Improving End-to-end Packet Delivery in High Traffic Multi-hop Wireless Ad Hoc Networks

Syed Rehan Afzal*, Majid Nabi*, Sander Stuijk*, and Twan Basten*†
* Eindhoven University of Technology, The Netherlands
† TNO-ESI, Embedded Systems Innovation by TNO, The Netherlands
{s.r.afzal, m.nabi, s.stuijk, a.a.basten}@tue.nl

ABSTRACT

Accurate link quality estimation is an important building block in quality aware routing. In an inherently lossy, unreliable and dynamic medium such as wireless, the task of accurate estimation becomes very challenging. Over the years ETX has been widely used as a reliable link quality estimation metric. However, more recently it has been established that under heavy traffic loads ETX performance gets significantly worse [4, 18, 19]. Contributions made in this paper are twofold. Firstly, we examine the ETX metric's behavior in detail with respect to the MAC layer and UDP data; and identify the causes of its unreliability. Secondly, we present the design and implementation of the xDDR link quality measurement metric – a variation of ETX - motivated by the observations made in our analysis. Our experiments show that xDDR substantially outperforms minimum hop count, ETX and HETX in packet delivery ratio.

Categories and Subject Descriptors

C.2.1 **[Computer-Communication Networks]**: Network Architecture and Design — *Wireless communication*

General Terms

Design, Experimentation, Measurement, Performance

Keywords

Asymmetric link quality, Link-quality measurement, Wireless ad hoc networks

1. INTRODUCTION

A wireless transmission medium is inherently lossy, dynamic as well as unpredictable. The goal of quality driven routing is to achieve a more deterministic behavior such that the network resources can be efficiently utilized resulting in improved performance in terms of delivery ratio, bandwidth, latency, throughput etc. Accurate link quality estimation is pivotal and paramount for the success of quality driven routing. A number of quality driven metrics have been proposed over the years where throughput and packet delivery ratio have gained special interest. Among them the Expected Transmission Count (ETX) [7] metric has been one of the most widely used metrics aimed at improving end-to-end throughput in wireless networks.

In this paper we discuss the state of the art and make three main contributions. Firstly, we experimentally investigate in detail the accuracy and effectiveness of ETX estimates and demonstrate the main reasons for the metric's short-comings. We examined the role of MAC layer in influencing and effecting ETX estimates at network layer. Several papers in the past have explored MAC layer alongside ETX but only to advocate a cross-layer link quality estimation solution [20, 21]. In this paper we present the first experimental study on the relationship of ETX metric's link estimation with respect to varying MAC retransmission limit. Secondly, we propose xDDR (estimated Directional Delivery Ratio) an improved link quality estimation mechanism that is motivated by the observations made from the behavior of ETX. Comparisons in end-to-end packet delivery ratio confirm the overall gain yielded as a result of more accurate link level estimates.

2. RELATED WORK

Finding the best quality-aware metrics in order to improve routing and network throughput has attracted much attention in recent years. Apart from hop count other widely used metrics include RTT (Per-hop Round Trip Time) [10], PktPair (Packet Pair delay) [11], ETX (Expected Transmission count) [7] and ETT (Expected Transmission Time) [13].

RTT [10] calculates round trip times between neighboring nodes by sending beacon packets at periodical times. Receiving nodes acknowledge the packet by a reply encapsulating the timestamp of reception. The calculation of RTT thus accounts for queuing delay, channel contention and lossy links. In PktPair, each node uses two back-to-back packets (one small, and one large) to its neighbors. Receiving nodes in return calculates the delay between the two packets and reports it back to the sender. The sender node then uses an exponentially weighted moving average of these delays with respect to each neighbor. The routing algorithm then chooses the path with the minimum delay. The advantage of PktPair over RTT is that it eliminates the queuing delay problem that exists in RTT [5].

The ETX metric is a routing metric particularly for finding high end-to-end throughput paths in multi-hop wireless networks by measuring link-level packet loss rates. ETX has been extensively shown to outperform other routing metrics in yielding higher throughputs in static wireless multi-hop networks [15, 16, 17]. Draves et al. [5] compared minimum hop count, RTT, PktPair, and ETX for setups with varying bandwidths on the DSR [15] routing protocol. They established that ETX outperforms the other three metrics in a static wireless network.

As our scheme is similar to ETX, we will discuss it in more detail in this section and Section 4. The ETX of a link is the estimated number of retransmissions required to send a packet over that link. ETX of a link can be written as ETX(link) $=1/d_f \times d_r$. DeliveryForward (d_f) is the probability of successful packet delivery in the forward direction while the opposite direction is DeliveryReverse (d_r). To calculate d_f, each node broadcasts beacon messages with period etxTimeInterval (ε) to its neighbor nodes. d_f then is the ratio between number of beacons sent and number of beacons received within a fixed window etxWindowSize by the neighbor. Similarly, d_r is the rate of success of beacon messages in the opposite direction. ETX of a route is the sum of ETX of all the links comprising that particular route. Among competing routes for the same stream, the route with the minimum ETX sum is recognized to concede the highest throughput. Although, in our observations regarding ETX's throughput performance in mobile ad hoc networks, we found that ETX is not always the best in estimating link quality. Similar observations sometimes discussed in different context have also been reported in the past.

Waharte et al. [18] revealed that the throughput from ETX is similar, and in some cases even worse than using a hop-count metric. They conclude that in case of increased traffic load, the interference avoidance strategy of ETX does not lead to a better end-to-end throughput. It was inferred that this is because it selects longer paths which add more self-interference and may lead to flow starvation [5]. Furthermore [19] and more recently [4] found that during high traffic scenarios throughput of ETX deteriorates considerably mainly due to collisions between route discovery broadcast (RREQ) packets and ETX beacons. Tran et al. [4] proposed HETX which circumvents this effect by calculating ETX over a previous window if the current ETXwindow overlaps with RREQ discovery times. Finally Das et al. [3] performed comprehensive measurements over a two mesh test bed to study the instability in ETX link quality measurements. Their experiments showed that introducing a transfer of just one large file in the network resulted in link ETX value increases up to 10000% suggesting much lower throughput than actually experienced by the link.

3. NETWORK ARCHITECTURE

This section describes the detailed network architecture as well as the quality metrics employed. Our wireless network is modeled as a directed graph comprising nodes and edges (N, E) transmitting connectionless UDP streams between each other where $\mathrm{N} = \{\mathrm{n}_1, \mathrm{n}_2, \ldots, \mathrm{n}_\mathrm{g}\}$ and $\mathrm{E} = \{\mathrm{e}_1, \mathrm{e}_2, \ldots, \mathrm{e}_\mathrm{h}\}$. An edge <$n_1$, n_2> is a directional link from *rootNode* $n_1 \in N$ to *childNode* $n_2 \in N$ when node n_2 is within the transmission range of n_1 (1-hop neighbor). We use $\mathbb{N}$ to represent the set of natural numbers, $\mathbb{R}$ represents the set of real numbers and $\mathbb{R}^{\geq 0}$ to represent the set of non-negative real numbers. Since our nodes communicate in a connectionless environment, in-band signaling mechanisms cannot be employed at the network layer in order to compute link-level delivery ratios. This is because of the fact that intermediate nodes are transparent to the stop time and data rate of buffered packets received from their respective 1-hop neighbors. Therefore, between a pair of in-range nodes such as n_1 and n_2, we compute the directional delivery ratio estimate represented as $\vec{\lambda}(\mathrm{n}_1, \mathrm{n}_2, \mathrm{t}') \in \mathbb{R}$ with t' the current timestamp. $\vec{\lambda}(\mathrm{n}_1, \mathrm{n}_2, \mathrm{t}')$ may be unequal to $\vec{\lambda}(\mathrm{n}_2, \mathrm{n}_1, \mathrm{t}')$ due to the asymmetry in link delivery ratios widely observed in wireless mobile communication [1] and $0 \leq \vec{\lambda} \leq 1$.

For a given node n_m, we have $E_m \subseteq E$ consisting of directional edges with the first-hop neighbors of n_m where $n_m \in \mathbb{N}$. We have a set of connectionless UDP streams S, to transmit from a source to a specific destination node. A given stream $s \in S$ is described as follows:

$$\forall s \in S = s(n_k, n_l, f, t_{st}) \in N \times N \times \mathbb{R}^{\geq 0} \times \mathbb{R}^{\geq 0}$$

Where, n_k is the stream source, n_l is the stream destination, f is the data rate of the transmission and t_{st} represents the start time. For a given stream $s \in S$, $\mu(s, t')$ denotes total number of packets transmitted till timestamp t'. Similarly $\delta(s, t')$ represents the total number of packets successfully received by destination node n_l till time t' where $\delta \leq \mu$. Consequently, for a given stream the end-to-end packet delivery ratio (PDR) is:

$$\mathrm{PDR}(\mathrm{s}, \mathrm{t}') = \frac{\delta(\mathrm{s}, \mathrm{t}\prime)}{\mu(\mathrm{s}, \mathrm{t}\prime)} \times 100 \quad (1)$$

In our network link-layer feedback is active and sender's MAC layer attempts to retry packets that are unsuccessful in first attempt for (ɍ-1) times where ɍ represents the retry limit. A stream transmission can only be initiated if there exists one or more possible routes represented as set $s^\tau(n_k, n_l, f, t_{st}, t'_1) = \{\tau_1, \tau_2, \tau_3, \ldots\}$ to reach from the stream source n_k to destination n_l. t'_1 represents a given timestamp and therefore $s^\tau(n_k, n_l, f, t_{st}, t'_1)$ may or may not be equal to $s^\tau(n_k, n_l, f, t_{st}, t'_2)$. Since we employ source routing, each route entails a list *nodelist* comprising ordered, disjoint and loop free set of nodes that can act as relay nodes between the stream source and destination. In Figure 1, the three route options $\{\tau_1, \tau_2, \tau_3\}$ for transmission from n_k to n_l includes nodelist $\{n_1, n_4, n_7\}$, $\{n_5, n_8\}$ and $\{n_3, n_6, n_9, n_{10}\}$ respectively. A route τ therefore can be decomposed into {source, node list, destination}. In other words, the route $\tau_1 = \{n_k, n_1, n_4, n_7, n_l\}$ where the expression $n_1 \in \tau_1$ indicates that node n_1 is included in the route τ_1. Moreover, $\vec{\sigma}(\tau_1, t')$ is the end-to-end directional delivery ratio estimate for a given route τ_1 at timestamp t', calculated from link-level delivery ratios $\vec{\lambda}(n_k, n_1, t'_1)$, $\vec{\lambda}(n_1, n_4, t'_2)$, $\vec{\lambda}(n_4, n_7, t'_3)$ and $\vec{\lambda}(n_7, n_l, t'_1)$ corresponding to all the edges present in the route τ_1 (further discussed in Section 4 and 5). Finally, we have a set τ'_s that comprises the route ids of only the routes that are employed by the stream s for transmission.

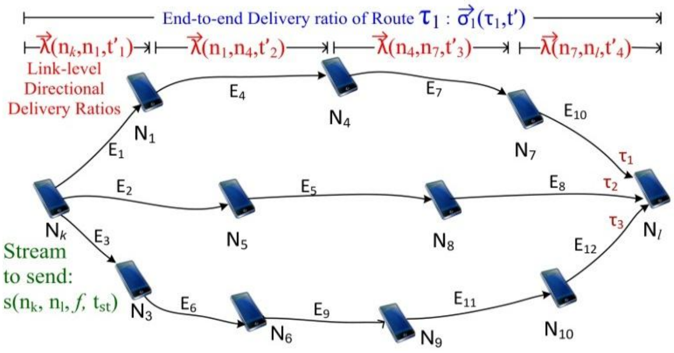
End-to-end Delivery ratio of Route τ_1 : $\vec{\sigma}_1(\tau_1, t')$
$\vec{\lambda}(n_k, n_1, t'_1)$
$\vec{\lambda}(n_1, n_4, t'_2)$
$\vec{\lambda}(n_4, n_7, t'_3)$
$\vec{\lambda}(n_7, n_l, t'_4)$
Link-level Directional Delivery Ratios
N_1
N_4
N_7
E_1
E_4
E_7
E_{10}
τ_1
τ_2
τ_3
E_2
E_5
E_8
N_k
N_5
N_8
N_l
E_3
E_{12}
Stream to send: $s(n_k, n_l, f, t_{st})$
N_3
E_6
E_9
E_{11}
N_6
N_9
N_{10}

Figure 1: Network topology

Our goal is to accurately estimate $\vec{\sigma}(\tau, t')$ using link level delivery ratios $\vec{\lambda}$ for all possible route options available for a given stream. This facilitates the communicating devices to select the route with the highest expected PDR among the contending route options.

4. MOTIVATION: xDDR (Expected Directional Delivery Ratio)

Before we get into the detailed description of the protocol, it's imperative to look into the importance of accurate link quality estimates as well as the factors that affect link quality estimation. Link quality estimation in wireless networks is a challenging problem due to the lossy and dynamic behavior of wireless links [2]. In connectionless communication environments an in-band signaling mechanism cannot be employed to compute and monitor link-level delivery ratios at the network layer. This is due to the fact that the network layer on the intermediate nodes is transparent to the stop time and data rate of buffered packets received from their respective 1-hop neighbors. We therefore need to estimate the link quality as opposed to measuring it. Several techniques such as RTT, PktPair, ETX and ETT employ proactive control messaging to estimate the link quality.

4.1 Importance of accurate estimates

Accurate link quality estimation is one of the most pivotal building block in quality aware routing as this information is later employed by routing/path selection modules to make local and network wide decisions. The main focus of multi-hop quality driven routing protocols is to improve overall network capacity and/or the performance of individual streams. Accurate localized quality measurements play an integral role towards appropriate end-to-end or network wide decision making particularly in the following situations: (a) selecting *best relay nodes* in order to reduce the recovery cost of frames lost due to link-quality fluctuation; (b) identifying *high-quality channels* since due to the use of shared medium link-qualities differ from channel to channel within the same node; (c) anticipating *route failures* by diagnosing and identifying faulty nodes/links and regions and (d) supporting *end-to-end QoS* driven applications, where accurate aggregated wireless link-quality information enables communicating applications such as VoIP/IPTV to adapt streams to sustainable service levels. On the other hand, inaccurate estimates can cause the participating nodes to continuously select bad nodes/links which in turn may affect the end-to-end QoS.

4.2 Problems with ETX

As discussed before, ETX has been widely used as a link-quality estimation metric and has been shown to outperform minimum hop-count, RTT and PktPair. More recently however it has been shown that the ETX metric performs poorly under high traffic environments [4]. We ran a series of experiments to study the ETX link metric behavior especially trying to reproduce previously claimed short-comings.

The authors of [3] revealed the instability of ETX in their experiments where they used a large TCP flow for downloading a file between a pair of communicating nodes. Their results indicate that the ETX values of 30% of the links increase disproportionately to about 10000% after adding only the large traffic load TCP flow. The authors suggest that ETX-based routing protocols use a random number for measuring link quality. However, a detailed investigation of such claim was not performed. We investigated the observations made by [3] and calculated that a 10000% link ETX value increase is possible, when only 1% of ETX packets get successfully sent and received between a pair of in-range nodes. In other words, if for instance our etxWindowSize is 100, the etxTimeInterval rate is 0.1 sec and only one packet gets successfully sent and received between a pair of in-range nodes during the entire etxWindowSize (1 ETX packet in 10 seconds duration); in this case deliveryForward (d_f)=0.01 and delivery-Reverse (d_r)=0.01. Consequently:

$$Link\ ETX = \frac{1}{0.01 \times\ 0.01} = 10000$$

This reflects a 10000% increase from the highest ETX value of 1 when both d_f and d_r are 1. In our experiments, even in heavy traffic (CBR-1000) and heavy node densities (70 nodes in 500 m×500 m), we observed as much as 5200% increase in individual link ETX values (Figure 3). We believe the reasons for such inaccurate estimations are the same as reported in [3].

4.2.1 *Role of MAC layer and the broadcast problem:*

To understand what impact the MAC layer has on the behavior of ETX, we would looked at the correlation between PDR estimates calculated using ETX broadcast and actual data transmission with respect to varying MAC retransmissions. We used the INETMA-NET [9] extension running on the OMNeT++ [8] simulator for our experiments. We only use the deliveryForward (d_f) which in principle is the probability of packet delivery from one node to another [4, 7] and compare its relationship with the actual packet delivery ratio.

Algorithm 1 Directional Delivery Estimate and PDR

```
1: Compute link-level directional delivery ratio using ETX broadcast:
    $\vec{\lambda}(n_Q, n_R, t_1)$, where $n_Q, n_R \in N$ & $Q \neq R$
2: Stream to send $s(n_k, n_l, f, t_{st})$ with start time $t_{st}$ which attempts to transmit till the end of simulation at data rate $f$.
3: Compute estimated end-to-end directional delivery ratio $\vec{\sigma}$ for each available route using $\vec{\lambda}$ of all links in the respective route:
    $\vec{\sigma}(\tau_x, t_2) = \prod_{n_O, n_P \in \tau_x}(\vec{\lambda}(n_O, n_P, t_2))$, where $O \neq P$
    ($t_1$ and $t_2$ represent discrete timestamps)
4: Stream $s(n_k, n_l, f, t_{st})$ selects a route $\tau_x$ if and only if
    $max_{\tau_U \in s^\tau}[\vec{\sigma}(\tau_U, t_2)] = \vec{\sigma}(\tau_x, t_2)$
```

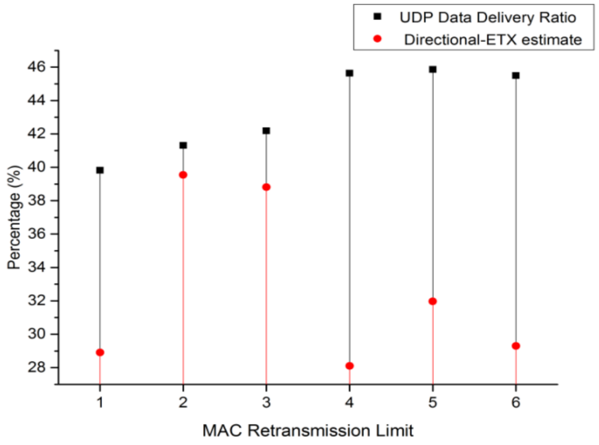

Figure 2: Data Delivery Ratio v/s Directional ETX estimate

A selected route τ_x continues to serve the stream until one or more links becomes too lossy and result in a RouteError (RERR) message, eventually causing a new discovery. However we want to study the correlation between our delivery ratio estimates and the actual delivery ratio achieved by the network. For this purpose, throughout the simulation run of 200 s, we superimpose a new route discovery every 10 seconds by erasing all routing tables in the network. The data rate is 500 packets per sec (CBR-500), the etxWindowSize is 100 and the etxTimeInterval rate is 0.1sec. The network comprises 30 nodes on a 400 m × 400 m terrain with varying MAC retransmission limits (ɍ: 1–6). The sender and the receiver of the streams are placed on the diagonals at coordinates respectively. For experimental correctness we ran 20 trials for each of the six retransmission limits where each trial entails a different placement of all the nodes except the stream’s sender and receiver.

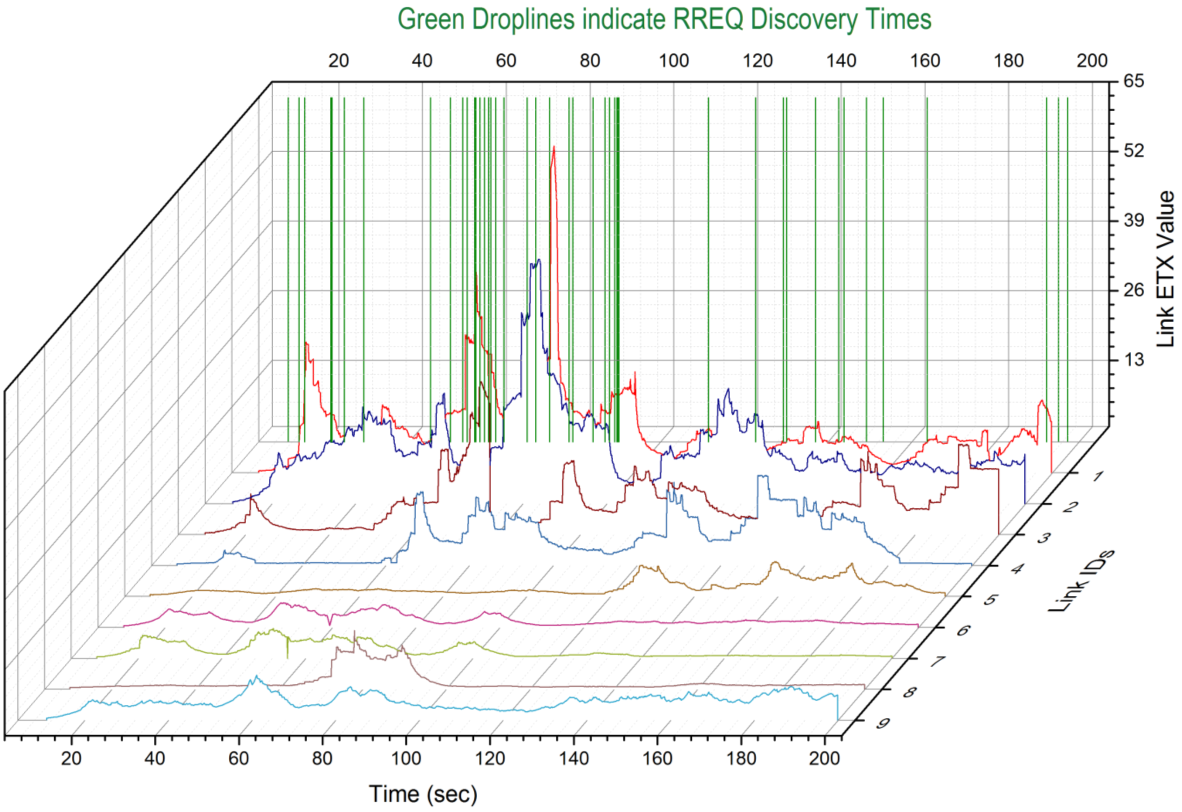

Figure 3: Link ETX estimate variation and route discoveries

For each route discovery, the receiver node suggests a route with the highest cumulative PDR estimate $\vec{\sigma}$ to carry on the transmission. We compute average $Avg[\vec{\sigma}(\tau_s')]$ for all the routes τ_s' employed during the course of each simulation as well as the actual PDR yielded from selecting those routes. For each MAC retransmission limit ɍ, $Avg[\vec{\sigma}(\tau_s')]$ and PDR are further averaged for the 20 trials and then compared in Figure 2. We can observe that our actual data delivery ratio shows a linear increase as we increase the MAC retransmission limit and starts to saturate at retry limit 4-6 (since, at some stage the overhead of retransmitted packets start to cause more link contention than gain in the data delivery ratio). On the other hand, the ETX-based delivery ratio estimate behaves very differently. While our unicast UDP traffic is getting higher delivery as a result of an increase in MAC retransmissions, the ETX broadcast packets are suffering more losses resulting in lower delivery ratio estimates. Statistical analysis shows that the Pearson correlation coefficient of Figure 2 is -0.24 between the delivery ratio estimate and the actual data delivery ratio. This implies weak/low correlation indicating ETX isn't a good estimator of data delivery ratio. This somewhat random behavior in link estimation is also visible in other setups that we ran with different traffic rates or node densities but is more evident in high traffic scenarios. We believe this insufficient correlation between ETX and actual packet delivery ratio ensues due to a number of reasons. First and foremost is the intrinsic difference in how the physical layer and MAC layer treat broadcast packets as compared to unicast. The broadcast probe packets used for ETX are small and are sent at the lowest possible data rate (1Mbps for 802.11b/g, 6Mbps in case of 802.11a) [6]; they may not experience the same loss rate as data packets sent at higher rates. Furthermore when the buffer gets full MAC gives lower priority to the in-buffer unicast packets than broadcast and starts discarding unicast packets first. Moreover, the metric does not directly account for link load or data rate. A heavily loaded link may have very low loss rate, and two links with different data rates may have the same loss rate.

Furthermore, in high traffic scenarios wireless channels rely heavily on the underlying link-level feedback and RTS/CTS mechanism provided by the retransmissions; a feature which is only applicable to unicast packets.

4.2.2 *ETX under Heavy Traffic load:*

In wireless environments heavy communication traffic load results in more packet drops, packet buffer overflows as well as link breakages. In MANETs such service denials cause routing errors, thereby initiating route discovery via broadcast flooding. The authors of [4] and [9] studied the behavior of ETX under such route request packets as well as heavy traffic in general. They observed that higher traffic load entails more frequent link route discoveries and eventually a greater probability of incorrect ETX estimation. To study this behavior, we ran high traffic simulation with 5 senders sending heavy UDP traffic of data rate 1000 packets per seconds (CBR-1000) towards 5 distinct receivers. The etxWindowSize is 100 and the etxTimeInterval (ε) is 0.1 s in a network comprising 70 nodes on a 500 m × 500 m terrain with the MAC retransmission limit ɍ set to 6. Figure 3 illustrates ETX estimate variation of 9 randomly picked links over a duration of 200 seconds. The green colored vertical drop lines indicate the times route discoveries are initiated. The figure shows that Link-5 retains an ETX value of approximately 1 until 120 seconds indicating that nearly all ETX packets transmitted over this link have been successful. Furthermore, we see that some links exhibit much higher fluctuations in ETX value distribution as compared to others. Link-1 and Link-2 experience an ETX increase as high as 5200% and 3000% respectively.

In such high constant data rate, the routes discovered are not stable and recurrently broken; resulting in numerous new consecutive route discoveries and consequently increasing the traffic load. This continuous flooding further increases the probability of ETX broadcast packet failures, as compared to unicast data packets, leading to wrong estimates. This behavior can cause serious degradation in the network PDR. When link quality measurements are affected by flooding of request discoveries, nodes may choose low quality paths that may easily be broken. This continuous selection of bad paths may continue repeatedly and thereby adversely affect the overall network's performance.

5. xDDR – Detailed Metric and Protocol Description

In this section, we detail our link quality estimation metric xDDR (expected Directional Delivery Ratio) – a variation of ETX – motivated by the observations made in the previous section. We highlighted two main attributes affecting ETX link quality estimates: (1) MAC layer's disparity in handling with respect to broadcast versus unicast, and (2) the effect of route broadcast flooding on link quality estimates. xDDR is a selective unicast beaconing mechanism that estimates directional delivery ratios of links over a fixed window and attempts to exclude link's performance during the route discovery flooding phase. Using xDDR to achieve high-PDR end-to-end paths involves two main processes, namely: (1) selective proactive beaconing; (2) a reactive routing module translating link quality estimates to path quality estimates, and providing sub-modules for route selection and reply.

5.1 Selective Proactive Beaconing

We established in the previous section that broadcast packets do not employ the MAC layer RTS/CTS mechanism and are transparent to the MAC retransmission limit resulting in a weak correlation between link quality estimates and delivery ratio of the actual traffic that is entirely based on unicast transmissions. In order to identify the 1-hop neighbors, in the initialization phase each node broadcasts proactive beacon messages. During this phase each node $\mathrm{n_M}$ maintains a list of its first neighbors $\mathrm{E_M}$. Node $\mathrm{n_M}$ then transmits the proactive unicast beacon message addressed to the address of its 1st hop neighbors. Unicast beacons require a particular address to send to and each network layer packet may entail multiple physical packet transmissions (depending on the retransmission limit ɍ and successful delivery). In order to contain the extra overhead we limit the number of unicast beacon recipients. Node $\mathrm{n_M}$ therefore selects a randomly picked subset of $\mathrm{E_M}$ as unicast beacon recipients (represented as $\mathrm{E'_M}$) to transmit packets at rate etxTimeInterval ε. In a broadcast based scheme each node transmits ε×Ŧ number of packets where Ŧ represents the network lifetime. In case of xDDR, the number of packets transmitted physically varies from node to node as a result of additional factors such as $\mathrm{E'_M}$ and MAC retries. The best case therefore is $\varepsilon \times |\mathrm{E'_M}| \times \text{Ŧ}$ and $\varepsilon \times |\mathrm{E'_M}| \times (\text{ɍ} - 1) \times \text{Ŧ}$ is the worst case. In our experiments we limit each node to have at most 4 unicast beacon recipients. The benefit of using unicasts instead of broadcasts is two-fold. Firstly, the unicast beacon messages adhere to the RTS/CTS mechanism just as the regular data packets. Secondly, it estimates data delivery ratios while accounting for the underneath MAC retransmission limit.

Each of the recipients from $\mathrm{E'_M}$ maintains a window xDDR-Window logging timestamps of the packets received from $\mathrm{n_M}$. This window is used to derive the xDDR measurement window represented as ŵ which in essence is the window used to compute link-level directed delivery ratio $\vec{\lambda}$. The xDDRWindow in practice is twice the size of the ETXWindow. We have to be careful with the size of ETX and xDDRWindow size since it directly affects the freshness of the link quality estimate. The purpose of a longer window is to (whenever possible) exclude link quality measurement during timeslots that coincide with RREQ discovery flooding periods.

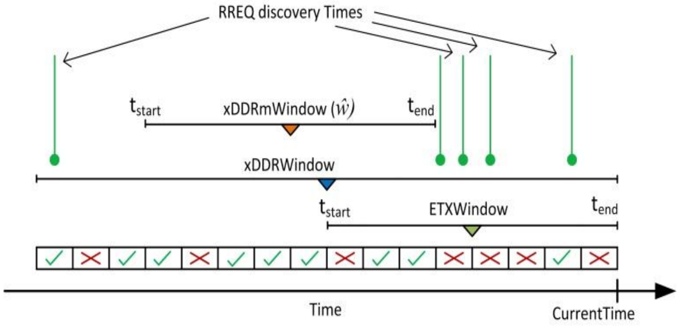

Figure 4: xDDR measurement window w.r.t ETXwindow

Figure 4 illustrates how instead of measuring ratio from the last slot as in ETX, xDDR derives $\hat{w}$ from xDDRWindow effectively excluding measurement during RREQ flooding phases. Given the xDDR measurement window ŵ and unicast sending interval ε, a node $\mathrm{n_O}$ calculates the directed delivery ratio $\vec{\lambda}(n_M, n_O, t')$ as:

$$\vec{\lambda}(n_M, n_O, t') = \frac{count(t' - \hat{\mathrm{w}}, t')}{\hat{\mathrm{w}}/\varepsilon}$$

$Count(t' - \hat{\mathrm{w}}, t')$ computes the number of proactive beacon packets received during the xDDR measurement window ŵ, while $\hat{\mathrm{w}}/\varepsilon$ is the total number of unicast packets sent. In Figure 5 we present the pseudo-code for calculating the link-level directed delivery ratio $\vec{\lambda}$.

Algorithm 2 xDDR window selection

```
T→ list of received packet timestamps in reverse chronological order
xDDRWindowSize := 2 × ETXWindowSize

if !ETXWindowConflict then
    $t_{end}$ := currentTime
else if ConflictFreeWindowAvailable
    $t_{end}$ := timeOfLastDiscovery
else
    $t_{end}$ := currentTime
endif
    $t_{start}$ := $t_{end}$ – (ETXWindowSize × ε)

for each t ∈ T
    if $t_{start} \le t \le t_{end}$
        count++
    end if
end for
```

5.2 Routing Module

We modified the DSR routing protocol to incorporate our link estimation metric. When a node $\mathrm{n_k}$ wants to transmit a stream $\mathrm{s(n_k, n_l, f, t_{st})}$ to a node $\mathrm{n_l}$, it initiates RREQ route discovery with a unique connectionID and floods the broadcast packets across the network. When a node receives a RREQ discovery of whom it's not the destination, it embeds its nodeID in the {nodelist} and updates the end-to-end delivery ratio estimate $\vec{\sigma}$ of the path by multiplying its delivery ratio estimate to the accumulated delivery ratio estimates of the previous hops. When the first RREQ packet τ_1 eventually reaches the destination, it waits for time t_{wait} to receive other contending route options. It then selects the route with the highest $\vec{\sigma}$ which is then reverse routed back to the source as unicast RREP message.

Algorithm 3 Directional Delivery Estimate and PDR

```
1:  Each node $n_Q$ selects a subset $E'_Q$ as unicast recipients
2: Compute link-level directional delivery ratio over $\hat{w}$ not including link performance during RREQ discovery phase:
        $\vec{\lambda}(n_Q, n_R, t') = \frac{count(t' - \hat{w}, t')}{\hat{w}/\varepsilon}$, $n_Q, n_R \in N$ & $Q \neq R$
2: Stream to send s($n_k$, $n_l$, $f$, $t_{st}$) with start time $t_{st}$ & attempts to transmit till the end of simulation at data rate $f$.
3: $\vec{\sigma}(\tau_x, t'') = \prod_{n_O, n_P \in \tau_x}(\vec{\lambda}(n_O, n_P, t_2))$, where $O \neq P$
    ($t'$ and $t''$ represent discrete timestamps)
4: Stream s($n_k$, $n_l$, $f$, $t_{st}$) selects a route $\tau_x$ if and only if
    $max_{\tau_U \in s^\tau}[\vec{\sigma}(\tau_U, t_2)] = \vec{\sigma}(\tau_x, t_2)$
```

6. Experimental Setup and Results

For our implementation and experimentation we used OMNeT++ [8], an extensible, modular, component-based C++ simulation framework, using the INETMANET [9] extension which is specifically dedicated to wireless and mobile ad hoc networks. Figure 6 shows related network parameters.

Parameter	Value	Parameter	Value
SimulationTime	300 s	t_{wait}	10 ms
Seeds/experiment	20	MAC	802.11g
Terrain	500m×500m	MAC bitrate	54 Mbps
MAC Retry Limit	1-6 (Figure 7)	Number of nodes	30 (Figure 7) 20-45 (Figure 8)
UDPSendInterval	0.005 s	xDDRWindowSize	200
UDPStartTime	11 s	ETXWindowSize	100
Trans AntennaGain	-1.4 dB	BeaconingInterval	0.1 sec
WLAN Radio Sensitivity	-90 dBm	Radio Transmit Power	1.0 mW

Figure 6: Simulation parameters

In Figure 7 we compare ETX and xDDR for PDR estimates with respect to actual packet delivery ratio. For experimental correctness we ran 20 trials for each of the six retransmission limits where each trial entails different placement for all the 30 nodes except the stream's sender and receiver (1 sender and 1 receiver). For each route discovery, the receiver node suggests a route with the highest cumulative PDR estimate $\vec{\sigma}$ to carry on the transmission. We compute average $Avg[\vec{\sigma}(\tau'_s)]$ for all the routes τ'_s employed during the course of each simulation. At the end of each simulation run, we compute the actual PDR yielded from selecting those routes. For each MAC retransmission limit, $Avg[\vec{\sigma}(\tau'_s)]$ and PDR are further averaged for the 20 trials and then compared for correlation. The y-axis represents the percentage difference between the route's end-to-end delivery ratio estimate vs the actual end-to-end delivery ratio yielded as a result of selecting those routes. Lower difference indicates better correlation and vice versa. We can see that as compared to ETX, xDDR has been able to estimate the end-to-end delivery ratios more accurately.

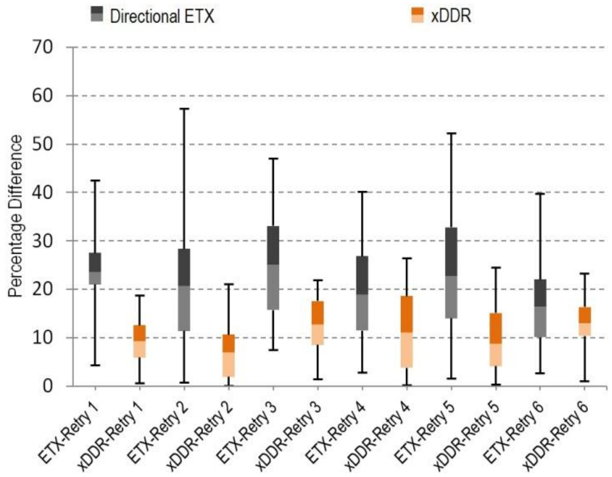

Figure 7: Difference in PDR Estimate and UDP Delivery Ratio

Furthermore in order to document the role of accurate estimates in achieving higher end-to-end delivery ratio, we ran experiments with high traffic comprising 5 senders and 5 receivers placed at the boundaries of a 500 m × 500 m terrain transmitting UDP packets at the rate of 200 CBR. MAC retry limit is 6. Figure 6 lists other relevant network parameters.

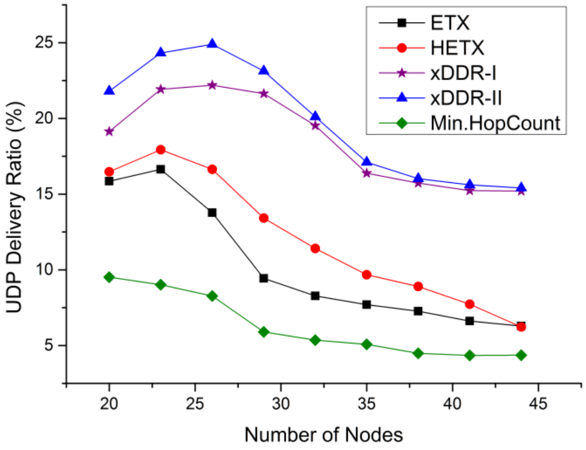

Figure 8: UDP Data Delivery Ratio comparison

Figure 8 shows the UDP data delivery ratio yielded as a result of route selection assisted by a particular link-level metric. We compare the end-to-end packet delivery ratio achieved by quality-driven route selection using DSR while employing minimum hop count, ETX, HETX and xDDR-I and xDDR-II as quality metrics. For the purpose of clarity, we classified xDDR in two versions. xDDR-I is link estimation mode that employs only the selective unicast beaconing (and same window size as ETXWindow), while xDDR-II uses selective unicast beaconing as well as xDDRWindow. For a given number of nodes we ran 20 trials of the experiment, each resulting in a different network topology of the intermediate nodes. While xDDR on average improves absolute packet delivery ratio percentage of 14%, 10% and 8% as compared to Minimum Hop Count, ETX and HETX respectively. This implies a 3 times gain over Minimum Hop Count, a 1.94 times gain over ETX and a 1.7 times gain over HETX in actual packet delivery.

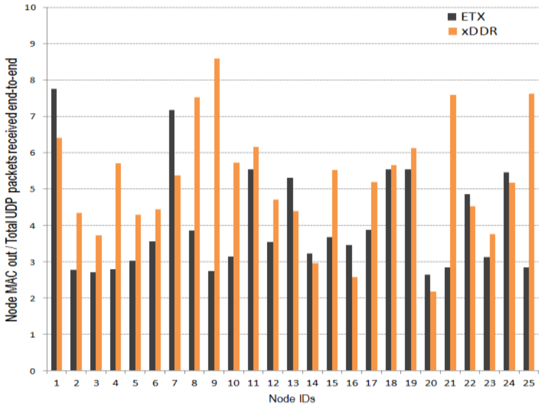

Figure 9: Ratio of packets transmitted by MAC layer of each node w.r.t. total UDP packets delivered

Picking a lossy route inaccurately as the best performing route in terms of PDR results in data being transmitted through links that may require more retransmissions per successful packet delivery than a healthier route. We observed the overall packets leaving each nodes MAC layer (MAC-out) and compared it to number of successful UDP packets received. Figure 9 represents the ratio of number of packets leaving MAC-out (includes routing discovery packets, ARP, xDDR or ETX packets, UDP packets etc.) with respect to total number of UDP packets received end-to-end in the network of 25 nodes. We believe this much overhead can be acceptable for applications seeking higher accuracy and end-to-end packet delivery ratio.

7. CONCLUSIONS

In this manuscript we proposed xDDR, a link quality directional packet delivery ratio estimation metric designed after exploring the behavior of ETX metric. xDDR employs selective unicasts to estimate directional delivery ratios of links and avoids the RREQ flooding phase problem by omitting the overlapping slots during link quality estimation. Doing so we are able to achieve improved accuracy in link-level delivery ratio estimation, which coupled with the quality aware routing module results in accurate end-to-end route estimates. We compared xDDR with minimum hop count, HETX [4] and ETX [7] and demonstrated the overall improvement in end-to-end packet delivery ratio. xDDR on average improves absolute packet delivery ratio percentage of 14%, 10% and 8% as compared to Minimum Hop Count, ETX and HETX respectively. This implies 3 times gain over Minimum Hop Count, 1.94 times gain over ETX and 1.7 times gain over HETX.

ACKNOWLEDGMENTS

This work was supported by the SenSafety project in the Dutch Commit program, www.sensafety.nl.

REFERENCES

[1] J. Zhao and R. Govindan. "Understanding packet delivery performance in dense wireless sensor networks". In the Proceedings of Proc. First ACM Sensys Conf., Nov. 2003.

[2] N. Baccour, A. Koubaa, L. Mottola, M. Zuniga, H. Youssef, C. Boano, and M. Alves. "Radio link quality estimation in wireless sensor networks: A survey". ACM Transaction on Sensor Networks, Sept. 2012.

[3] S.M.Das, H.Pucha, K.Papagiannaki, and Y.C.Hu, "Studying wireless routing link metric dynamics," in Proceedings of ACM SIGCOMMIMC '07, October 2007, pp .327–332.

[4] A. T.Tran; M.K.Kim, "Characteristics of ETX Link Quality Estimator Under High Traffic Load in Wireless Networks", HPCC_EUC 2013, pp.611,618, 13-15 Nov. 2013

[5] R. Draves, J. Padhye, and B. Zill. Comparison of routing metrics for static multi-hop wireless networks. InProc. ACM SIGCOMM Conference, September 2004.

[6] R. Riggio, N.Scalabrino, D.Miorandi, G.Iacovoni, "Wireless mesh networks: state of the art and research directions". Create-Net Technical Report N. 200600006. April 21, 2006.

[7] D. Couto, D. Aguayo, J. Bicket and R. Morris, "A high-throughput path metric for multi-hop wireless routing," Wireless Networks, vol. 11, no. 4, pp. 419–434, 2005.

[8] OMNeT++ Network Simulation Framework, http://www.omnetpp.org/ [accessed February 2015]

[9] INETMANET Extension for OMNET++, https://github.com/inetmanet/inetmanet [accessed February 2015]

[10] A. Adya, P. Bahl, J. Padhye, A. Wolman, & L. Zhou, "A multi-radio unification protocol for IEEE 802.11 wireless networks", BroadNets '04.

[11] S. Keshav. "A control-theoretic approach to flow control", in proceedings of SIGCOMM 1991. ACM Press, 1991.

[12] J .D. Padhye, R. P. Draves Jr., and B. D. Zill, "System and method for link quality routing using a weighted cumulative expected transmission time metric," US Patent 2009.

[13] R. Draves, J.Padhye, and B.Zill, "Routing in multi-radio, multi-hop wireless mesh networks", in the proceedings of MobiCom 2004, pp.114-128.

[14] D. Johnson and D. Maltz. Dynamic source routing in ad-hoc wireless networks. Proc. of SIGCOMM '96, Aug. 1996.

[15] D. De Couto, D. Aguayo, J. Bicket, and R. Morris, "A high-throughput path metric for multi-hop wireless routing," MobiCom '03, pp. 134-146, 2003.

[16] D. De Couto, "High-Throughput Routing for Multi-Hop Wireless Networks" PHD thesis, Massachusetts Institute of Technology, USA, 2004.

[17] D.L. Johnson and G.P. Hancke, "Comparison of two routing metrics in OLSR on a grid based mesh network," Ad hoc Networks, 2008, available online 4 May 2008.

[18] S. Waharte, B. Ishibashi, R. Boulaba, and D. Meddour, "Performance study of wireless mesh networks routing metrics," in in Proc. IEEE/ACS AICCSA 2008, 1100–1106.

[19] X. Ni, Performance evaluation of ETX on grid based wireless mesh networks, MPhil Thesis Report EET-UNSW. University of New South Wales, Australia, 2008

[20] P. D. Marco, C. Fischione, G. Athanasiou, and P.-V. Mekikis, "MACaware routing metrics for low power and lossy networks," in Proceedings of the IEEE INFOCOM Workshop, Apr. 2013, pp. 79–80

[21] A. Chan, S.-J. Lee, X. Cheng, S. Banerjee, and P. Mohapatra, "The impact of link-layer retransmissions on video streaming in wireless mesh networks," in Proceedings of Internation Wireless Internet Conference. ACM, 2008.