that the interfaces of a MR are assigned to different channels. We formally model the backbone WMN as a graph  $G=(V, E)$  where  $V$  is the set of nodes (MRs and GWs) and  $E$  is the set of links. Let  $l_{ij}$  denotes the link between two MRs  $v_i$  and  $v_j$  ( $v_i, v_j \in V$ ). In  $V$ , there are GWs that provide connectivity to the Internet. We define the load  $L_g[t]$ , of gateway  $g$ , at time  $t$  as the average interface queue length [5, 6]. It can be measured by monitoring the size of packets buffered in the GW interface over a time period. The current load  $L_g[t]$ , at time  $t$ , is based on the previous estimated load  $L_g[t-1]$  and the volume of traffic  $V[t]$  that the gateway  $g$  has processed during the last time period  $[t-1, t]$ :

$$L_g[t] = \alpha \frac{V[t]}{C_g} + (1-\alpha)L_g[t-1] \quad (1)$$

where  $\alpha \in [0, 1]$  and  $C_g$  is the maximum queue length of gateway  $g$ .

We assume that each node  $v_i$  maintains a routing table  $RT_i$  that stores routing information to each destination  $d$ ;  $|RT_i| = N$  is the size of  $RT_i$ . For MR  $v_i$  ( $i = 1, 2, \dots, |V|$  and  $v_i \in V$ ), we denote by  $RT_i^t$  its routing table at time  $t$ . Each entry in  $RT_i^t$  includes  $\{d, (f_1, f_2, \dots, f_n), k_{ij}, v_j\}$  where  $n > 0, j \neq i, k_{ij}$  is the channel used on link  $l_{ij} \in E$ ,  $v_j$  is the next hop towards the destination  $d$ , and  $f_1 \dots f_n$  represent the characteristics (e.g., Interference Ratio and Loss Ratio) of  $l_{ij}$  when using  $k_{ij}$ . For each incoming packet,  $v_i$  uses the information stored in its routing table  $RT_i^t$  to assign the outgoing link to be used to forward the packet towards its destination.

Let us define  $RT_i^{t+1}$  as the routing table at time  $t+1$  of node  $v_i$ .  $RT_i^{t+1}$  is a combination of three sub routing tables: (1)  $r_{nc}^t$ : contains links that experience no changes between  $t$  and  $t+1$ ; (2)  $r_c^{t+1}$ : contains links that experience some changes between  $t$  and  $t+1$  (this change may be an improvement or a degradation of the quality of the link); and (3)  $r_n^{t+1}$ : contains new links observed between  $t$  and time  $t+1$  that were not present at  $t$ . The routing tables  $r_{nc}^t$ ,  $r_c^{t+1}$  and  $r_n^{t+1}$  have the same fields and a link can belong to only one of them.

$$RT_i^{t+1} = r_{nc}^t + r_c^{t+1} + r_n^{t+1} \quad (2)$$

## 4.2 Link Quality Metric and Node stability

### 4.2.1 Link Quality Metric ( $LQM_l$ )

$LQM_l$  is considered as one of the link characteristics stored in the routing table. According to [21]  $LQM_l$  is defined as a weighted function of two parameters: IR (Interference Ratio) and CL (Congestion Level). IR estimates the interference level in the network through the Signal to Noise Ratio (SNR) and the Signal to Interference-plus-Noise Ratio (SINR). IR for a node  $u$  on link  $l$  (the receiver  $u$  and the sender  $v$  are the ends of link  $l$  operating on the same channel  $i$ ) is defined as follows:

$$IR_l(u) = \frac{SINR_l(u)}{SNR_l(u)} \quad (3)$$

where  $SINR_l(u)$  and  $SNR_l(u)$  are defined in Eq. (4) and Eq. (5):

$$SINR_l(u) = \frac{P_l}{Noise + \sum_{w \in N_{l-v}} \tau_w \cdot P_u(w)} \quad (4)$$

$$SNR_l(u) = \frac{P_l}{Noise} \quad (5)$$

where,  $P_l$  is the signal strength,  $N_l$  is the set of nodes from which node  $u$  can hear a packet that interferes with link  $l$  and  $\tau_w$  gives the amount of time that node  $w$  occupies the channel.

CL is closely related to link utilization ( $U(l, t)$ ) during time period  $t$ .  $U(l, t)$  is defined as follows:

$$U(l, t) = \frac{\sum_{i \in Succ(l,t)} Size_i}{BW_l} \quad (6)$$



where,  $Succ(l, t)$  denotes the set of packets that have successfully traversed the link  $l$  during period time  $t$  and  $Size_i$  denotes the size of packet  $i$  ( $i \in Succ(l, t)$ ).

Thus, by using the moving average, we define CL as follows:

$$CL(l, t) = \beta U(l, t) + (1 - \beta) CL(l, t - 1) \quad (7)$$

where  $\beta \in [0, 1]$

By grouping IR and CL, LQM<sub>l</sub> for link  $l$  operating on channel  $i$  is defined as follows:

$$LQM_l = \left(1 - \left(\frac{1}{2}\right)^n\right) \times IR_l + \left(\frac{1}{2}\right)^n \times CL_l \quad (8)$$

where,  $n$  is the number of  $l$ -hop neighboring nodes that share channel  $i$  (number of interfering nodes) with the node computing its  $LQM_l$ .

The rationale behind the definition of LQM<sub>l</sub> shown in Eq. (8) is our intention to capture interferences and congestion, in computing LQM, while giving dynamically more weight to interferences (they impact considerably the network performance). A bigger value of  $n$ , gives more importance to interferences in computing the link quality; For  $n = 1$ , interferences and congestion have the same weight (0.5 each) to compute the link quality. Obviously, the smaller is the LQM<sub>l</sub> of a given link, the better its quality.

#### 4.2.2 Node stability-based routing

Before defining the stability of a node let us discuss the stability index which we associate to a link in a routing table. The stability index of a link is computed using the link quality history including the current quality; a link is said to be acceptable (resp. unacceptable) when its  $LQM_l$  is smaller (resp. bigger) than a predefined threshold. It is clear that a link that changes frequently between acceptable and unacceptable will not be chosen to route packets; indeed, these changes will create oscillations in the network causing its instability.

We propose an algorithm, called Stability Index Algorithm (SIA), that computes the link stability index using link oscillations between acceptable and unacceptable, link persistence, and link quality improvement/deterioration. In the following, we present the pseudo-code of the algorithm used to compute the stability index ( $S_l^{t+1}$ ) of link  $l$  at time  $t+1$ . SIA is executed by each node (mesh router)  $i \in V$ .

The algorithm starts by adding acceptable or unacceptable links (link quality bigger or smaller than a predefined threshold) in the routing table. Even if a link is unacceptable, it will be added in the routing table with a stability index equal to 0 (an acceptable link is added with stability index equal to 1) which corresponds to line 6 (vs. line 4) in the algorithm. Thus, based on these values at the initialization, the routing protocol will avoid selecting newly created links ( $S_l^t = 1$ : links having no history in the past) or bad quality links ( $S_l^t = 0$ ). After each period where the link quality does not change (line 9 with a small  $\beta$  value)

we increment by 1 the stability index of the link. If the link quality has improved relative to the past value, its stability index will be incremented by 2 (line 13 of the algorithm). The values 1 and 2 in lines 10 and 13 respectively are based on a binary representation to distinguish between different link states. Thus, to follow the evolution of an acceptable link (stable-improved), we used the values  $1=2^0$  and  $2=2^1$ . When the link quality has deteriorated but still acceptable ( $<\epsilon$ ) we divide its accumulated stability index by 2 (line 16); thus, if the link degradation continues, its index stability will exponentially approaches to zero. In the case of a link degradation above the threshold ( $\epsilon$ ), the stability index will be set to 0 (line 18).

### Algorithm 1: Stability Index Algorithm (SIA)

#### Variables

$l$ : link  
 $(LQM_l)^t$ : Link quality of  $l$  measured at time  $t$   
 $(LQM_l)^{t+1}$ : Link quality of  $l$  measured at time  $t+1$   
 $S_l^t$ : Stability index of link  $l$  at time  $t$   
 $S_l^{t+1}$ : Stability index of link  $l$  at time  $t+1$   
 $\epsilon$ : Predefined threshold.  
 $\beta$ : is a constant belonging to  $[0,1]$

#### Algorithm

```

1  For (each  $l \in RT_i^{t+1}$ ) do
2      if ( $l \notin RT_i^t$ ) /* new link*/
3          if ( $(LQM_l)^{t+1} < \epsilon$ ) /*accepted link*/
4               $S_l^{t+1} \leftarrow 1$ 
5          else
6               $S_l^{t+1} \leftarrow 0$ 
7          end if
8      else /* the link already exists*/
9          if ( $|(LQM_l)^{t+1} - (LQM_l)^t| \leq \beta$ ) /* the link quality has not changed*/
10              $S_l^{t+1} \leftarrow S_l^t + 1$ 
11          else /* the link quality has changed*/
12              if ( $(LQM_l)^{t+1} < (LQM_l)^t$ ) /* the link quality has improved*/
13                   $S_l^{t+1} \leftarrow S_l^t + 2$ 
14              else /*the link quality worsened and ( $(LQM_l)^{t+1} > (LQM_l)^t$ ) */
15                  if ( $(LQM_l)^{t+1} < \epsilon$ )
16                       $S_l^{t+1} \leftarrow S_l^t / 2$ 
17                  else
18                       $S_l^{t+1} \leftarrow 0$ 
19                  endif
20              endif
21          endif
22      endif
23  end for

```

To compute the stability of a node  $v_i$  we use (1) the stability index of the links that  $v_i$  maintains in its routing table (see SIA for details); and (2) the entropy function: a measure of the uncertainty and the disorder in a system.

Using Equation (10), that defines the probability of the stability of a link relative to other links (in the routing table of node  $v_i$ ) having a non-zero stability index, we define the entropy  $H_i(\Delta_t)$  at node  $v_i$  during time interval  $\Delta_t$  in Equation (11).

$$p_l = \frac{S_l^{t+1}}{\sum_{l \in RT_i^{t+1}} (S_l^{t+1})} \quad (10)$$

$$H_i(\Delta_t) = \frac{-\sum p_l \log p_l}{\log N_i} \quad (11)$$

where  $N_i = |\text{RT}_i^{t+1}|$  (number of entries in the routing table of  $v_i$ ). Let us note that a high value of the entropy (varies between 0 and 1) means a more stable node.

The complexity to compute the proposed entropy value including algorithm 1  $O(4\Delta(m)+1)$  where  $\Delta(m)$  is the degree of node  $m$  (i.e., the number of neighbors, of node  $m$ , sharing the same channel); We can generalize the result to  $O(n)$  where  $n$  is the number of nodes in the network. At time  $t$ , every node computes its entropy (see Equation 11) and broadcasts it to its  $1$ -hop neighbors. Upon the reception of this information, a node stores it in its routing table and uses it for best next hop to forward data packet to its destination. Indeed, a node will select the best path, from a set of possible paths (in its routing table) to a given destination, that has the most stable node (i.e., the node with the biggest entropy) and forming a link that belongs a set of forwarding links for the selected gateway (see section 4.2.4) as next hop.

### 4.2.3 Gateway Selection

In WMNs, traffic usually passes through gateways (GWs) to the Internet. In this paper, we concentrate on selecting stable routes to a selected gateway. Thus, we need to select the most appropriate gateway to the Internet; then, we determine the most stable path from a source to this gateway. Load has been used as a metric for gateway selection. [45, 46, 47, 48]; however, using exclusively load to select a gateway may considerably degrade the network performance. Indeed, if each source selects a least loaded gateway, then most of the sources will send traffic towards this gateway making it more loaded than others. These changes will cause rerouting to least loaded gateways; this process will be repeated causing major oscillations and thus network instability.

In this paper, we propose a probability-based scheme to select Gateways. Let us assume that each GW broadcasts Gateways Advertisement messages GWADV to MRs. The format of GWADV message is as follows:

idM	idG	iface	Load
-----	-----	-------	------

Fig. 2 GWADV message

where  $idM$  is the  $id$  of the message,  $idG$  is the identifier of the gateway broadcasting the advertisement message GWADV, and  $iface$  is the interface whose load is measured. The load at each gateway is computed using Equation (1).

Once a source MR receives GWADV, it selects the gateway to the Internet; the proposed selection scheme makes use of probabilities. More specifically, we compute the probability to use a given gateway (among all gateways in the network). The probability  $P_{ig}^t$  that node  $i$  selects gateway  $g$  at time  $t$ , is defined as follows:

$$P_{ig}^t = \alpha P_{ig}^{t-1} + (1-\alpha) I(g = \underset{g' \in G_i}{\operatorname{argmin}} q_{g'}^t) \quad (12)$$

where  $q_{g'}^t$  is the load of gateway  $g'$  (see Equation 1),  $I()$  is a function whose value is either 1 if  $g'$  is the least loaded gateway (LLG), otherwise 0,  $G_i$  is the set of gateways known to node  $i$  and  $\alpha$  is a tunable parameter whose value is between 0 and 1.

In order to achieve convergence, the proposed probability function needs to be initialized; we define  $P_{ig}^0$  (probability at time 0) as follows:

$$P_{ig}^0 = \frac{1/hop_g}{\sum_{g' \in G_i} 1/hop_{g'}} \quad (13)$$

where  $hop_g$  is the number of hops to gateway  $g$  and  $G_i$  is the set of gateways known to node  $i$ .

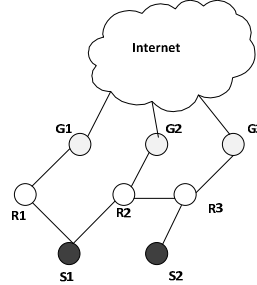


Fig. 3 Gateway selection illustration

For better understanding of the selection scheme, let us consider the example shown in Fig. 3. The sample WMN consists of (a) three gateways G1, G2 and G3; (2) three intermediate MRs R1, R2 and R3; and (c) two sources S1 and S2.

Let us assume that (a) periodically, each gateway sends advertisements (i.e., GWADV messages) to MRs; (b) the number of hops from S1 to G1, G2 and G3 is 2, 2, 3 respectively; and (c) for the number of hops from S2 to G1, G2 and G3 is 5, 3, 2 respectively. At the start-up, upon receipt of GWADV from gateways, S1 and S2 compute, using Equation 13, their probabilities to each Known gateway. The probabilities that S1 (resp. S2) selects G1, G2 and G3 are 0.37 (resp. 0.19), 0.37 (resp. 0.32) and 0.25 (resp. 0.48) respectively. If  $\alpha = 0.77$  and at  $t = 1$  G1 is the LLG (Least Loaded Gateway), then  $I(G1) = 1$ ,  $I(G2) = 0$  and  $I(G3) = 0$ . The probabilities for S1 to select G1, G2 and G3 are (using Equation 12)  $P_{S1G1}^1 = 0.77 P_{S1G1}^0 + 0.23 * 1$ ,  $P_{S1G2}^1 = 0.77 P_{S1G2}^0 + 0.23 * 0$ , and  $P_{S1G3}^1 = 0.77 P_{S1G3}^0 + 0.23 * 0$ . Tables 1 and 2 show the probabilities to select the gateways G1, G2, and G3 by S1 and S2 respectively; the gateway with the biggest probability is selected.

Table 1: Gateway Selection by S1

Period	LLG	$P_{S1G1}^t$	$P_{S1G2}^t$	$P_{S1G3}^t$	Selected Gateway
t = 0		0,37	0,37	0,25	G1
t = 1	G1	0,51	0,28	0,19	G1
t = 2	G2	0,40	0,45	0,15	G2
t = 3	G3	0,31	0,35	0,34	G2
t = 4	G3	0,24	0,27	0,49	G3
t = 5	G2	0,18	0,44	0,38	G2
t = 6	G1	0,37	0,34	0,29	G1
t = 7	G1	0,51	0,26	0,23	G1
t = 8	G2	0,40	0,43	0,17	G2
t = 9	G2	0,30	0,56	0,13	G2

Table 2: Gateway Selection by S2

Period	LLG	$P_{S2G1}^t$	$P_{S2G2}^t$	$P_{S2G3}^t$	Selected Gateway
t = 0		0,19	0,32	0,48	G3
t = 1	G1	0,38	0,25	0,37	G1
t = 2	G2	0,29	0,42	0,28	G2
t = 3	G3	0,22	0,32	0,45	G3
t = 4	G3	0,17	0,25	0,58	G3
t = 5	G2	0,13	0,42	0,44	G3
t = 6	G1	0,33	0,32	0,34	G3
t = 7	G1	0,49	0,25	0,26	G1
t = 8	G2	0,37	0,42	0,20	G2
t = 9	G2	0,29	0,56	0,16	G2

We observe that S1 and S2 do not select the same gateway all the time (e.g., t=0, t=3, t=5, t=6); thus, a source does not always select the least loaded gateway.

#### 4.2.4 Loop Free

Routing loops are serious problem for distributed routing protocols. They occur when packets continue to be routed indefinitely through the same set of routers. This problem may cause (1) An increase of network load and consequently an increase of packet losses and a decrease of available bandwidth; (2) the overload of routers processors, with unnecessary packet forwarding, slowing down traffic flows and the convergence of the routing protocol; and (3) the loss or the untimely processing of routing updates.

In order to prevent loops in NSR, we propose a Loop-Free Forwarding Algorithm (LFFA); the basic idea behind the proposed algorithm is to find a set of alternative links to forward packets towards a given gateway. This can be formulated as a graph theory problem. We model the Wireless Mesh Network as a network graph  $G(V, E)$  where  $V = \{1, \dots, n\}$  is the set of nodes (mesh routers) and  $E$  is the set of directed links. Let  $GW \subseteq V$  be the set of gateways in the WMN. We associate to each gateway  $g$  in  $GW$  a directed spanning tree  $T_g$  rooted at gateway  $g$  and having edges  $E_{T_g}$  directed towards  $g$  and all the vertices in  $V_g = \{V \setminus GW\} \cup g$ .  $T_g$  is obtained by selecting for each node in  $V_g$  (other than  $g$ ) its outgoing link that is traversed by the shortest path between this node and  $g$ ; This can be realized using Dijkstra algorithm. Having the graph  $G(V, E)$  and  $|GW|$  spanning trees for each gateway  $g$ , the problem consists of determining a set of directed links  $E_{F_g} \subset E$  such that the directed graph  $F_g = (V; E_{T_g} \cup E_{F_g})$  is acyclic. Algorithm 2 shows the pseudo-code of the proposed LFFA.

## Algorithm 2 Loop Free Forwarding Algorithm (LFFA)

### Input :

A graph  $G(V, E)$

GW a set of gateways

|GW| spanning trees  $T_g = (V, E_{T_g})$  each rooted at gateway  $g \in GW$

### Output :

$E_{F_g}, \forall g \in GW$  the set of forwarding links for gateway  $g$

$F_g = (GW, E_{T_g} \cup E_{F_g}), \forall g \in GW$ , a directed forwarding graph.

### Algorithm

```
1 Step 0:
2   Set  $GW^* = GW$ 
3   Set  $E_{F_g} = \{ \}, \forall g \in GW$ 
4 Step 1:
5   Select a gateway  $g \in GW^*$ 
6   Set  $E^* = \{E - E_{T_g}\}$ 
7   Let  $d_g$  be the depth of tree  $T_g$ 
8   Let  $L_i, i \in \{1, 2, \dots, d_g\}$  be the set of nodes which are at distance  $i$  from the gateway  $g$ 
9   Set  $k = d_g$ 
10 Step 2:
11   Select node  $u \in L_k$  such that the degree of node  $u, \delta(u) = \min_{u' \in L_k} \delta(u')$ 
12 Step 3:
13   Set  $E_{F_g} = E_{F_g} \cup e(u, w), \forall e(u, w) \in E^*$  directed link between  $u$  and  $w$ 
14 Step 4:
15   Remove all links  $e(w, u), \forall w \in V$  from  $E^*$ 
16   Remove node  $u$  from  $L_k$  and all links  $e(u, w)$  from  $E^*$ 
17   If  $L_k = \{ \}$  then  $k = k - 1$ 
18   If  $k \neq 0$  then go to step 2, otherwise go to 5.
19 Step 5:
20   Set  $F_g = (V, E_{T_g} \cup E_{F_g})$ 
21   Remove node  $g$  from  $GW^*$ .
22 If  $GW^* \neq \{ \}$  then go to step 1, otherwise STOP
```

The spanning tree can be constructed once during network deployment or whenever a MR joins or leaves the network (e.g., node failure). The complexity of algorithm 2 is  $O(|GW|n^2)$ , where |GW| is the cardinality of the set of gateways (i.e., number of gateways in the network) and  $n$  is the number of nodes. Let us note that in general only a few gateways are deployed in WMNs; thus, the complexity of algorithm 2 is  $O(n^2)$ . More importantly, algorithm 2 is used in some rare situations, namely network deployment, a MR joins/leaves the network, and channel reassignment if any. Thus, the impact on the complexity of algorithm 2 is limited if not negligible.

By using spanning trees, we can (1) create a topological sorting between a source node and gateways; and (2) ensure that the link on the shortest path between each node and the selected gateway will be included in the list of forwarding alternatives. Thus, in step 1, the algorithm chooses one gateway and creates a spanning tree rooted at the selected gateway. In step 2, the algorithm selects the farthest node with the minimum node degree in the spanning tree. In step 3, all outgoing links of the selected node are added to the list of forwarding alternatives. In step 4, all incoming links to the selected node are removed from the list of forwarding alternatives; this ensures that no packet forwarded from the selected node will return to it again in the future.

To better understand LFFA, let us consider the WMN shown in Fig. 4;

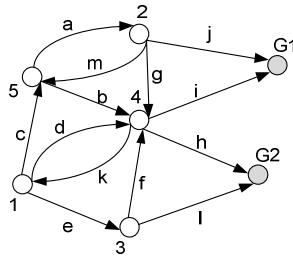


Fig. 4 Topology for loop free illustration

The network consists of (a) two gateways G1 and G2; and (b) Five mesh routers 1, 2, 3, 4 and 5. a, b, c, d, e, f, g, h, i, j, k, l are directed links (e.g., d is the link between MR 1 and MR 4 and k is the link between MR 4 and MR 1). Algorithm 2 takes as input the graph representing the WMN, the set of gateways  $GW = \{G1, G2\}$  and two spanning trees  $T_{G1}$  and  $T_{G2}$  rooted at G1 and G2 respectively. Fig. 5 shows the construction (Step 1) of the spanning tree rooted at G1; this spanning tree can be constructed once during network deployment or whenever a MR joins or leaves the network.

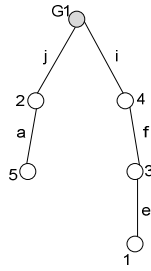


Fig. 5 Spanning tree rooted at G1

Table 3 summarizes all the operations performed by the algorithm for G1. (Similar operations for G2). Fig. 5 shows that the depth of the spanning tree is equal to 3; thus, 3 sets ( $L_k$  with  $k=3, 2$  and  $1$ ) are computed in the initialization phase; each set represents the MRs at a given level.

Table 3: Summary of operations performed by Algorithm 2 for Gateway G1

Initialization	Step 0	$GW^* = \{G1, G2\}, E_{FG1} = \{\}, E_{TG1} = \{a, e, f, i, j\}$
	Step 1	$E^* = \{b, c, d, g, k\}, d_{G1} = 3$ $L_1 = \{2, 4\}, L_2 = \{5, 3\}, L_3 = \{1\}$
K=3	Step 2	$u = 1$ (we select MR 1 from $L_3$ )
	Step 3	$E_{FG1} = \{\} \cup \{d, c\}$
	Step 4	$E^* = \{b, g\}, L_3 = \{\}$
K=2	Step 2	$u = 3$ (we select MR 3 from $L_2$ )
	Step 3	$E_{FG1} = \{d, c\}$ (the set is not changed)
	Step 4	$E^* = \{b, g, h\}, L_2 = \{5\}$
	Step 2	$u = 5$ (we select MR 5 from $L_2$ )
K=1	Step 3	$E_{FG1} = \{b, c, d\}$
	Step 4	$E^* = \{g\}, L_2 = \{\}$
	Step 2	$u = 2$ (we select MR 2 from $L_1$ )
	Step 3	$E_{FG1} = \{b, c, d\} \cup \{g\}$
	Step 4	$E^* = \{\}, L_1 = \{4\}$
	Step 2	$u = 4$ (we select MR 4 from $L_1$ )
Final Result	Step 3	$E_{FG1} = \{b, c, d, g\}$ (the set is not changed)
	Step 4	$E^* = \{\}, L_1 = \{\}$
	Step 5	$F_{G1} = E_{TG1} \cup E_{FG1} = \{a, e, f, i, j, b, c, d, g\}$ $GW^* = \{G2\}$

The final result for gateway G1 is the set  $E_{FG1}$  of alternative links towards it, Fig. 6 shows that  $E_{FG1} = \{a, e, f, i, j, b, c, d, g\}$ .

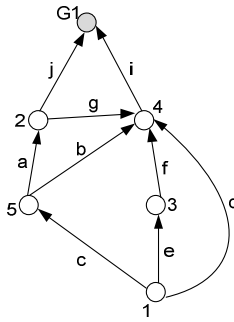


Fig. 6 Directed forwarding graph to G1

## 5. SIMULATION AND RESULTS

We study the performance of the proposed routing technique (NSR) using ns-2 with the module implementing multi-radio and multi-channel WMNs [24]; Table 4 shows the simulation parameters.

Table 4- NS2 simulation setup

Simulation Parameters	Values
Network Area	1000mx1000m
Link Data Rate	11Mbps
Frequency	9.14e+08 Hz
Traffic Type	CBR (UDP)
Network Load	1000-3000 Kbps
Simulation time	100s

The topology used for the backbone Wireless Mesh Network consists of 16 MRs randomly placed in a 1000x1000 area and 3 GWs (nodes 1, 2 and 3). The MAC layer protocol is IEEE 802.11 and the channel assignment is performed using the scheme in [25]. We use a realistic reception model by taking into consideration Bit Error Rate (BER) used in Intersil HFA3861B radio hardware [26] and the corresponding Frame Error Rate (FER). We use Constant Bit-Rate traffic (CBR) with UDP having a packet size of 1,000 bytes. Since most traffic is destined to the Internet, we assume that traffic is randomly generated from different source nodes and always transmitted, regardless of the network congestion condition, and may cross several intermediate nodes (i.e., multi-hop routing) to reach the Internet through a GW. All the simulations results are obtained after 100 runs with 95% confidence interval.

In the first set of simulations, we study the importance of using loop-free forwarding algorithm (LFFA). Thus, we compare in Fig. 7 the performance of NSR using the algorithm LFFA denoted by NSR-LFFA and NSR without using LFFA denoted by NSR-Without-LFFA. NSR-Without-LFFA selects the most stable node as next hop to forward traffic toward a selected gateway. We observe that when using the algorithm LFFA, the delay decreases which is very important for applications with end-to-end delay requirement. This is due to the fact that NSR-LFFA uses a spanning tree to avoid forwarding loops and to find a set of alternative links to forward packets towards a given gateway. Unlike NSR-Without-LFFA, each node focuses on sending traffic to the next most stable node which may create ping-pong effect between them until the loop is resolved; thus, choosing only the most stable node is not sufficient for efficient routing.



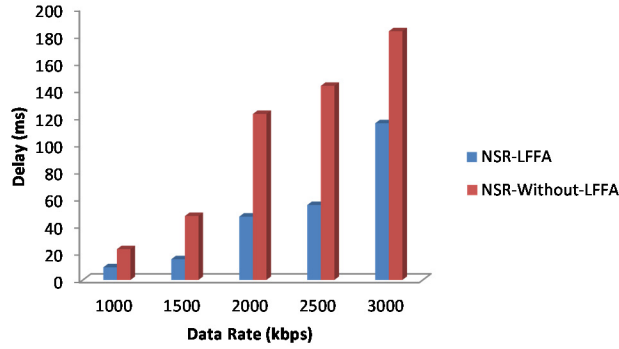


Fig. 7. Delay vs, Data Rate for NSR using and without using LFFA

In what follows, we have implemented and compared the performance of NSR-LFFA (denoted by NSR to simplify the notation) against RLBDR [21], MIC [23], ETX [20] and NEAREST-G (minimum number of hops to a gateway) based routing. The performance is evaluated in terms of (1) network average throughput: the amount of successfully transmitted bytes in a given time interval; (2) average end to end delay: the delay for sending CBR packets from source nodes to one of the gateways; (3) the loss rate: the average number of packets dropped inside the WMN over the number of sent packets; (4) the Gini index: measures the inequality of distributed traffic among gateways; and (5) the network stability: the mean of the stability index of MRs.

In the second set of simulations, we study the performance of NSR compared to other approaches when increasing the data rate. Fig. 8 shows that NSR outperforms RLBDR, MIC, ETX, NEAREST-G. Particularly, at high data rates (3000 kbps), NSR achieves 7%, 17%, 36% and 41% more throughput than RLBDR, MIC, ETX, NEAREST-G respectively.

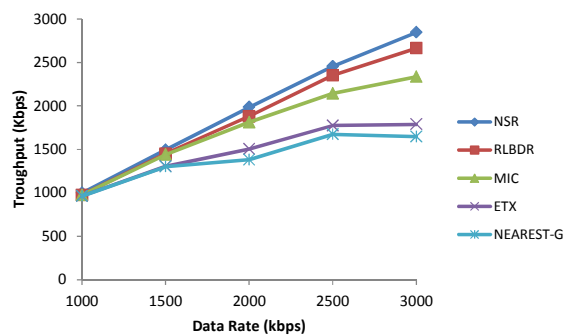


Fig. 8. Average throughput

In the third set of simulations, we study the performance in terms of delay. Fig. 9 shows clearly that NSR outperforms MIC, ETX, NEAREST-G. Particularly, at 1000 kbps, NSR performance is better than MIC, ETX and NEAREST-G by 31%, 29%, 16%, respectively. However, for data rates between 1000 kbps and 2000 kbps, RLBDR outperforms NSR by around 25%. This can be explained by the fact that NSR is not interested in minimizing the number of hops (from sources to gateways) but in

using the most stable nodes to route traffic. Thus, for RLBDR at low data rate, there is no problem around gateways and the traffic will follow its normal path (path with maximum reward). For data rates bigger than 2500 kbps, NSR outperforms RLBDR (e.g., at 2500kbps, NSR outperforms RLBDR by 49% and by 25% at 3000 kbps ); this is expected since at high data rates, interferences are high causing instability of nodes (in this case, a scheme that uses stable nodes to route traffic performs better). For RLBDR, at high data rates, congestion may occur around gateways; thus, longer paths (to get around congestion) will be used to reach gateways.

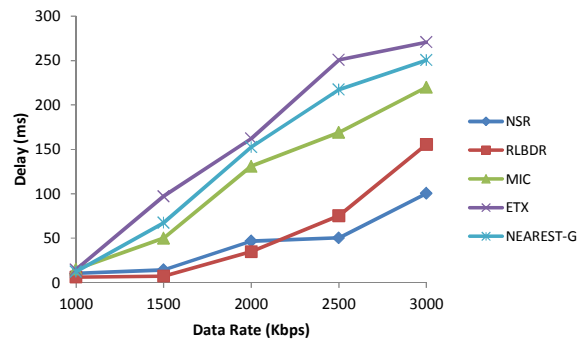


Fig. 9. Delay vs, Data Rate

In the fourth set of simulations, we study the impact of data rate variation on the packet loss ratio. Fig. 10 shows the total loss rate variation while Fig. 11 and Fig. 12 show the loss rate variation due to collisions and IFQ respectively. We observe that NSR outperforms RLBDR, MIC, ETX and NEAREST-G especially when the data rate increases. It is obvious that increasing data rates increases interferences (Fig. 11) and packet loss rate due to Interface Queue (IFQ) overflow (Fig. 12); this creates oscillations making the network unstable. This explains the performance of NSR since it uses paths with the most stable nodes.

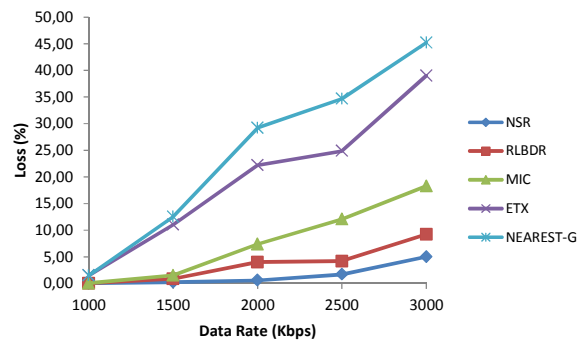


Fig. 10. Loss rate vs, Data Rate

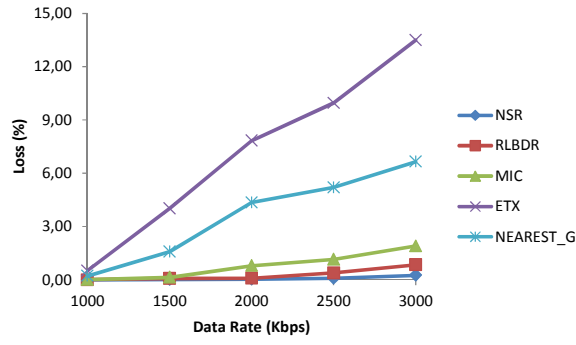


Fig. 11. Loss rate due to collisions vs. data rate

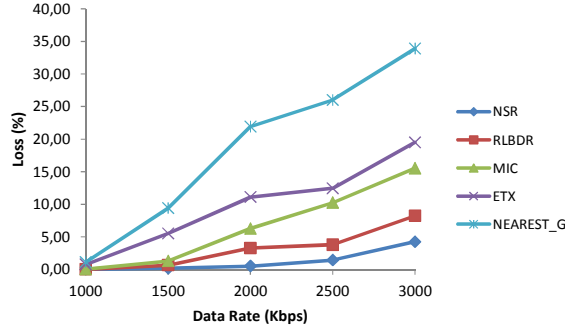


Fig. 12. Loss rate due to IFQ vs. data rate

Thus, we conclude that a routing protocol must not only consider routing metrics that capture the effect of interferences and packet losses but also metrics representing the stability of nodes. Indeed, by considering the stability of nodes, the routing protocol can avoid/reduce the well known problem of oscillations in WMNs and thus improve the network performance.

In the fifth set of simulations we study the behavior of our proposed gateway selection method in terms of traffic fairness between gateways. To perform this study, we make use of the Gini index. Originally the Gini index was proposed in the field of economics [43] to describe the distribution of wealth in human populations. It is a summary statistic of the Lorenz curve and a measure of inequality in a population. The Gini index was adopted in [44] to measure the instantaneous fairness of scheduling algorithms. In this paper, we use this index to measure the inequality in the distribution of traffic among gateways. Therefore, the Gini index will be the area between the Lorenz curve of the actual traffic served by gateways and the ideally Lorenz curve corresponding to a fair distribution of traffic among gateways. The Gini index  $G$  is defined as follows:

$$G = \left( \frac{2}{n^2 \bar{x}} \right) \sum_{i=1}^n \left( \left( i - \frac{n+1}{2} \right) x_i \right) \quad (14)$$

where:  $\bar{x} = \frac{1}{n} \sum_{i=1}^n x_i$ ,  $n$  is the number of gateways and  $x_i$  is the total amount of traffic in gateway  $i$ .  $G$  assumes values between 0 and 1; a low value of  $G$  indicates a more equal traffic distribution (fairness) among gateways.

In this set of simulations, we compare NSR with any load-based technique (Load curves) where the selected gateway is the least loaded gateway. We generated traffic with data rates of 1000, 2000 and 3000 kbps (see Fig 13 (a), Fig 13 (b) and Fig 13 (c) respectively). At 1000 kbps, Fig 13 (a) shows that during 79% of the simulation time, the value of the Gini index is around

0.2 for NSR whereas it is around 42% for the load curve. At 2000 kbps, Fig 13 (b) shows that during 74% of the simulation time the Gini index is around 0.2 for NSR whereas it is around 34% for the load curve. Finally, at 3000 kbps, Fig 13 (c) shows that 57% of the simulation time the Gini index for NSR is around 0.2 whereas it is 34% for the load curve. Thus, by choosing the least loaded gateway (load curves) the traffic is not fairly distributed among gateways compared to NSR. Since our proposed gateway selection scheme has a small Gini index during simulation time (around 0.20), we conclude that it achieves far better fairness than load-based selection schemes.

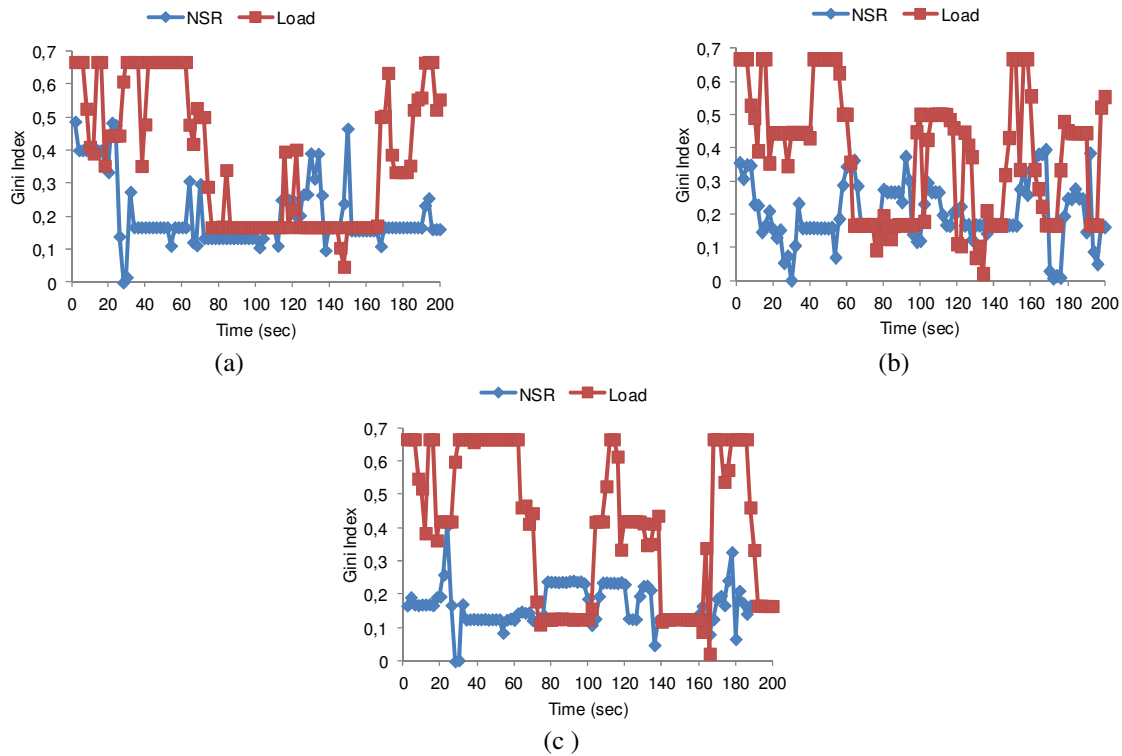


Fig.13. Gini Index vs. Time for a Data Rate for (a) 1000 kbps, (b) 2000 kbps and (c) 3000 kbps

Fig. 14 (a) and Fig 14 (b) show the percentage of traffic being served by different Gateways. Lorenz\_1000, Lorenz\_2000, and Lorenz\_3000 are Lorenz curves when generating traffic at 1000, 2000 and 3000 kbps respectively. Line of equality is the curve indicating a perfect load balancing among gateways. Fig 14 (a) shows that NSR forwards 27% of traffic to the Internet through one gateway and 58% of traffic through two gateways (i.e., 27% through the first gateway and 31% through the second gateway) compared to 33,33% and 66,66% respectively in the best case scenario (represented by Line of Equality curve). At 3000 kbps, by using NSR, 20% of traffic passes through one gateway and 49% through two gateways. Fig 14 (b) shows that for the load case, at 1000 kbps 20% of traffic passes through one gateway and 44% through two gateways whereas at 3000 kbps 13% of traffic passes through one gateway and 37% through two gateways. We conclude that fairness decreases with the traffic load (for all schemes) and NSR outperforms (in all cases) load-based selection techniques.

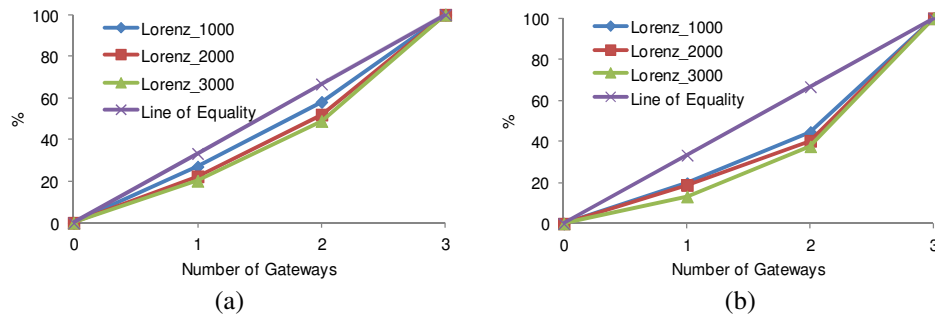


Fig.14. Percentage of traffic served by gateways (NSR case)

In the sixth set of simulation, we measure the stability of the network when using NSR. The network stability is computed as the average stability index of active nodes during a period of time. We classify the network stability into three classes (1) stable: when stability index is between 0.75 and 1; (2) marginally stable: when stability index is between 0.5 and 0.75; and (3) unstable: when stability index is below 0.5.

Fig. 15 shows that the network stability increases with the stability index. For example, with a data rate of 1000 kbps, the average stability index is around 0.9 (i.e., a stable network); this is expected since in a lightly loaded network, the quality of links is good (less interferences and data losses); thus, nodes are more stable (see Equation 11). When the data rate increases, more interferences and losses are introduced degrading the quality of links and thus the stability of nodes (e.g., at 2000 kbps, the average stability index is around 0.8 and at 3000 kbps the average stability index is around 0.7).

We observe that, using NSR, the network does not become unstable even when we increase the data rate; it is either stable or marginally stable. These findings show that by considering local stability (NSR selects most stable nodes to destination), one can achieve global stability (network stability)

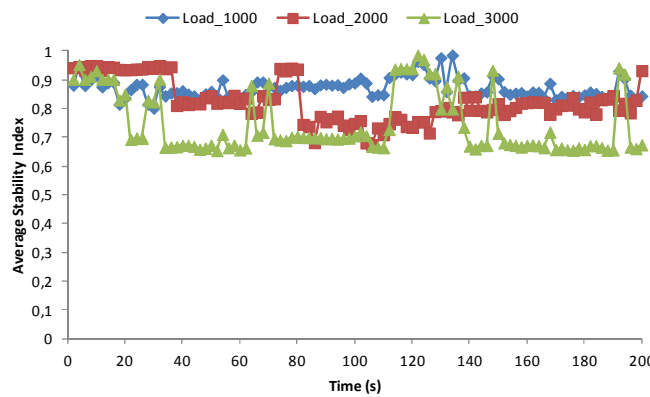


Fig.15. Average stability index

## 6. CONCLUSION

In this paper, we did study the impact of routing metrics on the performance of WMNs; our study findings led us to consider stability when routing traffic from sources to gateways. Thus, we did propose a novel routing technique, called NSR, that selects the most stable nodes in computing paths from sources to gateways. We also proposed a new gateway selection scheme that not only helps in avoiding frequent rerouting (and thus network instability) but also achieves fairness in distributing traffic

among the gateways in the network. Simulations results show that NSR outperforms all existing routing techniques (in terms of throughput, delay, loss rate, Gini index and stability index). In particular, it outperforms RLBD, MIC, ETX and NEAREST-G.

## ACKNOWLEDGMENT

We would like to thank Mr Vinicius da Cunha Martins Borges [52] for providing us with code for simulation. Also, we address our thanks to Dr. Abdelouab Belbekkouche for valuable discussions. The research reported in this manuscript has been supported in part by Natural Sciences and Engineering Research Council of Canada (NSERC) and Bell Canada.

## REFERENCES

- [1] Akyildiz, I. F., Wang, X., and Wang, W. (2005). Wireless mesh networks: a survey. *Computer Networks*, 47(4), 445-487.
- [2] Jun, J., and Sichitiu, M. (2003). The nominal capacity of wireless mesh networks. *IEEE Wireless Communications Magazine*. 10 (5), 8-14.
- [3] Stine, J.A. (2006). Exploiting smart antennas in wireless mesh networks using contention access. *IEEE Wireless Communications Magazine*. 13(2), 38-49.
- [4] Sundaresan, K. and Sivakumar, R. (2005). Routing in ad-hoc networks with MIMO links. *In Proceedings of 13th IEEE International Conference on Network Protocols*, pp. 11-.
- [5] Jaafar, W., Ajib, W., and Tabbane, S. (2007). The capacity of MIMO-based wireless mesh networks. *In Proceedings of 15th International Conference on Networks*, pp. 259 – 264.
- [6] Kyasanur, P., and Vaidya, N. H. (2005). Capacity of multi-channel wireless networks: impact of number of channels and interfaces. *In Proceedings of the 11th annual international conference on Mobile computing and networking*, pp. 43-57.
- [7] Kodialam, M. and Nandagopal, T. (2005). Characterizing the capacity region in multi-radio multi-channel wireless mesh networks. *In Proceedings of the 11th annual international conference on Mobile computing and networking*, pp. 73-87.
- [8] Yang, Y., Wang, J. and Kravets, R. (2005). Designing routing metrics for mesh networks. *In Proceedings of IEEE Workshop on Wireless Mesh Networks*.
- [9] Campista, M. E. M., Esposito, P. M., Moraes, I. M., Costa, L. H. M. K., Duarte, O. C. M. B., Passos, D. G., Albuquerque, C. V. N. D., Saade, D. C. M., and Rubinstein, M. G. (2008). Routing Metrics and Protocols for Wireless Mesh Networks. *IEEE Network: The Magazine of Global Internetworking*, 22 (1), 6-12.
- [10] Boukerche, A. (2004). Performance evaluation of routing protocols for ad hoc wireless networks. *Journal Mobile Networks and Applications*, 9(4), 333-342.
- [11] Clausen, T., and Jacquet, P. (2003). Optimized link state routing protocol (OLSR). *IETF internet draft RFC 3626*

- [12] Perkins, C.E., and Royer, E.B. (1999). Ad-hoc on-demand distance vector routing. *In Proceedings of Second IEEE Workshop on Mobile Computing Systems and Applications*, pp. 90-100.
- [13] Park, V.D., and Corson, M. S. (1997). A highly adaptive distributed routing algorithm for mobile wireless networks. *In Proceedings of Sixteenth Annual Joint Conference of the IEEE Computer and Communications Societies*, pp. 1405-1413.
- [14] Koksals, C. E., and Balakrishnan, H. (2006). Quality-aware routing metrics for time-varying wireless mesh networks. *IEEE Journal on Selected Areas in Communications*, 24 (11), 1984–1994.
- [15] Gaertner, G., Onuallain, E., Butterly, A., Singh, K., and Cahill, V. (2004). 802.11 Link Quality and its Prediction -An Experimental Study. *Lecture Notes in Computer Science*, Vol 3260, pp. 147-163.
- [16] Aydin, I., Jaikao, C., and Shen, C.C. (2005). Quorum-based match-making services for wireless mesh networks. *In Proceedings of the International Conference on Wireless And Mobile Computing, Networking And Communications*, pp. 377-382.
- [17] Ramachandran, K., Sheriff, I., Belding, E., and Almeroth, K. (2007). Routing Stability in Static Wireless Mesh Networks. *In Proceedings of the 8th Internatinoal Conference on Passive and Active Network Measurement*, pp. 73-82.
- [18] MeshNet, UCSB, <http://moment.cs.ucsb.edu/meshnet/>
- [19] Roofnet, MIT, <http://pdos.csail.mit.edu/roofnet/doku.php>
- [20] De Couto, D. S. J., Aguayo, D., Bicket, J., and Morris, R. (2003) A high-throughput path metric for multi-hop wireless routing. *In Proceedings of the 9th annual international conference on Mobile computing and networking*, pp. 134-146.
- [21] Boushaba, M., Hafid, A., Belbekkouche, A., and Gendreau, M. (2013). Reinforcement learning based routing in wireless mesh networks. *Wireless networks*, 19(8), 2079-2091.
- [22] Boushaba, M., Hafid, A., and Gendreau, M. (2013). Local Node Stability-Based Routing for Wireless Mesh Networks. *In Proceedings of IEEE Wireless Communications and Networking Conference*, pp. 1950—1955.
- [23] Yang, Y., Wang, J., and Kravets, R. (2005). Interference-aware load balancing for multihop wireless networks. *Technical Report UIUCDCS-R-2005-2526*, Department of Computer Science, University of Illinois at Urbana-Champaign.
- [24] Calvo, R.A., Campo, J.P. (2007). Adding Multiple Interface Support in NS-2, <http://personales.unican.es/aguerocr/files/ucMultiIfacesSupport.pdf>.
- [25] Beljadid, A., Hafid, A., and Gendreau, M. (2007). Optimal design of broadband wireless mesh networks. *In Proceedings of Global Telecommunications Conference*, pp. 4840-4845
- [26] Intersil HFA3861B Chip Specification, <http://www.chipdocs.com/pndecoder/datasheets/INTRS/HFA3861B.html>.
- [27] Ye, B., Jayasumana, A. P., and Piratla, N. (2006). On Monitoring of End-to-End Packet Reordering over the Internet. *In Proceedings of the International conference on Networking and Services*, pp. 3-3.

- [28] Moussaoui, A., Semchedine, F., & Boukerram, A. (2014). A link-state QoS routing protocol based on link stability for Mobile Ad hoc Networks. *Journal of Network and Computer Applications*, 39, 117-125.
- [29] Clausius, R. (1850). *Annalen der Physik und Chemie*, vol. 79, pp. 368-397 and 500-524.
- [30] Boltzmann, L. (1877). Über die Beziehung zwischen dem zweiten Hauptsatze der mechanischen Wärmetheorie und der Wahrscheinlichkeitsrechnung respective den Sätzen über das Wärmegleichgewicht. *Wiener Berichte*, 2 (76), 373–435.
- [31] Shannon, C. E., and Weave, W. (1949). *The Mathematical Theory of Communication*, Urbana, IL: University of Illinois Press.
- [32] Lu, J., Valois, F., Dohler, M., and Barthel, D. (2008). Quantifying Organization by Means of Entropy. *IEEE Communication Letters*, 12 (3), 185-187.
- [33] Sneppen, K., Trusina, A., and Rosvall, M. (2007). Hide-and-seek on complex networks. *Europhysics Letters (EPL)*. 69(5), 853-859.
- [34] Wei, W., Wang, B., Zhang, C., Kurose, J., and Towsley, D. (2008). Classification of access network types: Ethernet, wireless LAN, ADSL, cable modem or dialup?, *Computer Networks Journal*, 52 (17), 3205-3217.
- [35] Chen, H., Sun, B., and Zeng, Y. (2006). QoS multicast routing algorithm in MANET: an entropy-based GA. *Computational Intelligence Lecture Notes in Computer Science*. pp. 1279-1289.
- [36] An, B., Lee, J.-S., and Kim, N.-S. (2009). An Entropy-based Routing Protocol using Mobility in Mobile Ad-hoc Wireless Sensor Networks. *In Proceedings of IEEE 11th International Conference on Advanced Communication Technology*, pp. 949-952.
- [37] Gui, C., Sun, B., and Xia, C. (2008). An Entropy-Based Long-Life Multipath Routing Algorithm in MANET. *In Proceedings of Fourth International Conference on Natural Computation*, pp. 174-178.
- [38] Lian, J., Li, L., Zhu, X., and Sun, B. (2008). A QoS-aware routing protocol based on entropy for mobile ad hoc network. *In Proceedings of 4th International Conference on Wireless Communications, Networking and Mobile Computing*, pp. 1-4.
- [39] Toh, C.-K. (1997). Associativity-based routing for ad hoc mobile networks. *In Wireless Personal Communications*. 4(2), 103-139.
- [40] Dube, R., Rais, C. D., Wang, K.-Y., and Tripathi, S. K. (1997). Signal stability-based adaptive routing (SSA) for ad hoc mobile networks. *In IEEE Personal Communications*, 4(1), 36-45.
- [41] Rao, R. N., and Kesidis, G. (2006). On the Relation Between Capacity and Number of Sinks in an Sensor Network. *In Proceedings of Global Telecommunications Conference*, pp. 1-5.



- [42] Neishaboori, A., Kesidis, G. (2008). SINR-sensitive routing in wireless 802.11 mesh networks. *In Proceedings of the 5th IEEE International Conference on Mobile Ad Hoc and Sensor Systems*, pp. 623-628.
- [43] Cowell, F. A. (1977). Measuring inequality: *Techniques for the social sciences* (Wiley, New York)
- [44] Shi, H., Sethu, H., and Kanhere, S. (2005). An evaluation of fair packet schedulers using a novel measure of instantaneous fairness. *Computer communications*, 28(17), 1925-1937.
- [45] Huang, C-F., Lee, H-W., and Tseng, Y-C. (2004). A two-tier heterogeneous mobile Ad Hoc network architecture and its load-balance routing problem. *Journal of Mobile Networks and Applications*, 9(4), 379-391.
- [46] Kumar, R., Misra, M., and Sarje, A. K. (2007). An efficient gateway discovery in ad hoc networks for internet connectivity. *In Proceedings of International Conference on Computational Intelligence and Multimedia Applications*, pp. 275-282.
- [47] Shin, J., Lee, H., Na, J., Park, A., and Kim, S. (2005). Load balancing among internet gateways in ad hoc networks. *In Proceeding of 62nd Vehicular Technology Conference*, pp. 1677-1680.
- [48] Kim, Y., Jeong, Y., Seo, M., and Ma, J. (2007). Load balanced Mesh Portal Selection in Wireless Mesh Network. *In Proceedings of IEEE Military Communications Conference*, pp. 1-6.
- [49] Tellenbach, B., Burkhart, M., Schatzmann, D., Gugelmann, D., and Sornette, D. (2011). Accurate network anomaly classification with generalized entropy metrics. *Computer Networks*, 55(15), 3485-3502.
- [50] Karayiannis, N. B., and Nagabhushan Kaliyur, S. M. (2006). An entropy-constrained algorithm for routing of communication networks. *Computer communications*, 29(16), 3182-3196.
- [51] Moussaoui, A., and Boukeream, A. (2015). A survey of routing protocols based on link-stability in mobile ad hoc networks. *Journal of Network and Computer Applications*, 47, 1-10.
- [52] Borges, V.C.M., Pereira, D., Curado, M., and Monteiro, E. (2009). Routing Metric for Interference and Channel Diversity in Multi-Radio Wireless Mesh Networks. *In Proceedings of the 8th International Conference on Ad-Hoc, Mobile and Wireless Networks*, pp. 55-68.
- [53] De Couto, D. S., Aguayo, D., Chambers, B. A., and Morris, R. (2003). Performance of multihop wireless networks: Shortest path is not enough. *ACM SIGCOMM Computer Communication Review*, 33(1), 83-88.
- [54] Li, D., Liu, H., and Vasilakos, A. (2006). Comparative Analysis of Protocols in Wireless Ad Networks. *Ambient intelligence, wireless networking, and ubiquitous computing*, 167.
- [55]XXXXXXXXXX