Extended Abstract: Revisiting Minimum Cost Reliable Routing in Wireless Mesh Networks*

Gentian Jakllari
Department of Computer Science
University of California
Riverside, CA, USA
jakllari@cs.ucr.edu

Stephan Eidenbenz
Los Alamos National Laboratory
Los Alamos, NM, USA
eidenben@lanl.gov

Nicolas Hengartner
Los Alamos National Laboratory
Los Alamos, NM, USA
nickh@lanl.gov

Srikanth Krishnamurthy
Department of Computer Science
University of California
Riverside, CA, USA
krish@cs.ucr.edu

Michalis Faloutsos
Department of Computer Science
University of California
Riverside, CA, USA
michalis@cs.ucr.edu

ABSTRACT

We revisit the problem of computing the path with the minimum cost in terms of the expected number of link layer retransmissions in wireless mesh networks. Unlike previous efforts (such as the popular ETX) we account for the fact that link layer protocols (such as the IEEE 802.11 MAC) incorporate a non-zero but finite number of retransmission attempts per packet. A key observation that motivates this work is that the performance of a path depends not only on the number of links on the path and their qualities, but also on the ***relative positions*** of the links on the path. In particular, the closer a lossy link to the destination, the higher is its impact on the performance of that path. We design a new path metric that captures all of the above factors and we call this metric ETOP. In this paper, we provide a synopsis of the analytical computation of ETOP. We also implement a routing strategy based on ETOP on a 25-node experimental testbed and provide sample results to showcase the performance with ETOP.

Categories and Subject Descriptors

C.2 [**Computer-Communication Networks**]: Wireless communication.

General Terms

Algorithms, Experimentation, Reliability.

*A full version of this paper is available as technical report at `www.cs.ucr.edu/~jakllari`. This work was partially supported by the NSF CAREER Grant 0237920

MobiCom'07, September 9–14, 2007, Montréal, Québec, Canada.

Keywords

Wireless Mesh Networks, Reliable Routing, Implementation.

1. INTRODUCTION AND BACKGROUND

In wireless networks, retransmissions are typically invoked at the link and transport layers to cope with high packet losses. Minimizing the cost incurred due to such retransmissions is critical for ensuring the efficient use of resources and a high overall throughput. De Couto et al [5], based on experimental data, argued that routing algorithms should consider the *quality* of the links on a path in addition to the number of hops, in order to reduce the end-to-end retransmissions cost. To this end, a new link metric was introduced to quantify the expected number of transmissions and retransmissions that are required to deliver a packet over a particular link. In particular, if p_s is the probability of a successful transmission over the particular link, the expected number of transmissions, ETX, to deliver a packet over that link is computed as [5]:

$$ETX = \sum_{j=1}^{\infty} j(1 - p_s)^j p_s = \frac{1}{p_s} \qquad (1)$$

The end-to-end cost of a path is then defined as the *sum* of the ETX values of the links on the path; the routing layer simply computes routes that minimize this cost. Other related efforts [3, 6, 7, 12, 14, 4] have used ETX in conjunction with other parameters (such as the link bandwidth [7]) for improving routing performance in multihop wireless networks.

A fundamental assumption underlying Equation (1) is that there are an infinite number of retransmission attempts per packet at the link layer. As per this assumption, the link layer never drops a packet and a transport layer protocol will never invoke retransmissions. In practice, however, a bounded number of retransmission attempts are made. Therefore, one has to account for the possibility that a packet may be *dropped* at the link layer. With a reliable transport protocol, a dropped packet at the link layer will trigger an end-to-end retransmission, starting at the source. Depending on which link the drop occurred, the cost associated with the new end-to-end retransmission will vary. A packet drop that is close to the destination is more expensive, since it induces retransmissions on all preceding links that were successfully traversed prior to the drop.

In this work, we revisit the problem of computing the path with the minimum cost in terms of the expected number of link layer transmissions (including retransmissions). The key observation that motivates our work is that in addition to the number of links on a path and the quality of those links (as considered by previous efforts), the end-to-end performance on a path depends on the *relative positions* of the links on the path. In a nutshell, it is better to encounter bad links at the beginning of a path, than later, in the proximity of the destination. We have constructed a path metric that accurately captures the impact of the relative ordering of the links (in addition to the impact of the number of links that make up their path and their individual link qualities) on the expected number of link layer transmissions. We call our metric the Expected number of Transmissions On a Path or **ETOP** for short. Our contributions are summarized below:

- We analytically compute the ETOP cost of a path. Note that since we consider a bounded number of retransmission attempts at the link layer, the path metric ETOP, cannot be computed as a simple sum of link level metrics.

- We implement ETOP as part of a routing protocol. We evaluate its performance on an indoor wireless mesh network consisting of 25 nodes and compare it with that of ETX-based routing [5]. We observe that ETOP-based routing computes paths that yield significantly higher TCP goodput (by as much as 65 %) compared to ETX-based routing.

In this *extended abstract* we provide a synopsis of our work and provide and discuss sample experimental results.

2. COMPUTING ETOP

In this section, we present a brief synopsis on the analytical computation of ETOP - the cost of a path in terms of the expected number of link layer transmissions - assuming that we know the delivery probability of each link. The key novelty is that we consider realistic assumptions for modeling the MAC and transport layer retransmissions. In more detail, if a packet transmission fails at the link layer, a finite number of retransmissions are attempted for each packet (as in IEEE 802.11). If all the retransmissions fail, the packet is *dropped*. The responsibility of reliably delivering data is then left to a transport protocol (such as TCP). At each instance of a packet drop at the link layer, the packet is "re-sent" again from the source, and we will refer to this as an **end-to-end attempt** or simply an **e2e attempt**.

Assumptions. In our model we make the following assumptions

a. The probability of a successful link transmission does not change between retransmission attempts. In other words, we assume that the probability of transmission failures on successive attempts on a link are independent and identically distributed (IID).

b. We assume that the power and bit-rate used for each transmission are fixed. If nodes are allowed to change their transmission properties, the probability of success will vary.

Network representation and notation: We model the wireless network as a directed graph $G(V, E, w)$, where V is the set of nodes and E the links. Every link $i \in E$ is assigned a weight $0 < p_i \leq 1$, which represents the packet delivery probability over that link with a single transmission attempt.

Consider the problem of sending a packet from node v_0, the source, to the node v_n, the destination, along a n-link path $(v_0, \ldots v_n)$. The source, node v_0, initiates an e2e attempt. First, the packet is passed on to the link layer, which will transmit it to node v_1.

If successfully received by node v_1, it will then be transmitted to node v_2, and so forth, until the packet reaches node v_n. There is a probability $0 < p_i \leq 1, i = 1, .., n$ that the packet, when transmitted by node v_{i-1}, will reach node v_i. If the packet transmitted by node v_{i-1} does not reach node v_i, it will be transmitted again by the link layer of node v_{i-1}. Up to K transmission attempts (including the initial attempt) are made, and the packet will be dropped if the K^{th} transmission fails to reach node v_i. The drop is reported to the transport layer of node v_0. In response, the transport layer of v_0 initiates a new e2e attempt for the same packet.

One or more e2e attempts may be needed to deliver the packet from node v_0 to node v_n. For every e2e attempt, there is a cost: the number of link level transmissions during this attempt. Let T_n be a random variable that represents the sum of the costs of all the e2e attempts made in order for a packet to be delivered from node v_0 to node v_n. *We compute the* expected *value of T_n as a function of link weights, p_i, and the bound on the number of link level transmissions, K.*

Let Y_n denote the random variable representing the number of e2e attempts required for the packet to be delivered to the destination on the n-hop path. Let M_ℓ denote the number of consecutive hops that are successfully traversed along the path, beginning at node v_0, in the ℓ^{th} e2e attempt. Thus, $M_\ell = 0$ if the packet fails to reach node v_1 from node v_0, and $M_\ell = n$ if the message has reached v_n. If $M_\ell < n$, the $(l+1)^{\text{st}}$ e2e attempt begins. We assume that the random variables $M_1, M_2, \ldots$ are independent and identically distributed (IID) and can be represented by a single random variable M. This implies that the effects experienced on the different e2e attempts are independent and identical. Let $H_{\ell,j}$ denote the number of link layer transmissions needed to deliver the packet from node v_j to node v_{j+1} in the ℓ^{th} e2e attempt. If the message has successfully traversed the link from v_j to v_{j+1}, $H_{\ell,j} \leq K$; else, if the message fails to reach node v_{j+1} from node v_j, then, $H_{\ell,j} = K$ and a new e2e attempt is started at node v_0. For each node v_j, we assume that $H_{1,j}, H_{2,j}, \ldots,$ are IID random variables and we use the notation H_j to represent this common random variable.

Using the model and the random variables defined above, we are able to compute ETOP as the expected cost in terms of the number of link layer transmissions T_n. Due to lack of space, we simply state the two main theorems that result from our analysis. The detailed analysis and the proofs of the theorems can be found in [9].

THEOREM 1. *The expected number of transmissions for delivering a packet over a path $(v_0, \ldots v_n)$, ETOP, is:*

$$\begin{aligned}\mathbb{E}[T_n] &= \left(K + \sum_{j=0}^{n-2} \left(\mathbb{E}[H_j | H_j \leq K]\, \mathbb{P}[M > j | M < n]\right)\right) \\ &\times \quad \mathbb{E}[Y_n - 1] + \sum_{j=0}^{n-1} \mathbb{E}[H_j | H_j \leq K]. \qquad (2)\end{aligned}$$

Our final goal is to use ETOP as part of a routing algorithm. Therefore we transform Equation (2) into a recursive function that lends itself to a greedy routing approach.

THEOREM 2. *The expected number of transmissions for delivering a packet over a path $(v_0, \ldots v_n)$, ETOP, can be written as a recursive function:*

$$\mathbb{E}[T_{n+1}] = \frac{1}{\pi_{n+1}} \mathbb{E}[T_n] + K \frac{1 - \pi_{n+1}}{\pi_{n+1}} + E_n \qquad (3)$$

Where $\pi_i, i = 1, .., n$ is the probability that the packet is not dropped on the link (v_{i-1}, v_i). In our model $\pi_i = 1 - (1 - p_i)^K$. In addition:

$$E_i = E[H_i | H_i \leq K] = \sum_{j=1}^{K} j \frac{(1-p)^{j-1} p}{1 - (1-p)^{K+1}}$$

3. EXPERIMENTAL EVALUATION OF ETOP

3.1 Implementation

In order to quantify the benefits of using ETOP, we implement a routing strategy based on the Equation (3) on a 25 node indoor wireless mesh network. In this section, we describe the network and provide details of our implementation.

Our Experimental Network Our testbed is an indoor wireless mesh network and consists of 25 Soekris net4826 nodes , deployed on one floor our building. We have equipped nodes with EMP-8602-6G 802.11a/b/g WiFi cards [1], which embed the Atheros AR5006 chipset; the cards are controlled by the latest Linux Mad-Wifi driver. Each card is connected to a 5-dBi gain, external omnidirectional antenna. We use the 802.11a mode in order to avoid interference from co-located 802.11b/g networks; our testbed is the only 802.11a network in the area.

Details of the Routing Implementation: We implement ETOP as part of a modified version of the dynamic source routing protocol (DSR) [13, 5] for the Linux kernel. We chose DSR because (i) it is one of the most popular protocols for multihop wireless networks and hence, its implementations are readily available and (ii) it allows a source to choose the path to the destination (as required by ETOP[1]). We consider the previously proposed metric ETX for comparison and use the implementation of the routing strategy based on ETX [5]. For ease of notation we refer to ETOP-based routing as ETOP-R and to ETX-based routing as ETX-R. We use the link-measurement component that is implemented as a separate element in Click [11]; it runs on every node and uses small broadcast packets to estimate the delivery probability from a node to each of its neighbors [5].

Implementation of ETX-R: The delivery probabilities computed by the link-measurement component are used to compute the ETX metric as described in Section 1. With ETX-R, when a node forwards a RREQ it includes not only its address, but also the ETX metric on the link to its predecessor node (the node from which it received the RREQ). This information is then reported back to the source through the RREPs. At the source, this information is passed on to the link-measurement component, which maintains a cache (the ETX link cache) of all the known nodes and the ETX metrics of their corresponding links. Whenever the source needs a route to a specific destination, it issues a request to the link-metric component. If the destination is in the cache, the link-metric component will return the route with the minimum ETX-weight, computed by running Dijkstra's weighted shortest path algorithm on the topology constructed with the nodes and links in the ETX link cache. Note that as in [5], the route error messages (RERR messages) that are induced by DSR [10] are disabled during the experiments. This is because this functionality of DSR is not utilized with either ETX-R or ETOP-R.

Implementation of ETOP-R To implement our ETOP-based algorithm, we made the following changes to the link-measurement component. First, we build a new cache for ETOP that is similar to the ETX link cache, except that the links are now represented by their delivery probabilities.

[1]ETOP is non-commutative.

The ETOP cache is not populated by collecting data via an explicit new mechanism, but is derived from the information in the ETX link cache. We exploit the simple relation between ETX and the link delivery probability given by Equation (1). In computing ETX, the authors assume that the probes compute the probability of successfully delivering a packet across a link; ETX is computed to be the inverse of this probability. With ETOP, the probability of successfully delivering a packet across a link is given by π_i, the probability that a packet is "not" dropped on a given link. Thus, in the above discussion, we equate π_i to $1/ETX$. This value is then used in computing the ETOP using Equation (3). Finally, Dijkstra's weighted shortest path was modified in order to return paths that minimize the ETOP metric. The modification consists in replacing the cost function used by ETX - the sum of ETX values of the links - with Equation (3). When the source needs a route to the destination it issues a request to the link-measurement component along with an option to indicate whether the Dijkstra's algorithm with ETX, or ETOP is to be used.

3.2 Impact of ETOP-R and ETX-R on long lived TCP Flows

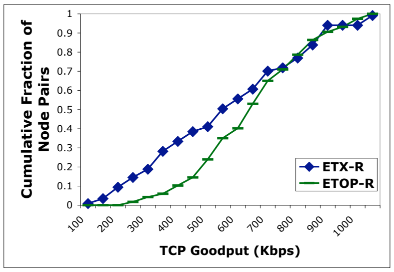
Given that ETOP-R is especially designed for use when a reliable transport layer is used in conjunction with a link layer scheme that supports a limited number of retransmissions, we consider the most popular choices that have these properties viz., TCP and the IEEE 802.11 MAC, respectively.

The Experiment Set up: For our experiment, we choose at random a large number of source-destination pairs, 110, out of the possible $25 \times 24 = 600$ combinations and run TCP sessions on each pair for 3 minutes. We use "Iperf" [2] to measure the maximum achievable TCP bandwidth (goodput).

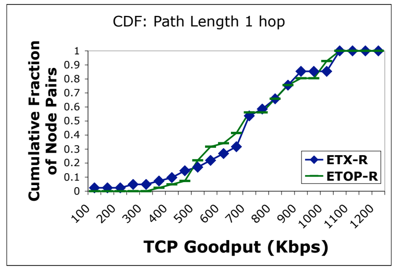
To make the results between the metrics as comparable as possible, the following setup (similar to that in [5]) is used. For each of the 110 node pairs, we run ETX-R immediately followed by ETOP-R. Thus, the results with the two metrics are obtained within minutes of each other; we expect the channel conditions to have changed little during this time. We repeat the experiment six times and compute an average to reduce the impact of temporal variations. On every path, the protocols are allowed to run for 90 secs to achieve stable operations. Then, the source pings the destination for 5 sec, at a rate of one packet per second, in order to allow the protocols to discover the paths to the destination. Subsequently, the source initiates a TCP connection with the destination. Furthermore, as in [5], prior to starting the connection, the source waits for 15 secs in order to allow enough time for the routing component to collect the link statistics.

ETOP-R improves TCP goodput over ETX-R: In Figure 1 the CDF and the medians[2] of the distribution of the measured TCP goodputs for ETOP-R and ETX-R are depicted. The CDF for all the 110 node-pairs, depicted in Figure 1(a), shows that ETOP-R performs better than ETX-R, by as much as 65%, in a wide range of goodputs. A more detailed look into the data, Figures 1(b)-1(e), reveals that the regime of goodput values where ETOP-R offers significant improvements correspond to those achieved by node-pairs separated by three or more hops; the regime of goodput values where the statistical performance of ETOP-R and ETX-R are similar, correspond to those achieved by node-pairs one or two hops away. This is expected, since, for the node pairs that are separated by one or two hops the position of the link has little or no impact and thus, ETOP-R can offer little or no improvements.

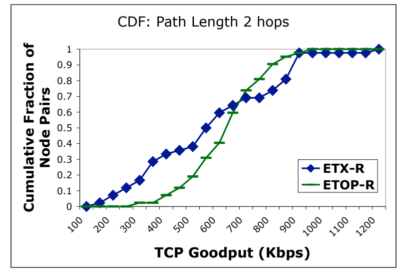
[2]When the distribution of the data is skewed (as it is in our case), the median is more representative of a typical observation than the mean[8].



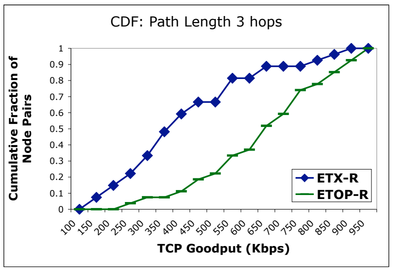
(a) CDF of the TCP goodput for all 110 paths.



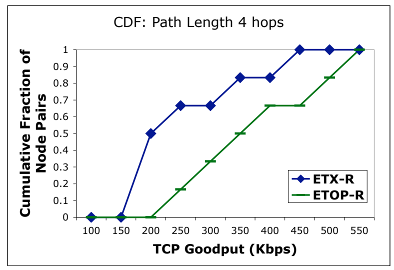
(b) CDF of the TCP goodput for node pairs that are 1-hop apart



(c) CDF of the TCP goodput for node pairs that are 2-hops apart



(d) CDF of the TCP goodput for node pairs that are 3-hops apart



(e) CDF of the TCP goodput for node pairs that are 4-hops apart

Path Length	# Pairs	Median TCP Goodput (Kbps)	
		ETX-R	ETOP-R
All Lengths	110	523.1	640.4
1-hop	39	681.7	665.8
2-hops	41	546.5	632.7
3-hops	23	382.3	648.2
4-hops	7	272.8	422.3

(f) Median TCP goodput for all path lengths.

Figure 1: The CDF of the goodputs distribution for the 110 node pairs, Figure 1(a); organized by path length in Figures 1(b),1(c),1(d),1(e). The median goodput for all path lengths is depicted in Figure 1(f). ETOP-R offers significant improvement, by as much as 65%, over ETX-R for the node pairs separated by 3 or more hops.

4. CONCLUSIONS

In this work, we revisit the problem of computing the path with the minimum cost in terms of the number of link layer retransmissions in multi-hop wireless networks. The key feature that distinguishes our work is that we consider a finite number of link level retransmissions unlike previous efforts (such as ETX); this makes the problem significantly more involved. We demonstrate that in addition to the magnitude of the link reliabilities on a path, the relative ordering of the links is critical in computing the correct minimum cost path. We provide a synopsis of an analytical model to compute a path metric ETOP, that captures this cost. We implement ETOP-based routing and perform sample experiments on a 25 node indoor mesh network to quantify and evaluate its performance. We compare the performance of the paths computed with our metric with those computed with a routing strategy based on using ETX. We provide sample results that show that our scheme outperforms the ETX by as much as 65 % in terms of achievable TCP goodput.

5. REFERENCES

[1] Emp-8602 6g wireless card. http://www.netgate.com.

[2] Iperf-tool.http://dast.nlanr.net/projects/iperf/.

[3] S. Banerjee and A. Misra. Minimum energy paths for reliable communication in multi-hop wireless networks. In *ACM MobiHoc 2002*.

[4] Can Emre Koksal and Hari Balakrishnan. Quality-Aware Routing Metrics for Time-Varying Wireless Mesh Networks. *IEEE JSAC*, 24(11), November 2006.

[5] D. S. J. De Couto, D. Aguayo, J. Bicket, and R. Morris. A high-throughput path metric for multi-hop wireless routing. In *ACM MobiCom 2003*.

[6] Qunfeng Dong, Suman Banerjee, Micah Adler, and Archan Misra. Minimum energy reliable paths using unreliable wireless links. In *ACM Mobihoc 2005*.

[7] R. Draves, J. Padhye, and B. Zill. Routing in multi-radio, multi-hop wireless mesh networks. In *ACM MobiCom 2004*.

[8] R. Jain. *The Art of Computer Systems Performance Analysis*. John Wiley and Sons, Inc., 1991.

[9] G. Jakllari, S. Eidenbenz, N. Hengartner, S. Krishnamurthy, and M. Faloutsos. Revisiting minimum cost reliable routing in wireless mesh networks. Technical Report, University of California, Riverside, 2006.

[10] David B. Johnson, David A. Maltz, and Yih-Chun Hu. *The Dynamic Source Routing Protocol for Mobile Ad Hoc Networks (DSR). Internet draft (work in progress)*. IETF, April 2003.

[11] Eddie Kohler, Robert Morris, Benjie Chen, John Jannotti, and M. Frans Kaashoek. The click modular router. *ACM Trans. Comput. Syst.*, 18(3):263–297, 2000.

[12] Karim Seada, Marco Zuniga, Ahmed Helmy, and Bhaskar Krishnamachari. Energy-efficient forwarding strategies for geographic routing in lossy wireless sensor networks. In *ACM SenSys '04*.

[13] Audun Tornquist. Modular and adaptive ad hoc routing in click. MS thesis, University of Colorado, 2001.

[14] Alec Woo, Terence Tong, and David Culler. Taming the underlying challenges of reliable multihop routing in sensor networks. In *ACM SenSys '03*.