An interference-aware routing metric for Wireless Mesh Networks

Riduan M. Abid* and Biaz Saâd

Department of Computer Science and Software Engineering,
Auburn University, AL 36849, USA
E-mail: abidmoh@auburn.edu
E-mail: biazsaad@auburn.edu
*Corresponding author

Abstract: In Wireless Mesh Networks (WMNs), most routing metrics measure losses using the ‘direct’ broadcast approach that incurs traffic overhead. On the other hand, the ‘indirect’ approach requires the measurement of contention and interference levels. Unfortunately, the two known interference models (protocol and physical) are inapplicable on real-world implementations. This work presents an adaptation of the physical interference model that relies on a probabilistic frames arrival model and can be implemented. This adaptation allows the design of a new interference-aware routing metric. This latter is described and compared to ETX and IEEE 802.11s Airtime using a real-world indoor pre-IEEE 802.11s Wireless Mesh test-bed.

Keywords: WMNs; wireless mesh networks; mobile communications; IEEE 802.11s; hybrid wireless mesh protocol; airtime; routing; loss rate; physical interference model.


Biographical notes: Riduan M. Abid received a PhD in Computer Science in 2010 from Auburn University (USA). He received in 2006 the Excellence Fulbright Scholarship. He is currently an Assistant Professor of Computer Science at Al Akhawayn University (Morocco).

Biaz Saâd received a PhD in Computer Science in 1999 from Texas A&M University and a PhD in Electrical Engineering in 1989 from the University Henri Poincaré in Nancy (France). He is presently an Associate Professor of Computer Science at Auburn University. He has held faculty positions at the École Supérieure de Technologie de Fès and Al Akhawayn University (Morocco). His current research is in wireless networking and mobile computing. His research is funded by the US National Science Foundation. He received in 1995 the Excellence Fulbright Scholarship. He has served as chair and technical chair of several conferences.

1 Introduction

Performance on wireless mesh networks (Akyildiz et al., 2005; Amir et al., 2010) gravely suffers for losses due to wireless link deterioration and contention stemming from...
the broadcast nature of the medium. To cope with the shared nature of the medium, research community proposed different promising techniques (e.g., multiple channels (Liu and Knightly, 2003; Raniwala and Chiueh 2005; So and Vaidya, 2004), multiple radios (Kung et al., 2001; Draves et al., 2004), directional antennas (Ramanathan et al., 2006; Santivanez et al., 2005), network coding (Markopoulou et al., 2007; Shen et al., 2007; Gollakota et al., 2007). But, many issues remain for the medium access control and routing. Routing on mesh networks cannot rely on a simple hop count metric: a routing metric should assess the quality of a link. A determinant parameter of channel quality is the packet loss rate. Many candidate routing metrics require the measurement of the packet loss rate. So, accurately measuring loss rate is paramount to make optimal routing decisions.

To estimate wireless losses, various routing metrics use one of two basic approaches:

- the broadcast approach
- the passive approach.

In the broadcast approach, probe frames are periodically broadcast. The number of successfully received probes is used to infer the loss rate. This approach suffers fundamental shortcomings: First, broadcast frames incur a traffic overhead that directly impacts the overall network performance. Second, the use of the large averaging periods to minimise the traffic overhead produces often stale measurements. Measurements based on large averaging periods do not follow quickly enough the time-varying nature of the channel quality. Third, broadcast frames have a fixed frame size and are sent using the default low rate used for broadcast. The low data rate produces an underestimated packet loss rate. The measurements do not reflect the real traffic that consists of different frame sizes and data rates. In conclusion, the broadcast approach produces underestimated loss rate that do not follow the dynamically variability of the channel quality. The passive approach uses real traffic frames instead of probes. Thus, it overcomes all the broadcast approach shortcomings: overhead, lack or responsiveness, and inaccuracy. However, the passive approach suffers the other major shortcomings for dealing with situations where there is no data traffic.

This paper proposes an alternative approach that overcomes the shortcomings of both the broadcast and passive approaches. The proposed approach indirectly estimates the loss rate by observing the causes of losses instead of the direct measurement of the effect. The causes of wireless losses are basically contention and interference. Contention contributes to frame losses by:

- Inducing excessive buffering of frames while waiting for free medium. This results in frame dropping due to buffers overflow.
- Frames collisions which worsens in case of the absence of the RTS/CTS mechanism (the default setting on IEEE 802.11 network interface cards).

On the other hand, interference contributes to frame losses by having a desired signal disrupted by other interfering signals. This results in a corrupted signal which is interpreted as a loss because of (FCS) Frame Sequence Check alteration.

While contention measurement is quite an easy task, an accurate interference measurement remains challenging. There are two basic interference models (Gupta and Kumar, 2000):
An interference-aware routing metric for Wireless Mesh Networks

- the protocol interference model
- the physical interference model.

Both models present major practical challenges. The protocol interference model is based on measuring the transmission and interference ranges. The challenge with this model is related to the complexity of measuring such ranges in real-world environments, especially in ad-hoc networks that have no predefined topology. The situation worsens indoor because of the mobile and unpredictable nature of obstacles, e.g., humans, furniture, walls. In the physical interference model, which is a SINR-based (Signal-to-Interference-Noise-Ratio) model, the challenge is related to the model requiring access to the SINR values of all simultaneously transmitting stations. This feature is unavailable on commodity network interface cards that report the signal strength only for successfully received frames. To overcome this latter challenge, we propose a probabilistic interference measurement framework that uses frame arrival models as a means to infer probabilities of interference.

We adapt the physical interference model to become practical, and propose a new interference-aware: the Interference and Contention Estimator (ICE). ICE overcomes the shortcomings of both the broadcast and passive approaches. ICE does not use extra broadcast frames and overcomes the major shortcoming of probing idle links (which is inherent to the passive approach). The absence of data traffic becomes a property to detect by itself. The absence of traffic may be interpreted as low contention and low interference (in absence of hidden terminals). ICE also accounts for the asymmetric nature of wireless links by exchanging channel quality measurements, of both link directions, between the link endpoints. These measurements are piggybacked in the default IEEE 802.11s proactive broadcast routing frames without the need for any extra broadcast frames.

To evaluate the performance of ICE, we implemented a real-world indoor IEEE 802.11s WMN test bed. The implementation is open-source and available online (Abid et al., 2010). ICE is compared to ETX (De Couto et al., 2003) and to the IEEE 802.11s default Airtime metric (Bahr, 2006).

The primary contributions of this work are:
- propose a novel approach for accurate loss measurement
- propose a novel framework for interference measurement under the physical interference model
- propose a new interference-aware routing metric
- evaluate the performance of the newly proposed routing metric in an open-source real-world IEEE 802.11s WMN testbed.

The rest of this paper is organised as follows. In Section 2, motivation and relevant work is presented. Section 3 highlights the subtleties of link quality characterisation. Interference and contention measurements are addressed respectively in Sections 4 and 5. A new routing metric is presented in Section 6. In Section 7, we present the experimental settings and we finally conclude in Section 8.
2 Background and related work

2.1 Wireless Mesh Networks

Wireless Mesh Networks (WMNs) (Akyildiz et al., 2005; Amir et al., 2010) are emerging as a new technology with a rich set of applications, e.g., wireless community networks, wireless enterprise networks, transportation systems, home networking, and last-mile wireless Internet access. While the IEEE 802.11s mesh networking standard is still undergoing process, its main traits have been set, e.g., network architecture and MAC routing. IEEE 802.11s set HWMP (Hybrid Wireless Mesh Network) (Bahr, 2007) as the default routing protocol to be implemented in all IEEE 802.11s compliant devices. Airtime (Bahr, 2006) has been set as the default routing metric. The IEEE 802.11s TG (Task Group) previous meeting was held in May, 2010 (IEEE-TGs, 2010).

As multi-hop wireless networks, WMNs largely benefit from the valuable research led so far in multi-hop wireless networks in general, and particularly in MANETs (Mobile Ad hoc Networks) (IETF, 2010). MANETs, which failed to attract good civil applications, have tremendously contributed to the proliferation of WMNs: many WMNs protocols have been borrowed from MANETs. e.g., HWMP (Bahr, 2007), Radio-Metric Ad hoc On-demand Distance Vector (RM-AODV) (Aoki et al., 2006), Dynamic Source Routing (DSR) (Johnson and Maltz, 1996), Optimised Link State Routing (OLSR) (Clausen and Jacquet, 2003); The migration of such protocols profit from exemption on the stringent constraints of mobility and power consumption. These latter constraints are inherent in MANETs. However, such protocol migration should noticeably take into account major existing differences between MANETs and WMNs (Jun and Sichitiu, 2003).

In general, multi-hop wireless networks have been extensively addressed in literature, especially in Routing (Clausen and Jacquet, 2003; De Couto et al., 2003; Park and Corson, 1997; Perkins and Bhaghwat, 1994; Draves et al., 2004; Shila and Anjali, 2010). The capacity of such networks is deemed limited (Gupta and Kumar, 2000). To improve the capacity, different strategies have been proposed: load-balancing (Gao and Zhang, 2006; Kung et al., 2001; Shila and Anjali 2008; Bononi et al., 2009), multi-channel (Liu and Knightly, 2003; Raniwala and Chieu, 2005; So and Vaidya, 2004), multi-radio (Kung et al., 2001; Draves et al., 2004), packet scheduling (Lu et al., 1999), directional antennas (Ramanathan et al., 2006; Santivanez et al., 2005), multi-hop routing (Bahr, 2007; Clausen and Jacquet, 2003; Johnson and Maltz, 1996; Perkins and Bhaghwat, 1994; Perkins et al., 1999; Shila and Anjali, 2010), network coding (Markopoulou et al., 2007; Shen et al., 2007; Gollakota et al., 2007). However, all these strategies depend on one common issue: Link Quality Characterisation.

2.2 Routing metrics for link quality characterisation

When exposed to wireless link quality characterisation, we are first confronted with the large set of parameters that impact the overall wireless link quality, e.g., interference, loss rate, delay (and its variation, i.e., jitter), bandwidth, transmission power, transmission ranges, interference ranges, contention. Consequently, we ask which parameters should be considered (non) essential for link quality characterisation?
The common means for characterising wireless links are routing metrics. These differ in how they address the former question. While some metrics account solely for loss rate, others account additionally for bandwidth as well. Others account for delay, interference, or a combination of the formerly listed parameters. Given these different approaches, it becomes clearer that deciding on the critical or determinant link quality parameters is not a straightforward task. Another important point is that some routing metrics explicitly account for some parameters while implicitly accounting for others. For instance, RTT (Round Trip Time) (Padhye et al., 2004) explicitly accounts for delay. However, it implicitly accounts for contention and loss rate (The way it does so will be discussed later in this Section).

To further highlight the issue, we present next some of the most well-known metrics along with the parameters they account for. To emphasise how (i.e., explicitly or implicitly) a routing metric accounts for link quality parameters, we use the following notation:

\[ \text{metric} \_ \text{name}(E = \{p_1, p_2, \ldots \}, I = \{p_n, p_m, \ldots \}) \]

where \( p_i \) denotes the different link quality parameters, \( E \) is the set of explicit parameters, and \( I \) is the set of implicit parameters.

### 2.2.1 ETX (\( E = \{\text{Loss rate}\})

ETX (Expected Transmission Count) (De Couto et al., 2003) measures the expected number of transmissions needed to successfully transmit a frame. To compute ETX, every node periodically broadcasts \( N \) probe frames over a certain time period \( T \). Knowing \( N \) and \( T \), every node counts the number of successfully received probe frames. Then the (reverse) delivery ratio in the reverse link is computed as follows:

\[ d_r = \frac{R_r}{N} \]  

where \( R_r \) is the number of successfully received frames in the reverse direction.

Whenever a node broadcasts its \( N \) probe frames (always over the same time period \( T \)), it piggybacks on them the computed reverse delivery ratios \( (d_r) \) of all the one-hop links toward its neighbours. This way, every node can extract its forward delivery ratio \( (d_f) \) to any of its neighbours. Having both the forward and reverse delivery ratios to every neighbour, ETX is computed as follows:

\[ \text{ETX} = \frac{1}{d_f \times d_r}. \]

ETX accounts, explicitly and solely, for loss rate as a link quality parameter.

### 2.2.2 ETT (\( E = \{\text{loss rate, bandwidth}\})

ETT (Expected Transmission Time) (Draves et al., 2004) measures the required time for a frame to be successfully transmitted. ETT is computed as follows:

\[ \text{ETX} = \text{ETT} \times t \]
where $t$ is the average time for a single frame to be transmitted regardless of the transmission being successful or not.

$$t = \frac{S}{B},$$

where $S$ is the size of the probe frame, and $B$ is the link bandwidth.

ETT accounts explicitly for loss rate and bandwidth.

### 2.2.3 IEEE 802.11s Airtime ($E=\{\text{loss rate, bandwidth, channel characteristics}\}$)

IEEE 802.11s Airtime is a radio-aware metric that measures the amount of consumed channel resources when transmitting a frame over a particular wireless link (Bahr, 2006). Airtime is computed as follows:

$$\text{Airtime} = \left( O_{ca} + O_{p} + \frac{B}{r} \right) \times \frac{1}{1 - e_{p}}$$

(4)

where $O_{ca}$ and $O_{p}$ are constants quantifying the channel access overhead and the protocol overhead, respectively. $O_{ca}$ and $O_{p}$ depend solely on the underlying IEEE 802.11 modulation scheme (IEEE 802.11a/b/g). $B$ is the number of bits in the probe frame, and $r$ is the transmission rate (in Mbps). $e_{p}$ is the frame error rate. IEEE 802.11s did not set a specify way to measure $e_{p}$; It is left as a local implementation choice (IEEE, 2007).

Like ETT, Airtime accounts, explicitly, for loss rate and bandwidth. However, Airtime further accounts for the channel characteristics as well by means of $O_{ca}$ and $O_{p}$.

### 2.2.4 RTT ($E=\{\text{delay}\}, I=\{\text{contention, loss rate, queuing delay}\}$)

The RTT (Round Trip Time) (Padhye et al., 2004) of a hop is computed by sending a unicast probe frame bearing a timestamp. Upon receiving the probe, the receiver responds with a unicast frame echoing that probe. The initial timestamp (which is extracted for the echoed frame) is used to compute the elapsed time. The latter is the Round Trip Time (RTT).

Even though RTT accounts solely for delay as a link quality parameter, it also implicitly accounts for contention, queuing delay, and loss rate:

- RTT implicitly accounts for loss rate since a loss would induce re-transmitting the lost frame. Such re-transmissions increase the delay.
- RTT implicitly accounts for contention since high contention levels would increase the waiting time needed to grasp the medium, thus increasing the RTT.
- RTT implicitly accounts for the queuing delay as it is included in the RTT.
- RTT corresponds to the elapsed time of a frame leaving a station till it is received back.
- RTT accounts explicitly for delay, and implicitly for contention, queuing delay and loss rate.

### 2.2.5 PktPair ($E=\{\text{delay}\}, I=\{\text{contention, loss rate}\}$)

PktPair (Packet Pair) (Keshav, 1995) estimates the delay in a link by broadcasting two back-to-back frames: a short frame followed by a long frame. Only the first short frame is time-stamped. The receiver computes the difference in time between receiving the two frames, and echoes back the computed delay. In contrast to RTT, PktPair does not
account for queuing delays since the two frames are sent back-to-back (back-to-back frames will have the same queuing delay). However, PktPair implicitly accounts for contention and loss rate as in RTT.

PktPair accounts explicitly for delay, and implicitly for contention and loss rate.

### 2.2.6 WCETT (E={loss rate, bandwidth}, I={interference})

The WCETT (Weighted Cumulative Expected Transmission Time) (Draves et al., 2004) for a given path $p$ is computed as follows:

$$WCETT_p = (1 - \alpha) \times \sum_{j=1}^{N} ETT_j + \alpha \times \max X_j$$

where $X_j$ is the sum of the ETT values of all single hops, in a path $p$, that are operating in the same channel $j$ and $\alpha$ is a weighting parameter. $X_j$ accounts for channel diversity by accounting for the channel that may cause the highest intra-flow interference. This latter is as a result of hops operating in the same channel. WCETT was designed for multichannel multi-radio networks.

WCETT accounts for loss rate, bandwidth, and interference.

WCETT does not explicitly measure interference. It only approximates interference by accounting for the hops, in the same path, that are using the same channel (i.e., the number of links that can interfere with each other). In other words, WCETT is using a heuristic approach for interference measurement instead of an interference measurement model.

### 2.2.7 MIC (E={loss rate, bandwidth}, I={interference})

The MIC (Metric of interference and Channel switching) (Wang et al., 2005) for a given path $p$ is computed as follows:

$$MIC_p = \frac{1}{N \times \min(ETT)} \times \sum_{l \in p} IRU_l + \sum_{node \in p} CSC_i$$

where $N$ is the total number of nodes in the network. $IRU_l$ and $CSC_i$ are computed as follows:

$$IRU_l = ETT_l \times N_l$$

$$CSC_i = w_1 \quad \text{if} \quad CH(prev(i)) \neq CH(i)$$

$$CSC_i = w_2 \quad \text{if} \quad CH(prev(i)) = CH(i)$$

$$0 \leq w_1 << w_2$$

where $N_l$ is the number of neighbors interfering with link $l$. $CH(i)$ is the channel assigned to node $i$, and $prev(i)$ is its previous node along path $p$.

MIC is meant to account for both intra-flow and inter-flow interference: $CSC_i$ captures intra-flow interference by assigning it a high value ($w_1$) when the previous node is operating in the same channel. Otherwise, it is assigned a low value ($w_2$) otherwise. MIC accounts for inter-flow interference by scaling ETT (equation (7)) by $N_l$ (the number of interfering links).
MIC accounts for loss rate, bandwidth, and interference. However, like WCETT, MIC does not explicitly measure interference. It only estimates it by accounting for the number of interfering links, and the number of hops (in the same path) that are operating in the same channel. Similar to WCETT, MIC uses a heuristic approach for interference measurement instead of an interference measurement model.

2.3 Interference

In physics, interference is the addition (superposition) of two or more waves that result in a new wave pattern. In computer science, this superposition of signals/waves causes bit alterations. This in turn causes FCS (Frame Check Sequence) alteration. When the FCS of a frame is altered, the link layer drops the frame, thus generating a loss.

There is a general consensus about the impact of interference in wireless networks (Gupta and Kumar, 2000; Jain et al., 2003). Interferences degrade wireless network performance because of their direct and strong contribution to the generation of wireless losses. Interfering signals can originate from traffic in the same network or from undesired communications as well (e.g., microwaves, Bluetooth devices, and traffic in other networks). The accurate interference measurement is a challenging task for three main reasons:

- **High time complexity.** There are $O(n^2)$ links in a network with $n$ nodes. If only pairwise interferences are considered, then $O(n^2)$ links must be checked. In (Padmanabhan et al., 2005), we reduce this pairwise complexity from $O(n^2)$ to $O(n^3)$. However, this still only considers pairwise interferences. This does not represent real-world interference, where a signal can interfere with many signals (not just one).

- **Inadequacy of interference models for use in practical systems.** The protocol interference model (Gupta and Kumar, 2000), characterises interference by considering transmission and interference ranges. This presents strong inadequacy for use in practical systems due to the complexity of computing these ranges in ad-hoc topologies. The transmission/interference ranges vary in time because of mobile obstacles and variable transmission powers, rendering the computation more complex. Even, in a controlled-topology network, the computation is $O(n^3)$. The physical interference model (Gupta and Kumar, 2000), on the other hand, requires the simultaneous access to the SINR values of all interfering stations. This feature is unavailable in commodity NICs that report the signal strength of only one signal at a time. We refer back to this model with further details in Section 3.

- **Inadequacy of simulation tools.** Simulation tools are inadequate in representing interferences as they cannot capture the complex aspects of RF propagation such as multi-path fading, path loss, reflection, and diffraction. In indoor environments, the situation is further aggravated because of the unpredictable and mobile nature of obstacles, e.g., furniture, people, walls, etc. This has a direct and strong impact on the behaviour of most RF propagation characteristics, namely reflection, diffraction, and path loss. These facts render an accurate modelling of the interference phenomenon very complex. Thus, most simulation tools remain simplistic in the way they are modelling interference. For instance, ns-2 (Ns-2, 2010), the most widely used networking simulation tool in academia, uses a simplified version of the physical interference model (the capture threshold model) that accounts for only one
interfere at a time (Iyer et al., 2009). This is definitely not the case in real-world interference where there can be several simultaneous interferes.

3 Link quality characterisation

3.1 How to accurately measure wireless link quality

Given the large set of parameters that affects the overall quality of a wireless link (e.g., loss rate, transmission rate, interference, contention, delay, load, contention, transmission/interference ranges), we must first decide on the set of parameters to consider. There is a trade-off between keeping the parameters set as small as possible (in order to minimise measurements overhead) and having a representative parameters set capable of fairly characterising wireless link quality.

In the formerly reviewed link quality characterisation schemes (see Section 2), different parameters sets are used: while ETX (De Couto et al., 2003) accounts solely for loss rate, ETT (Draves et al., 2004), WCETT (Draves et al., 2004), and MIC (Wang et al., 2005) account for loss rate and transmission rate. Furthermore, WCETT and MIC account for interferences. On the other hand, RTT (Padhye et al, 2004) and PktPair (Keshav, 1995) account only for the delay. Other metrics account for the load as well (Gao and Zhang, 2006; Kung et al., 2001; Shila and Anjali, 2008). Hence, we see that deciding on the parameters set is a fuzzy process merely because of the correlations between these different link quality parameters: For instance, we see that incorporating both interferences and loss rate into link quality characterisation involves some redundancy since interferences are highly correlated to losses. Measuring both interference and losses means measuring the losses that are caused by interferences twice!

When seen from a network end-use perspective, link quality characterisation evolves around two main criteria:

- Receiving the whole requested data; this is characterised by the loss rate
- Receiving the requested data as quickly as possible; this is characterised by the link bandwidth, i.e., the transmission rate.

Thus, we state that loss and transmission rates are the parameters to be included in every optimal link quality characterisation approach. Still, other parameters can be included. However, we think that the previously mentioned trade-off between the parameters set granularity and the measurement overhead is highly worth considering. However, an accurate measurement of these parameters remains a challenging issue.

3.2 Need for accurate loss measurement

The common method for loss rate measurement in most routing metrics (e.g., ETX, ETT, WCETT, MIC) is the broadcast approach whereby periodical broadcast probe frames are used to ‘directly’ measure the losses. As presented in Section 1, the broadcast approach suffers three fundamental shortcomings, making loss rate measurement inaccurate:
Due to their broadcast nature, the probe frames add significant load to the network. Hence, they consume more network resources and increase the contention levels, thus negatively impacting the overall network performance.

In attempt to overcome the former shortcoming, large averaging periods are used (e.g., 10 seconds with 1 probe frame is a second (De Couto et al., 2003)). These large averaging periods do not respond to the time-varying nature of the channel link: In a wireless link, the channel quality can change in very short time periods, e.g., less than a second.

Broadcast frames are of fixed size and are sent at the ‘low’ constant rate. Hence they cannot represent the dynamics of real traffic frames which are sent using different frame sizes and different transmission rates (e.g., when using adaptive rate control schemes).

Taking these fundamental shortcomings into account, the broadcast approach would definitely yield inaccurate loss rate measurements. In attempt to overcome the broadcast approach shortcomings, the passive approach (Bhattacharjee et al., 2005) proposes the use of real traffic frames instead of broadcast frames. However, the passive approach suffers, in its turn, from the major shortcoming of probing idle links: Ideally, the passive approach should continuously track if a link is idle or not. When a link becomes idle, the passive approach should switch to the broadcast approach in order to generate some traffic. It should then switch back to the passive approach when the link becomes busy. Besides the continuous tracking of the idle-status of links, this switching back and forth will induce considerable overhead.

To overcome the shortcomings of both approaches, we propose the ‘indirect’ approach of estimating loss rate by measuring the causing events instead of directly measuring the losses (as with the broadcast and passive approaches). The events causing wireless losses are basically contention and interference. Unlike the passive approach, our approach does not suffer the fundamental shortcoming of probing idle links: The approach interprets the absence of traffic as valuable information to detect since this would indicate minimum contention and minimum interference. However, an accurate interference measurement is also required. As referred to Section 2.3, interference measurements impose inherent challenges. Next section highlights this issue and suggests a suitable solution.

4 Interface measurement

4.1 Interference measurement under the physical-interference model

4.1.1 The challenge

Before we state the problem of interference measurement, we should first select an underlying interference model. The protocol and the physical interference models are the most studied ones in literature (Iyer et al., 2009; Gupta and Kumar, 2000; Padmanabhan et al., 2005; Jain et al., 2003). Other interference models are basically variations of these models, e.g., the interference capture and the interference range models.

The protocol model is complex as it depends on measuring the transmission and interference ranges. This is hard to achieve on practical systems, especially in ad-hoc
networks where the topology is unpredictable. The physical model on the other hand is simpler as it relies solely on the signal strengths of interfering signals. This is the closest one to reality as it models the interference by accounting for the ‘physical’ quality of ‘wave superposition’, whereby interfering signals add up (i.e., superpose) to produce the resulting signal. In Iyer et al. (2009), we concluded that an SINR-based model is the minimum level of detail that should be employed to model wireless interference.

The physical interference model states that a communication from a node \( v \) to a node \( w \) is successful if and only if its SINR is above a certain threshold \( \beta \):

\[
\text{SINR}_{v,w} = \frac{SS_{v,w}}{N_w + \sum_{x \in I_w} SS_{x,y}} > \beta
\]  

(9)

where \( SS_{v,w} \) and \( SS_{x,y} \) denote the signal strengths of the unidirectional links \( |v, w| \) and \( |x, y| \). \( I_w \) is the set of nodes in the interference range of \( w \), and \( N_w \) is the thermal noise level at node \( w \).

Even though the physical model is the closest one to reality and exhibits a level of simplicity, it still imposes a fundamental problem with experimentation. In practice, applying the physical model inequality (9) is quite unfeasible for the following reasons:

- Commodity wireless card drivers can have access to the signal strength of only one frame at a time, whereas the physical interference model inequality (9) requires the simultaneous access to the signal strengths of all interfering frames.
- Commodity wireless card drivers can only track the signal strengths of successfully received frames, hence missing interference events.

In practice, we can estimate the average signal strengths of all interfering nodes but cannot know which nodes will be simultaneously transmitting. In such a non-deterministic situation, using probabilities is the natural and mathematical response. Accordingly, we propose weighting the signal strengths, in the physical interference model inequality (9), by probabilities of interference. The next section highlights this approach.

4.1.2 Our approach

When using the probabilities of interference as weights for the signal strengths, the physical model inequality (9) becomes:

\[
\text{SINR}_{v,w} = \frac{SS_{v,w}}{N_w + \sum_{x \in I_w} (P'_w(x,y) \times SS_{x,y})} > \beta
\]  

(10)

where \( P'_w(x,y) \) is the probability that a signal originating from node \( x \) towards node \( y \) will interfere, at node \( w \), with the signal originating from node \( v \) towards node \( w \).

This way, we propose a probabilistic approach for interference measurement under the physical model; where the signal strengths are weighted by the probabilities of interference. These probabilities would logically depend on the frame arrival rates of the concerned links: A pair of links operating at high transmission rates would have a high probability of interference than a pair whose transmission rates are lower. The next section highlights the issue and presents a solution.
To compute the interference probabilities $P_i^v(x, y)$ in equation (10), we need to model the IEEE 802.11 DCF traffic. In other words, we need to model the frame arrivals process. Assuming we have such a model, then the probability of two frames interfering can easily be computed. However, as far as we know, no frame arrivals model has been proposed yet for IEEE 802.11 DCF traffic: In IEEE 802.3 LANs, the traffic model was studied and it appears to be self-similar (Willinger et al., 1994). The remaining issue heretofore is “which model to use for IEEE 802.11 DCF traffic?”

Since no traffic model for IEEE 802.11 DCF has been proposed so far, and in an attempt to have this paper as the first one to propose a framework for interference measurement under the physical model using traffic modelling, we simplify the formerly stated problem by assuming that IEEE 802.11 DCF traffic follows a stochastic process. We then model frame arrivals using a Poisson process (Kingman, 1993). This also raises a fundamental case for future research that can provide a well-founded solution to accurate interference measurement under the well-known physical interference model. Last and not least, we further support our assumption by the following single argument:

- IEEE 802.11 DCF traffic is still exhibiting at least a considerable level of stochasticity/randomness through the use of the IEEE 802.11 backoff mechanism.

Under the former assumption, if the expected number of frame arrivals during a given time period $T$ is $\lambda$, then the probability that there will be exactly $k$ arrivals ($k$ being a non-negative integer, $k = 0, 1, 2, \ldots$) during $T$ is:

$$P(N = k, \lambda T) = \frac{e^{-\lambda T} \times (\lambda T)^k}{k!} \quad (11)$$

Let $F_{(v,w)}$ and $F_{(n,m)}$ denote the frames sent in the uni-directional links $(v, w)$ and $(n, m)$, and $f_{(v,w)}, f_{(n,m)}$ their respective arrival rates. Accordingly, $F_{(n,m)}$ would interfere, at $w$, with $F_{(v,w)}$ if and only if $F_{(n,m)}$ arrives at node $w$ during the time interval when $F_{(v,w)}$ is captured at receiver node $w$. Hence the time $T$, in equation (11), would correspond to the time interval during which frame $F_{(v,w)}$ occupies the channel. We denote it $\tau_{(v,w)}$:

$$\tau_{(v,w)} = \frac{S_{(v,w)}}{r_{(v,w)}} \quad (12)$$

where $S_{(v,w)}$ and $r_{(v,w)}$ are respectively the length and the transmission rate of frame $F_{(v,w)}$.

This way, the probability of having interference between two frames $F_{(v,w)}$ and $F_{(x,y)}$ corresponds to the probability of having one frame $F_{(x,y)}$ arriving during frame $F_{(v,w)}$ channel occupancy time $\tau_{(v,w)}$:

$$P_i^v(x, y) = P_r(N = 1, \lambda_{(v,w)} \tau_{(v,w)}) = e^{-\lambda_{(x,y)} \tau_{(v,w)}} \times \lambda_{(x,y)} \tau_{(x,y)}. \quad (13)$$

To avoid the overhead of computing the probabilities of interference for every single frame, we compute it over an averaging period $T$ instead of the frame transmission time $\tau$. Consequently, the probability of having an interference occurring between links $(v, w)$ and $(x, y)$ (where the former link is the desired one and the latter is the interfere) is the probability of having at least one frame $F_{(x,y)}$ sent during the channel occupancy time of link $(v, w)$:
631

An interference-aware routing metric for Wireless Mesh Networks

\[ P_r^e(x, y) = P_r(N \geq 1, \lambda_{x,y} \tau_{(v,w)}) = 1 - P_r(N = 0, \lambda_{x,y} \tau_{(v,w)}) = 1 - e^{-\lambda_{x,y} \tau_{(v,w)}} \]

Finally the physical model SINR inequality (11) converges as follows:

\[ SINR_{(v,w)} = \frac{SS_{(v,w)}}{N_e + \sum_{x \in \mathcal{N}_v} (1 - e^{-\lambda_{x,y} \tau_{(v,w)}}) \times SS_{(v,y)}}. \]

The \( \tau_{(v,w)} \) values are easily computed using equation (12). The arrival rates values \( \lambda_{x,y} \) can be easily obtained, as well, by tracking how many frames are sent in link \((x, y)\).

5 Contention measurement

In IEEE 802.11, contention refers to the situation where at least two stations are competing with each other for the free medium. The medium is sensed free when there is either no ongoing signal or when the ongoing signal(s) are below a certain carrier sense threshold.

We measure contention at a given node \( v \) by accounting for all frames that are heard at node \( v \), regardless of whether node \( v \) is the desired destination or not. We also account for the locally generated frames as these will generate contention in the vicinity of node \( v \). Thus, we define the Contention Indicator \( (CI_v) \) at a node \( v \), where \( 0 \leq CI_v \leq 1 \), as follows:

\[ CI_v = \frac{\sum_{i \in S} \tau_i}{\tau} \]

where \( P_i \) is the set of all frames heard at node \( v \), over a certain time period \( \tau \) (including frames sent or received by node \( v \)), and \( \tau_i \) is the transmission time of frame \( i \):

\[ \tau_i = \frac{S}{r} \]

where \( S \) is the frame size and \( r \) is its transmission rate.

6 Designing a new routing metric

Before discussing the issue of designing a new routing metric, we would like first to address two important relevant issues:

- the asymmetric nature of wireless links
- how to disseminate the computed SINR and contention values without using broadcast frames?
6.1 The asymmetric nature of wireless links

The quality of a wireless link depends on both the reverse and forward components of the link. The fact that most multi-hop routing protocols, e.g., AODV (Perkins et al., 1999), DSR (Johnson and Maltz, 1996), OLSR (Clausen and Jacquet, 2003), use the forward path in order to select the reverse path and vice versa, induces a radical contradiction when only the quality of the forward path is used to select the reverse path. To cope with this, it is essential for the routing metrics to incorporate both the qualities of the forward and the reverse components of a link into a single metric. For instance, when considering interference as a link quality metric, the interference levels at both destination and source should be incorporated into a single link quality metric since the link endpoints are likely to have different interference levels as a result of different power levels, different transmission rates, and different vicinities. The following general example further illustrates this point:

Let us formally denote \(|n, m)\) as the forward link from node \(n\) to node \(m\), and \(Q(n, m)\) as its quality, e.g., SINR, and let us assume the following arbitrary link quality values:

- \(Q(S, D1) = 3; Q(S, D2) = 2\)
- \(Q(D1, S) = 1; Q(D2, S) = 5\)

where \(S\) is a source node trying to select either node \(D1\) or node \(D2\) as its next hop.

If only the forward link quality is taken into account then \(S\) will pick \(D1\) since it corresponds to a better quality, e.g., larger SINR: \(Q(S, D1) > Q(S, D2)\). This disregards the fact that the reverse path \(|D2, S)\) is exhibiting a much better quality than the reverse path \(|D1, S)\). Ideally, \(S\) should select \(D2\) as its next hop instead of \(D1\) since the reverse path \(|D2, S)\) is much better than the reverse path \(|D1, S)\). This would cope with the difference in quality between the forward link components. When incorporating both link directions into a single metric, e.g., by using the product of the two link directions qualities, the selection of \(D2\) as the next hop proves logical:

- \(Q(S, D2) \times Q(D2; S) = 10 > 3 = Q(S, D1) \times Q(D1, S)\)

Hence it becomes very crucial to account for link quality metrics of both the reverse and forward components of a wireless link. The formerly defined SINR (14) and CI (15) parameters only account for one direction in a link. To take into account the asymmetric nature of the wireless links, we define the SINR of a link \((v; w)\) as follows:

\[
SINR_{(v; w)} = SINR_{(v; w)} \times SINR_{(w; v)}
\]  

(17)

where \(SINR_{(v; w)}\) and \(SINR_{(w; v)}\) are respectively the Signal-to-Interference-Noise ratios of the forward and the reverse unidirectional links \(|v, w)\) and \(|w, v)\). Similarly, we define the Contention Indicator of a link \((v; w)\), where \(0 \leq CI(v, w) \leq 1\), as follows:

\[
CI_{(v; w)} = CI_v \times CI_w = \frac{\sum_{e \in E_v} \tau_e \times \sum_{i \in I_v} \tau_i}{\tau_v^2}
\]  

(18)
6.2 How to disseminate the computed SINR and Contention values?

In order for every node to compute the asymmetric SINR and CI values for a link, it has to be aware of the forward SINR and CI values for all its neighbors. Consequently, whenever a node w computes the reverse SINR and CI values for all its neighbors, these values need to be disseminated to neighbors in order for them to be able to compute the asymmetric SINR and CI values according to equations (17) and (18). The question heretofore is: How to disseminate these values with minimum overhead?

Normally, broadcast probe frames are the means of disseminating such information. Since we are trying to minimise the overhead, we propose that every node piggybacks the computed SINR values for all reverse links to all its neighbors, as well as its own CI value, in the default IEEE 802.11 routing broadcast frames. Upon receiving a routing broadcast frame, every node looks up its MAC address, retrieves its forward SINR value, and combines it with the locally computed reverse SINR to compute the total asymmetric SINR value of the link.

6.3 A new routing metric

When designing routing metrics, e.g., ICE, one has to address the fundamental issue of whether or not a routing metric is suitable for a given routing protocol?

A systematic analysis of the relationships between routing metrics and routing protocols has been conducted in Yang and Wang (2008): It has been empirically proved that Isotonicity is a must for all flooding-based protocols, e.g., HWMP (Bahr, 2007) which is a variation of AODV (Perkins et al., 1999):

Given a quadruplet \( (S, \oplus, w, \leq) \), where \( S \) is the set of all paths, \( w \) is a function that maps a path to a weight (i.e., a routing metric), \( \leq \) is an order relation, and \( \oplus \) is the path concatenation operation, isotonicity is defined as follows:

- The quadruplet \( (S, \oplus, w, \leq) \) is isotonic if \( w(a) \leq w(b) \) implies both \( w(a \oplus c) \leq w(b \oplus c) \) and \( w(c' \oplus a) \leq w(c' \oplus b) \), for all \( a, b, c, c' \in S \).

To account for isotonicity, ICE for a given link \((v;w)\) is computed as follows:

\[
ICE_{(v,w)} = \frac{CI_{(v,w)}}{SINR_{(v,w)}} \times t
\]

(19)

where \( CI_{(v,w)} \) and \( SINR_{(v,w)} \) are respectively given by equations (17) and (18), and \( t \) is the transmission time. ICE of a given path \( p \) is the sum of the ICE values of its single-hop links:

\[
ICE(p) = \sum_{l \in L_p} ICE_l
\]

(20)

where \( L_p \) is the set of links in path \( p \). This way, ICE penalises routes with more hops.

Thus, we can easily see that ICE is isotonic since:

- ICE of a route is the sum of the ICE values of its hops.
- ICE of a hop is always a non-negative value.
To test the performance of ICE, and avoid the shortcomings of simulation tools when dealing with wireless indoor environments (see Section 2.3), we implemented a real-word indoor IEEE 802.11s WMN testbed. This implements the IEEE 802.11s HWMP routing protocol and the IEEE Airtime metric. The implementation is linux-based and open-source. The implementation details as well as the source code are made available online (Abid et al., 2010).

7.1 Topologies

We used two different topologies to test the new routing metric. In the first topology (see Figure 1), we deployed a wireless mesh network composed of 11 mesh nodes, one 802.11 client, and one 802.3 station. The 11-node WMN is connected to the internet.

Using Iperf (Iperf, 2010), we created UDP and TCP connections between the 802.11 client (connected to the WMN through a Mesh Access Point) and the 802.3 server (connected to the Internet).
In the second topology (see Figure 2), we connected two WMNs through the internet. The WMN on the left is composed of seven nodes, and the WMN to the right is composed of eight nodes. Similar to the first topology, UDP and TCP connections were created using Iperf. The major difference in this topology is that both the client and server are 802.11 stations located into two different WMNs.

Figure 2  WMN testbed, topology 2 (see online version for colours)

These WMNs were deployed on the second floor of the Shelby Centre for Engineering Technology at Auburn University.

7.2 Settings

The mesh points network interface cards run the Madwifi driver (Madwifi, 2010). The HWMP PREQ (Path REQuest) messages are periodically sent every 4 seconds. The ETX probe frames are 1024 Bytes long and sent every 1 second (De Couto et al., 2003). Transmission rates were extracted through the ‘/Proc’ system files, and the default rate adaptation algorithm was used, which is SampleRate (SampleRate, 2010) in Madwifi drivers. Madwifi allows for the creation of multiple virtual interfaces. Two virtual interfaces were created for ICE computation in every WMN node: One virtual was set to the Monitor mode, and the other to the ad-hoc mode. In the monitor mode, the network interface card can overhear all ongoing signals and access their characteristics, e.g., RSSI (Received Signal Strength Indicator), transmission rate, and frame length.

In both topologies, the WMN network interface cards are operating in the 802.11 g channel(1) while the clients are operating in the 802.11g channel(11). This is done to minimise the interference between the client and the WMN nodes.

Both TCP and UDP traffic were supported in order to fully ascertain the functionality of the WMN testbed. Real web surfing sessions, using the WMN testbed as a backhaul, were tested and successful.

During experiments, the TCP and UDP sessions were repeatedly run, using Iperf, over periods of 20 seconds. These were average.
7.3 Results

In both topologies, we see that ICE generally outperforms both ETX and Airtime (See Figures 3–6). In topology 1, ICE outperforms ETX by an average of 15% for UDP, and 24% for TCP traffic. ICE also outperforms Airtime by an average of 23% for both UDP and TCP traffic (See Table 1).

Figure 3  UDP throughput – topology 1 (see online version for colours)

Figure 4  TCP throughput – topology 1 (see online version for colours)
An interference-aware routing metric for Wireless Mesh Networks

Figure 5  UDP throughput – topology 2 (see online version for colours)

![UDP Throughput - Topology 2](image)

Figure 6  TCP throughput – Topology 2 (see online version for colours)

![TCP Throughput - Topology 2](image)

Table 1  UDP and TCP throughput averages (Mbps)

<table>
<thead>
<tr>
<th>Routing metrics</th>
<th>Topology 1</th>
<th></th>
<th>Topology 2</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>UDP</td>
<td>TCP</td>
<td>UDP</td>
<td>TCP</td>
</tr>
<tr>
<td>ETX</td>
<td>1.18</td>
<td>0.95</td>
<td>1.19</td>
<td>0.94</td>
</tr>
<tr>
<td>Airtime</td>
<td>1.10</td>
<td>0.96</td>
<td>1.28</td>
<td>0.92</td>
</tr>
<tr>
<td>ICE</td>
<td>1.36</td>
<td>1.18</td>
<td>1.41</td>
<td>1.01</td>
</tr>
</tbody>
</table>
In topology 2, ICE outperforms ETX by an average of 18% for UDP, and 7% for TCP traffic. ICE also outperforms Airtime by an average of 10% for UDP traffic and an average of 9% for TCP traffic.

However, we note that the performance over IEEE 802.11g is dismal and hope that the MAC layer of IEEE 802.11s will offer a better performance.

8 Conclusion

This paper proposes
- a novel interference approach for loss rate estimation that addresses the issues of the current broadcast and passive approaches
- a novel scheme to measure interference by adapting the physical interference model.

These schemes offer a solution that can be implemented on real world mesh networks. This approach may open a new research venue in such a direction, i.e., the adoption of the physical model for interference measurement. While promising, the proposed schemes suffer from some limitations: they do not address hidden terminals and use the simplistic Poisson distribution for packets inter arrival time.

To evaluate ours schemes, we implemented them on a real world IEEE 802.11s wireless mesh network that we built on top of IEEE 802.11g. We designed and implemented the new interference-aware routing metric (ICE). This paper compares ICE to ETX and Airtime using an indoor pre-IEEE 802.11s Wireless Mesh Network test bed that we implemented.

Experiments showed that ICE outperforms ETX and Airtime by an average of 16% extra throughput. This relatively good improvement comforts the ICE proof of concept. Still, we believe that ICE outperformance could be greater if deployed in large-scale WMNs with more path redundancy: in relatively small-sized topologies, all tested routing metrics (e.g., ETX, Airtime, and ICE) tend to select the same paths because there are not many alternatives. Therefore, ICE does not have opportunities to use its extra knowledge to select better routes Unfortunately, we could not verify the latter conjecture due to limited resources (number of stations and lack of a larger space). However, the fact that ICE is a passive approach guarantees outperformance thanks to the use of data traffic frames instead of broadcast frames, and insures further accuracy by use of real traffic. In contrast to the passive approach, ICE does not suffer the shortcoming of probing idle links as the absence of frames is a quality to detect by itself, and could be interpreted to represent minimum interference and contention levels in absence of hidden terminals.

As future work, we consider two directions. First, we plan to investigate the adaptation of ICE to multi-channel and multi-radio links: At this time, ICE currently accounts solely for inter-flow interferences. For multi-channel multi radio, ICE must account for intra-flow interference. Second, we are considering different frames arrival distributions instead of the Poisson distribution: a good candidate is the self-similarity Pareto distribution. Self similar distributions proved to model Ethernet traffic. Since IEEE 802.11 Ethernet are close by using CSMA and the same backoff mechanism, we expect more accurate loss rate measurements, Ultimately, we aim to implement wireless mesh networks that offer a performance similar to what users get with one hop IEEE 802.11g access points.
References


Iperf (2010) The Iperf Project, Available at: http://dast.nlanr.net/Projects/Iperf/


An interference-aware routing metric for Wireless Mesh Networks


