An Integrated Routing and Rate Adaptation Framework for Multi-rate Multi-hop Wireless Networks

Tae-Suk Kim, Gentian Jakllari, Srikanth V. Krishnamurthy, Michalis Faloutsos

An integrated routing and rate adaptation framework for multi-rate multi-hop wireless networks

Tae-Suk Kim · Gentian Jakllari · Srikanth V. Krishnamurthy · Michalis Faloutsos

Abstract In this paper, we propose a new integrated framework for *joint* routing and rate adaptation in multi-rate multi-hop wireless networks. Unlike many previous efforts, our framework considers several factors that affect end-to-end performance. Among these factors, the framework takes into account the effect of the relative positions of the links on a path when choosing the rates of operation and the importance of avoiding congested areas. The key element of our framework is a new comprehensive path metric that we call ETM (for expected transmission cost in multi-rate wireless networks). We analytically derive the ETM metric. We show that the ETM metric can be used to determine the best end-to-end path with a greedy routing approach. We also show that the metric can be used to dynamically select the best transmission rate for each link on the path via a dynamic programming approach. We implement the ETM-framework on an indoor wireless mesh network and compare its performance with that of frameworks based on the popular ETT and the recently proposed ETOP metrics. Our experiments demonstrate that the ETM-framework can yield throughput improvements of up to 253 and 368 % as compared with the ETT and ETOP frameworks.

Keywords Wireless mesh networking · Routing protocols · Experimentation

1 Introduction

The goal of this work is to maximize the end-to-end throughput of flows over a multi-rate multi-hop wireless network. The motivating observation is that the end-to-end throughput depends on a large number of factors that need be considered *jointly* during the selection and management of a route. Despite the significant amount of research in this area, we find that no previous work addresses together the issues of: (a) the selection of a path, (b) link rate adaptation on a per path-basis (and not per link in isolation), and (c) the effect of competing flows in the network. We discuss each of these issues below along with some previous work, which we revisit in more detail in Sect. 2.

1.1 Path selection

The routing metric should *capture* the influence of all of the factors that affect the end-to-end throughput. The following *interdependent* factors dictate the achievable throughput on an end-to-end path: (a) the number of links on the path, (b) the achievable rates on links (c) the quality of the links for the each of the possible rates and, (d) the relative positions of the links on the path. The last factor is a direct consequence of the typical use of a finite number of

T.-S. Kim
Samsung Advanced Institute of Technology (SAIT),
Samsung Electronics, Yongin, South Korea
e-mail: taesuk.kim@samsung.com

G. Jakllari (✉)
IRIT-ENSEEIHT, University of Toulouse, Toulouse, France
e-mail: jakllari@enseeiht.fr

S. V. Krishnamurthy · M. Faloutsos
Department of Computer Science and Engineering,
University of California, Riverside, Riverside, CA, USA
e-mail: krish@cs.ucr.edu

M. Faloutsos
e-mail: michalis@cs.ucr.edu

retransmission attempts at the link layer (as with 802.11) but as many transmissions as needed to deliver the packet with a reliable transport layer protocol (such as TCP) [1]; packet drops closer to the destination induce end-to-end (e2e) retransmissions on links that were successfully traversed on previous e2e attempts and this increases the load and thereby congestion. Previously proposed routing metrics (such as [1–3]) account for a sub-set of the above factors; to the best of our knowledge, *there is no routing metric that captures the impact of all of these factors*.

1.2 Rate adaptation

Link rates need to adapt to the changing channel conditions during the life of a routing path. Most current link-rate adaptation schemes [4–12] operate on a *per link* basis and do not consider end-to-end effects. In other words, these mechanisms do not take into account the position of the link along the path. As pointed out in [1], link layer packet drops close to the destination are expensive in terms of retransmission costs: the packet has to be retransmitted from the source assuming a reliable transport protocol. This suggests that we want the reliability of the links to increase as we approach the destination and one way to achieve this is to use *lower transmission rates* on these links. At the same time, simply reducing the rates could have a negative impact on the throughput. Route selection and rate adaptation will have to work hand in hand to consider the above effect; the best possible routes will have to be chosen and the rates that yield the best throughputs will have to be selected for each link on the chosen route.

1.3 Interaction with other flows

Preferring reliable or higher-quality links is beneficial for a flow but has an indirect disadvantage: it can create congestion. This congestion can decrease or even reverse the gains due to the reduced retransmission costs. We also consider this effect when we design our framework. Avoiding congested links has been considered previously [13, 14], but not jointly with path selection and rate adaptation.

In this paper, we propose a new integrated framework for *joint* routing and rate adaptation in multi-rate multi-hop wireless networks. Our framework takes into account all of the above mentioned factors. Our key novelty is a new metric that we call ETM (for expected transmission cost in multi-rate wireless networks), which effectively captures the expected time for a packet to be delivered successfully over a given path *considering multi-rate capabilities, the bounded number of link layer retransmissions, back-off times and queuing delays*. ETM is arguably the most comprehensive routing metric to date. From an end-to-end performance point of view, it enables us to: (a) find reliable high-quality paths, (b) identify the appropriate rate for each link, and (c) avoid congested areas.

The contributions of our work are as follows:

Computing the ETM metric: We analytically derive the ETM cost of a path.

Using the ETM metric: We solve the problem of finding the path with the minimum ETM cost (we refer to this as the Optimal Path Problem or OPP) using a greedy approach, and we show that the greedy solution is indeed optimal. We solve the problem of choosing the right rates on the minimum ETM path using a dynamic programming approach (we refer to this as the Optimal Rate Problem or ORP).

Implementing ETM: As a proof of feasibility, we implement the ETM framework by using the Roofnet module in the popular Click toolkit [15]. For a comparative evaluation, we also implement the ETT-and ETOP-frameworks [1, 3] and evaluate all these protocols on an indoor wireless mesh network consisting of 21 nodes. We primarily consider 802.11 as the link layer protocol and TCP as the e2e protocol, given that these are readily available and are likely to be used in today’s deployments.

Evaluating ETM: The ETM-framework yields higher TCP throughputs compared to the ETT-and the ETOP-frameworks. First, considering TCP flows in isolation, the median throughput with the ETM-framework improves by 131 and 30 % over that achieved with the ETOP-and ETT-frameworks. Second, considering multiple flows, we observe that the ETM metric manages to load-balance traffic successfully. The aggregate throughput improvements with ETM are 253 and 368 %, as compared with ETOP and ETT, respectively.

The remainder of the paper is organized as follows. Related work is described in Sect. 2 In Sect. 3, we analytically compute ETM. We formulate the Optimal Path and the Optimal Rate Problems and propose solutions for each in Sect. 4 We evaluate the ETM-framework in Sect. 5 We conclude the paper in Sect. 6

2 Related work

In this section, we first review the relevant related routing metrics. We then discuss related work on rate control.

2.1 Link quality based routing metrics

There are several previously proposed link quality based routing metrics for multi-hop wireless networks.

2.1.1 *The ETX metric*

Proposed by De Couto et al. [2], ETX computes the expected number of transmissions (including retransmissions) needed to send a packet over a link, by measuring the forward and reverse packet delivery ratios (PDR) between a pair of neighboring nodes. With ETX as the routing metric, the routing protocol finds routes with the least expected number of transmissions; however, when computing ETX, it is implicitly assumed that an infinite number of retransmissions are possible at the link layer. Furthermore, ETX does not consider the use of multiple transmission rates. Measurements on wireless testbeds [2, 16] show that the use of ETX results in routes that yield higher throughputs than with minimum hop count based routing.

2.1.2 *The ETT metric*

ETX [2] cannot identify high-throughput routes in multi-rate wireless networks since it only considers the packet delivery ratios (PDRs) at the basic rate on each link. ETT has thus been proposed in [3] for multi-rate wireless networks. The ETT metric for a given link is defined to be the expected time to send a 1,500-byte packet at the rate that yields the highest throughput on that link. ETT also accounts for the time taken for retransmissions (determined by the PDR) at each rate. The ETT cost of a route is the sum of the ETTs of each link on the route. ETT implicitly assumes (as with ETX) an infinite number of retransmissions on each link.

2.1.3 *The ETOP metric*

Unlike ETX and ETT, the ETOP metric proposed in [1] captures the impact of a finite number of retransmission attempts at the link layer. The authors identify that packet drops closer to the destination can be expensive due to this effect. However, one might expect that the ETOP metric will not work well in multi-rate wireless networks for the following reasons. **First**, ETOP estimates the link quality by using probes that are broadcated at the basic rate. This results in choosing paths that may not support high rates. **Second**, ETOP uses *small* probe packets and this does not accurately reflect the loss rates for the larger data packets; this can lead to the selection of *lossy* links from the perspective of data packets (even close to the destination). **Third**, the route produced by ETOP does not account for dynamic changes in link quality *in between* route changes. Thus, in spite of the fact that the chosen routes account for link positions, unforeseen retransmissions could result with temporal variations in link quality due to the improper choice of transmission rates by underlying rate adaptation mechanism (these mechanisms do not account for link positions). If such packet drops occur close to the destination, they will induce costly e2e retransmissions. It is critical that the right rate be chosen in order to prevent this effect. As later seen in Sect. 5 the performance of ETOP, observed on our wireless testbed in multi-rate settings is inferior to that of ETT; while the latter does not account for position of links on a route, it does account for the use of multiple transmission rates. Thus, it is evident that the choice of rate has a significant influence on the achieved performance. In our work here, we not only make route choices while accounting for the transmission rate, we also consider fine grained channel fluctuations which affect the choice of rate (ETOP does not consider fine grained channel fluctuations). Furthermore, unlike ETOP, we also capture the impact of network congestion in our metric to spatially separate flows.

2.1.4 *Other related efforts*

In [3], Draves et al. propose a new routing metric, WCETT to take the intra-flow interference into account. In [17] the authors propose a routing metric for selfishly behaving nodes. In [18] a new routing metric is introduced and analyzed. However, it requires every node to be equipped with multiple transceivers. In [19], Razak et al. introduce a routing metric that takes into account what they refer to as MAC interactions. However, an off-line and centralized computation is employed to quantify these MAC interactions. Koksal et al. [20] propose mETX and ENT; these metrics extend ETX to account for highly variable link reliabilities. The effect of short-term channel variation is accounted for in metric computation by utilizing both the average and standard deviation of the observed channel loss rates. Other efforts that attempt to reduce energy consumption due to retransmission costs in a mesh network setting include [21, 22]. The problem of load balancing in mesh networks has been studied in [13, 14].

Unlike our work, none of the above methods account for the finite number of retransmissions at the link layer, multi-rate capabilities, and the impact of queuing delay together.

2.2 Rate control

Rate control mechanisms that adapt to link quality variations are proposed in [5–7]. Qiao et al. [7] propose an approach which controls the sender’s rate dynamically to improve responsiveness to channel variations. Choi et al. [6] consider collision effects on rate control. There are several proposals that utilize the RTS/CTS exchange for rate control purposes [8–10]. In particular, the RTS and CTS messages are used to determine the quality of the channel (via signal-to-noise or SNR assessments). Some of the rate control mechanisms have been implemented on off-the-self network interface cards (NICs) and have been widely used. The Auto Rate Fallback (ARF) protocol [11] is the first commercial implementation that exploits the multi-rate capability of an IEEE 802.11 network. Implemented on MIT's Roofnet [4], SampleRate selects the data rate that has the smallest average per-packet transmission time in order to achieve high throughput. Onoe [12] is implemented in the 802.11 device driver for Atheros cards in Linux and Free BSD; it tries to find the highest data rate that suffers from less than 50 % loss rate.

The aforementioned approaches operate on a per-link basis and do not account for the impact of position of the link on the rate. As discussed earlier, since it is desired that links closer to the destination be more reliable, the rate selection on the links of a path will have to be correlated. In our work, we meet this requirement by jointly considering rate adaptation with routing.

3 Expected transmission cost in multi-rate wireless networks (ETM)

In this section, we analytically compute an expression for the ETM cost of a path. We begin with describing the backgorund of deriving the new metric ETM. The model under consideration and the metric of interest are derived subsequently. For the purposes of our analysis, we summarize the notation used in Table 1.

3.1 Background

Figure 1 shows the background of the ETM metric. For the purpose of maximizing the end-to-end throughput of flows over the multi-rate multi-hop wireless network, two critical factors to end-to-end throughput are investigated: the effect of rate adaptation and the flow congestion. The rate adaptation affects the end-to-end throughput but, most current rate adaptation schemes operate on a per link basis and do not consider the end-to-end effect. In multihop wireless networks, link layer packet drops close to the destination are more expensive in terms of retransmission costs than those close to the source of the packet [1]. Therefore, reliable links should be more desirable as a packet approaches the destination. Nevertheless, while preferring reliable or higher-quality links is beneficial for a particular flow, it has an indirect disadvantage. It can increase congestion as more flows will prefer the same reliable links. The added congestion can decrease or even reverse the gains due to the reduced retransmission costs. Therefore, it is necessary to have a routing metric that considers all of these factors. In this paper we propose such a metric, the ETM.

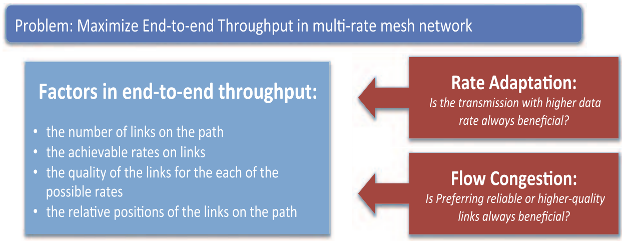

Problem: Maximize End-to-end Throughput in multi-rate mesh network
Factors in end-to-end throughput:
• the number of links on the path
• the achievable rates on links
• the quality of the links for the each of the possible rates
• the relative positions of the links on the path
Rate Adaptation:
Is the transmission with higher data rate always beneficial?
Flow Congestion:
Is Preferring reliable or higher-quality links always beneficial?

Fig. 1 ETM Background: its main goal and the factors that are considered

3.2 System model

We assume that each link can support R transmission rates. The number of possible transmission rates depends on the

Table 1 Notation used in analysis

Notation	Definition
R	The set of possible rates/the cardinality of the set
p_r	The packet transmission success probability with using rate r
K	The maximum number of transmission attempts at the link layer
z_n	The number of e2e transmissions required for the packet to be delivered to the destination on an n-link path $(v_0, \cdots, v_n)$
h	The number of consecutive hops that are successfully traversed along a path, beginning at the source node
l_{r_j}	The number of link layer transmissions needed to deliver the packet over link j with data rate r_j
T_r	The transmission time for a packet with data rate r
$B(k)$	The sum of the expected times spent in back-off over the k retransmission attempts
q_j	The queuing delay experienced by a packet at node j
L_{r_j}	The expected value defined as $\mathbf{E}\left[l_{r_j} \vert l_{r_j} \leq K\right]$
Q_j	The expected value for q_j
π_{r_j}	The probability that a packet transmitted with data rate r_j is not dropped on link j
ρ_i	The probability that a packet transmitted with rates $(r_1, \cdots, r_i)$ is delivered over i successive hops

physical layer of the system under consideration. For example, IEEE 802.11a supports a total of eight transmission rates. We interchangeably use R to denote both the set of possible rates and the cardinality of the set. For each rate r ($\in R$) there is an associated probability p_r of a packet transmission success. At the link layer the number of transmission attempts is limited to K; if a packet fails in K successive attempts, the link layer drops the packet. We assume that such a packet drop on a link induces a transport layer retransmission (with a protocol like TCP) from the source. Consider the traversal of a packet over a path (v_0, ..., v_n) consisting of $n + 1$ nodes (correspondingly, n links). For notational convenience, we label the link between node $v_{i\text{-}1}$ and node v_i as link i. Let the transmission rate used on the link between nodes $v_{i\text{-}1}$ and v_i be r_i and the corresponding probability of successfully delivering the packet across the link in a single attempt be p_{r_i}. First, the source node v_0 initiates an end-to-end (e2e) connection. It then sends packets to its link layer. A transmission is then performed by selecting a rate r_1 out of the R available rates for link 1. If the packet is received successfully within K transmission attempts by node v_1, then v_1 initiates a forwarding of the packet to node v_2, and so forth. However, if all K transmission attempts on any link, say link i, fail, then the packet will be dropped by the link layer of node $v_{i\text{-}1}$. The packet drop causes the transport layer of the source node v_0 to initiate an e2e retransmission. In such a case, the failed e2e transmission cost adds to the time consumed for delivering the packet from v_0 to v_n. In addition, each node is assumed to uses a FIFO queue for all the outbound packets; however, control packets have a priority over data packets. Our interest then, is in answering the following question: *given a path* (v_0, ..., v_n), *the corresponding rate set* (r_1, ..., r_n) *and the associated delivery probability set* $(p_{r_1}, \cdots, p_{r_n})$, *what is the expected transmission time required for a packet to be successfully delivered end to end (from* v_0 *to* v_n)? Towards answering this, we derive our proposed metric, ETM, which provides an estimate of the above expected transmission time.

4 Computing ETM

Let z_n denote the random variable representing the number of e2e transmissions required in order for the packet to be delivered to the destination on an n-link path (v_0, ..., v_n). Let h_i denote the number of consecutive hops that are successfully traversed along the path, beginning at node v_0, in the i^{th} e2e transmission attempt. Then, we have $h_i = 0$ if the packet fails to reach node v_1 from node v_0, and $h_i = n$ if the packet reaches the destination, v_n. In particular, $h_i < n$ indicates that the $(i + 1)^{th}$ e2e retransmission is attempted. We assume that the random variables h_1, h_2, ... are independent and identically distributed (**i.i.d**); this implies that the effects experienced on the different e2e transmission attempts are independent and identical. Since only the short-term fading [23] can be expected to affect the e2e retransmissions, this assumption is reasonable. With this, we represent the variables by a single random variable h. Let l_{i,r_j} denote the number of link layer transmissions needed to deliver the packet over link j with data rate r_j in the *ith* e2e transmission attempt. If the packet has successfully traversed link j, we have $l_{i,r_j} \leq K$. Otherwise, $l_{i,r_j} = K$ and a new e2e transmission attempt is started at node v_0. For each node v_j, we assume that $l_{1,r_j}, l_{2,r_j}, \cdots$ are *i.i.d* random variables, and the notation l_{r_j} is used to represent this common random variable.

In calculating the time taken for a packet to be delivered over a link, we account for the following three components: (1) the time for transmitting a packet, (2) the time for which the node backs-off between transmission attempts, and (3) the average queuing delay experienced by a packet at the transmitter node.

For the calculation of (i), we assume that the transmission of a packet at rate r takes T_r seconds [24]. For the calculation of (ii), let $B(k)$ be the sum of the expected times spent in back-off over the k retransmission attempts, including the random period prior to the initial transmission attempt. The 802.11 MAC randomly selects the back-off window CW_i for the i^{th} retransmission attempt of a packet from the interval (0, $2^{i\text{-}1}CW_{min}$), where CW_{min} is the minimum back-off window size. The interval increases to (0, CW_{max}) with the attempts, and after that, there is no longer an increase; CW_{max} is the maximum back-off window size. We assume that $CW_{max} = 2^6\, CW_{min}$ conforming to the IEEE 802.11 standard [25]. Then, as derived in [3], we can express $B(k)$ as

$$B(k) = T_{slot} \sum_{i=1}^{k} \mathbf{E}[CW_i] = \begin{cases} \frac{CW_{min}}{2}(2^k - 1)T_{slot}, & k \leq 7 \\ \frac{CW_{min}}{2}\{63 + 64(k-7)\}T_{slot}, & k \geq 8, \end{cases} \tag{1}$$

where T_{slot} denotes the size of each slot in time units. Unfortunately, the calculation of (iii) is not straightforward; the variation in queuing delay depends on the network dynamics (e.g., the traffic load and the channel capacity) as well as the underlying MAC protocol. Statistical models [26, 27] have been applied to analyze the queuing behavior in a wireless network using the 802.11 MAC protocol. In this work, we simply assume that measured values of the average queuing delays are available[1].

[1] We discuss how this is measured in our implementation later.

Let the queuing delay experienced by a packet at node j be q_j. Then, if the packet is delivered over link j after l_{r_j} transmission attempts, the total time consumed on that link is given by $l_{r_j}T_{r_j} + B(l_{r_j}) + q_j$.

For the general case of a n-link path, the total cost $\mathbf{C}(\mathbb{R}_n)$ for successfully transmitting a packet with rates $\mathbb{R}_n = (r_1, r_2, \ldots, r_n)$ over the path, consists of the cost due to $z_n - 1$ unsuccessful transport layer transmission attempts and the cost incurred with the one last successful transmission, and is formally given by

$$\mathbf{C}(\mathbb{R}_n) = \sum_{i=1}^{z_n-1}\left\{\sum_{j=1}^{h_i}\left(l_{r_j}T_{r_j} + B(l_{r_j}) + q_j\right) + KT_{r_{h_i+1}} + B(K) + q_{h_i+1}\right\} + \sum_{j=1}^{h_{z_n}}\left(l_{r_j}T_{r_j} + B(l_{r_j}) + q_j\right). \tag{2}$$

Note that with z_n set to 1, Eq. (2) can model the transmission cost of traffic with no end-to-end retransmissions (such as UDP traffic). From Eq. (2), we derive the ETM cost in the following proposition.

Proposition 1 *The expected transmission cost, -, for delivering a packet over path* $(v_0, \ldots, v_n)$ *with rates* $\mathbb{R}_n = (r_1, \ldots, r_n)$ *is*

$$\begin{aligned}\mathbf{E}[\mathbf{C}(\mathbb{R}_n)] &= \mathbf{E}[z_n - 1]\left\{\sum_{j=1}^{n-1}\left(L_{r_j}T_{r_j} + B(l_{r_j}) + Q_j\right)\mathbf{P}[h > j-1|h<n]\right. \\ &\quad \left. + \sum_{j=1}^{n}\left(KT_{r_j} + B(K) + Q_j\right)\mathbf{P}[h = j-1|h<n]\right\} \\ &\quad + \sum_{j=1}^{n}\left(L_{r_j}T_{r_j} + B(l_{r_j}) + Q_j\right),\end{aligned} \tag{3}$$

where, $L_{r_j} = \mathbf{E}\left[l_{r_j}|l_{r_j} \le K\right]$ *and* $Q_j = \mathbf{E}[q_j]$.

Proof By taking the expectation on both sides of Eq. (2), and replacing the term within the first summation on the right-hand side of Eq. (2) with Λ_i, we have

$$\mathbf{E}[\mathbf{C}(\mathbb{R}_n)] = \mathbf{E}\left[\sum_{i=1}^{z_n-1}\Lambda_i\right] + \mathbf{E}\left[\sum_{j=1}^{h_{z_n}}\left(l_{r_j}T_{r_j} + B(l_{r_j}) + q_j\right)\right]. \tag{4}$$

We derive the two terms of Eq. (4) separately for clarity. From the fact that z_n is independent of Λ_i (due to the assumption that each e2e attempt experiences i.i.d losses), $\mathbf{E}\left[\sum_{i=1}^{z_n-1}\Lambda_i\right]$ reduces to $\mathbf{E}[z_n - 1]\mathbf{E}[\Lambda]$.

Omitting the index relating to the e2e transmission attempt i (since these attempts are i.i.d):

$$\mathbf{E}[\Lambda] = \mathbf{E}\left[\sum_{j=1}^{h}\left(l_{r_j}T_{r_j} + B(l_{r_j}) + q_j\right) + KT_{r_{h+1}} + B(K) + q_{h+1}\right]. \tag{5}$$

Conditioning on h (the number of links traversed during an unsuccessful attempt) in Eq. (5), we have

$$\begin{aligned}\mathbf{E}[\Lambda] &= \sum_{i=0}^{n-1}\left\{\sum_{j=1}^{i}\left(\mathbf{E}\left[l_{r_j}|l_{r_j} \le K\right]T_{r_j}\right.\right. \\ &\quad \left.\left. + \mathbf{E}\left[B\left(\mathbf{E}\left[l_{r_j}|l_{r_j} \le K\right]\right)\right] + \mathbf{E}\left[q_j\right]\right)\right\}\mathbf{P}[h = i|h<n] \\ &\quad + \sum_{i=0}^{n-1}\left(KT_{r_{i+1}} + B(K) + \mathbf{E}[q_{i+1}]\right)\mathbf{P}[h = i|h<n].\end{aligned} \tag{6}$$

By expanding the first term on the right-hand side of Eq. (6), and using the relation $\sum_{j=i}^{n\text{-}1}\mathbf{P}[h = j|h < n] = \mathbf{P}[h > i - 1|h < n]$, we have

$$\begin{aligned}\mathbf{E}[\Lambda] &= \sum_{i=1}^{n-1}\{\mathbf{E}[l_{r_i}|l_{r_i} \le K]T_{r_i} + \mathbf{E}[B(\mathbf{E}[l_{r_i}|l_{r_i} \le K])] \\ &\quad + \mathbf{E}[q_i]\}\mathbf{P}[h > i-1|h<n] + \sum_{i=0}^{n-1}\left(KT_{r_{i+1}} + B(K)\right. \\ &\quad \left. + \mathbf{E}[q_{i+1}]\right)\mathbf{P}[h = i|h<n].\end{aligned} \tag{7}$$

Note that in the above expression, $l_{r_j} \le K$ if a link j is successfully traversed with rate r_j. Furthermore, in simplifying Eq. (6) to Eq. (7) $\sum_{j=1}^{0} = 0$ is used.

Since $h_{z\_n} = n$, the second term in Eq. (4) reduces to

$$\mathbf{E}\left[\sum_{j=1}^{h_{z_n}}\left(l_{r_j}T_{r_j} + B(l_{r_j}) + q_j\right)\right] = \sum_{j=1}^{n}\left(\mathbf{E}\left[l_{r_j}|l_{r_j} \le K\right]T_{r_j} + \mathbf{E}\left[B\left(\mathbf{E}\left[l_{r_j}|l_{r_j} \le K\right]\right)\right] + \mathbf{E}\left[q_j\right]\right). \tag{8}$$

Inserting Eq. (7) and (8) into Eq. (4) and using the definition of L_{r_j} and Q_j, Eq. (3) is obtained. □

Now we estimate L_{r_j}, $\mathbf{P}[h > j-1|h<n]$ and $\mathbf{P}[h = j - 1|h < n]$ (in Eq. (3)) in order to express $\mathbf{C}(\mathbb{R}_n)$ in terms of the link success probabilities for each rate. From its definition, L_{r_j} can be further expressed as

$$L_{r_j} = \sum_{i=1}^{K}\frac{i \cdot \mathbf{P}\left(l_{r_j} = i\right)}{1 - \mathbf{P}\left(l_{r_j} > K\right)} = \sum_{i=1}^{K}\frac{i \cdot \left(1 - p_{r_j}\right)^{i-1}p_{r_j}}{1 - \left(1 - p_{r_j}\right)^{K}}. \tag{9}$$

We define π_{r_j} to be the probability that a packet transmitted with data rate r_j is not dropped on link j, i.e., $\pi_{r_j} = 1 - (1 - p_{r_j})^K$. With this, the probability that a packet transmitted with rates (r_1, ..., r_i) is delivered over i successive hops is given by $\rho_i = \pi_{r_1} \times \cdots \times \pi_{r_i}$. We then obtain $\mathbf{P}[h > j] = \pi_{r_1} \times \cdots \times \pi_{r_{j+1}} = \rho_{j+1}$. In particular, from $\mathbf{P}[h \geq j] = \rho_j$, $\rho_0 = 1$ is induced. Furthermore, since $z_n = l$ indicates that there are $l - 1$ transport layer transmission failures before the first success, z_n has a geometric distribution with parameter $\mathbf{P}[h \geq n] = \rho_n$. With the new variables defined above, $\mathbf{P}[h > j - 1 | h < n]$ and $\mathbf{P}[h = j - 1 | h < n]$ in Eq. (3) can be expressed in the following way:

$$\mathbf{P}(h > j - 1 | h < n) = \frac{\mathbf{P}(h > j - 1) - \mathbf{P}(h \geq n)}{1 - \mathbf{P}(h \geq n)} = \frac{\rho_j - \rho_n}{1 - \rho_n}. \tag{10}$$

and

$$\begin{aligned}\mathbf{P}(h = j - 1 | h < n) &= \frac{\mathbf{P}(h \geq j - 1) - \mathbf{P}(h > j - 1)}{1 - \mathbf{P}(h \geq n)} \\ &= \frac{\rho_{j-1} - \rho_j}{1 - \rho_n}.\end{aligned} \tag{11}$$

With these new simplifications, we now re-express $\mathbf{C}(\mathbb{R}_n)$ in Lemma 1, in terms of the link success probabilities.

Lemma 1 *The expected transmission cost,* $\mathbf{E}[\mathbf{C}(\mathbb{R}_n)]$, *for delivering a packet with rates* $\mathbb{R}_n = (r_1, \ldots, r_n)$ *over path* $(v_0, ..., v_n)$ *is given by*

$$\begin{aligned}\mathbf{E}[\mathbf{C}(\mathbb{R}_n)] = \sum_{j=1}^{n} \Bigg\{ & \frac{\rho_j}{\rho_n} \left(L_{r_j} T_{r_j} + B\left(L_{r_j}\right) + Q_j\right) \Bigg) \\ & + \frac{\rho_{j-1} - \rho_j}{\rho_n} \left(K T_{r_j} + B(K) + Q_j\right) \}.\end{aligned} \tag{12}$$

Proof By substituting Eq. (10) and Eq. (11) into Eq. (3), and by using the fact that z_n follows a geometric distribution, Eq. (3) reduces to

$$\begin{aligned}\mathbf{E}[\mathbf{C}(\mathbb{R}_n)] = & \left(\frac{1}{\rho_n} - 1\right) \Bigg\{ \sum_{j=1}^{n-1} \frac{\rho_j - \rho_n}{1 - \rho_n} \left(L_{r_j} T_{r_j} + B(l_{r_j}) + Q_j\right) \\ & + \sum_{j=1}^{n} \frac{\rho_{j-1} - \rho_j}{1 - \rho_n} \left(K T_{r_j} + B(K) + Q_j\right) \} \\ & + \sum_{j=1}^{n} \left(L_{r_j} T_{r_j} + B(l_{r_j}) + Q_j\right) \\ = & \sum_{j=1}^{n} \frac{\rho_j}{\rho_n} \left(L_{r_j} T_{r_j} + B(l_{r_j}) + Q_j\right) \\ & + \sum_{j=1}^{n} \frac{\rho_{j-1} - \rho_j}{\rho_n} \left(K T_{r_j} + B(K) + Q_j\right).\end{aligned} \tag{13}$$

Putting both the terms on the right-hand side in the last equation together within a single summation yields Eq. (12). □

Lemma 2 *ETM satisfies the following recursive equation:*

$$\begin{aligned}\mathbf{E}[\mathbf{C}(\mathbb{R}_{n+1})] = & \frac{\mathbf{E}[\mathbf{C}(\mathbb{R}_n)]}{\pi_{r_{n+1}}} + \frac{1 - \pi_{r_{n+1}}}{\pi_{r_{n+1}}} \left(K T_{r_{n+1}} + B(K) + Q_{n+1}\right) \\ & + L_{r_{n+1}} T_{r_{n+1}} + B(L_{r_{n+1}}) + Q_{n+1}.\end{aligned} \tag{14}$$

where $\mathbf{E}[\mathbf{C}(\mathbb{R}_0)] = 0$.

Proof Let $\mathbf{E}[\mathbf{C}(\mathbb{R}_{n+1})]$ be the cost of path $(v_0, ..., v_{n+1})$. From Eq. (12), $\mathbf{E}[\mathbf{C}(\mathbb{R}_{n+1})]$ is given by

$$\begin{aligned}\mathbf{E}[\mathbf{C}(\mathbb{R}_{n+1})] = & \frac{1}{\pi_{r_{n+1}}} \sum_{j=1}^{n} \Bigg\{ \frac{\rho_j}{\rho_n} \left(L_{r_j} T_{r_j} + B\left(L_{r_j}\right) + Q_j\right) \\ & + \frac{\rho_{j-1} - \rho_j}{\rho_n} \left(K T_{r_j} + B(K) + Q_j\right) \Bigg\} + L_{r_{n+1}} T_{r_{n+1}} \\ & + B\left(L_{r_{n+1}}\right) + Q_{n+1} + \frac{\rho_n - \rho_{n+1}}{\rho_{n+1}} \left(K T_{r_{n+1}} + B(K) + Q_{n+1}\right) \\ = & \frac{\mathbf{E}[\mathbf{C}(\mathbb{R}_n)]}{\pi_{r_{n+1}}} + L_{r_{n+1}} T_{r_{n+1}} + B(L_{r_{n+1}}) + Q_{n+1} + \frac{1 - \pi_{r_{n+1}}}{\pi_{r_{n+1}}} \\ & \cdot \left(K T_{r_{n+1}} + B(K) + Q_{n+1}\right).\end{aligned} \tag{15}$$

□

As the equation shows, ETM has the recursive property, and can be decomposed into the expected retransmission cost due to packet failure on the last link (the first two terms of Eq. (14)) and the expected packet transmission time on the last link including queuing delay.

5 Our algorithms

In this section, we revisit the Optimal Path Problem (**OPP**) and the Optimal Rate Problem (**ORP**), that we had mentioned earlier in Sect. 1 The solution to OPP should find the best path with the least ETM cost for a given node pair. The solution to ORP should select the best rates for each link on a given path. Although the metric is non-commutative, a greedy algorithm can solve OPP; the solution to ORP is based on a dynamic-programming approach.

5.1 Our route selection algorithm

Formally OPP may be posed as follows: *Given a node pair, find the path between the pair with the minimum ETM cost.*

Lemma 3 *ETM satisfies* ***right-isotonicity*** *and* ***right-monotonicity****.*

Proof Right-isotonicity states that the order relation between the weights (costs) of any two paths is preserved if both of them are appended by a common third path [28, 29].

We assume two paths A and B whose start and end node are same, satisfying $\mathbf{E}[\mathbf{C}(\mathbb{R}_A)] \leq \mathbf{E}[\mathbf{C}(\mathbb{R}_B)]$. In order to prove the right-isotonicity, we need to show if $\mathbf{E}[\mathbf{C}(\mathbb{R}_A)] \leq \mathbf{E}[\mathbf{C}(\mathbb{R}_B)]$, then $\mathbf{E}[\mathbf{C}(\mathbb{R}_{A\oplus C})] \leq \mathbf{E}[\mathbf{C}(\mathbb{R}_{B\oplus C})]$, where $p \oplus q$ denotes the path formed by the concatenation of path p and q in order. Utilizing the result of Lemma 2, we derive the ETM of a path $L, \mathbf{E}[\mathbf{C}(\mathbb{R}_L)]$, as the function of $\mathbf{E}[\mathbf{C}(\mathbb{R}_l)]$, which is the ETM of the path from a source to intermediate node l on L, as follows

$$\mathbf{E}[\mathbf{C}(\mathbb{R}_L)] = \frac{\mathbf{E}[\mathbf{C}(\mathbb{R}_l)]}{\pi_{r_1}\cdots\pi_{r_\omega}} + \sum_{i=1}^{\omega}\frac{(1-\pi_{r_i})\Gamma_i + \Lambda_{i-1}}{\pi_{r_i}\cdots\pi_{r_\omega}} + \Lambda_\omega, \tag{16}$$

where ω is the number of links required for node l to reach the end node of L along with the path, and Γ_i and Λ_i are $KT_{r_i} + B(K) + Q_i$ and $L_{r_i}T_{r_i} + B(L_{r_i}) + Q_i$, respectively. $\mathbf{E}[\mathbf{C}(\mathbb{R}_{A\oplus C})]$ and $\mathbf{E}[\mathbf{C}(\mathbb{R}_{B\oplus C})]$ are then expressed as

$$\mathbf{E}[\mathbf{C}(\mathbb{R}_{A\oplus C})] = \frac{\mathbf{E}[\mathbf{C}(\mathbb{R}_A)]}{\pi_{r_1}\cdots\pi_{r_\eta}} + \sum_{i=1}^{\eta}\frac{(1-\pi_{r_i})\Gamma_i + \Lambda_{i-1}}{\pi_{r_i}\cdots\pi_{r_\eta}} + \Lambda_\eta, \tag{17}$$

$$\mathbf{E}[\mathbf{C}(\mathbb{R}_{B\oplus C})] = \frac{\mathbf{E}[\mathbf{C}(\mathbb{R}_B)]}{\pi_{r_1}\cdots\pi_{r_\eta}} + \sum_{i=1}^{\eta}\frac{(1-\pi_{r_i})\Gamma_i + \Lambda_{i-1}}{\pi_{r_i}\cdots\pi_{r_\eta}} + \Lambda_\eta, \tag{18}$$

where η is the number of links of path C. This leads to $\mathbf{E}[\mathbf{C}(\mathbb{R}_{A\oplus C})] \leq \mathbf{E}[\mathbf{C}(\mathbb{R}_{B\oplus C})]$ since it is assumed that $\mathbf{E}[\mathbf{C}(\mathbb{R}_A)] \leq \mathbf{E}[\mathbf{C}(\mathbb{R}_B)]$; this is the proof of right-isotonicity. To prove right-monotonicity, we need to show $\mathbf{E}[\mathbf{C}(\mathbb{R}_A)] \leq \mathbf{E}[\mathbf{C}(\mathbb{R}_{A\oplus B})]$. Due to non-negativity of link cost and $\pi_{r_i} \leq 1$, it is derived that $\mathbf{E}[\mathbf{C}(\mathbb{R}_{A\oplus B})] - \mathbf{E}[\mathbf{C}(\mathbb{R}_A)] \geq 0$. □

It is shown that Dijkstras algorithm is guaranteed to find the lightest paths if and only if the path weight structure is right-isotonic and right-monotonic [29]. Based on this, we propose an optimal route selection algorithm for OPP. To this end, we modify the well known Dijkstra's algorithm for finding the shortest path between any two nodes. In the original Dijkstra's algorithm, the path cost *dist*[*v*] from the source node to node *v* via node *u* is given by *dist*[*v*] = *dist*[*u*] + *cost*(*u*, *v*), where *cost*(*u*, *v*) is the link cost between *u* and *v*. In our case, we use the following equation for computing *dist*[*v*]:

$$\begin{aligned} dist[v] = \min_{r_{u,v}\in R}\Bigg\{&\frac{dist[u]}{\pi_{r_{u,v}}} + L_{r_{u,v}}T_{r_{u,v}} + B(L_{r_{u,v}}) + Q_{u,v} \\ &+ \frac{1-\pi_{r_{u,v}}}{\pi_{r_{u,v}}}\left(KT_{r_{u,v}} + B(K) + Q_{u,v}\right)\Bigg\}, \end{aligned} \tag{19}$$

where $r_{u,v}$ and $\pi_{r_{u,v}}$ are the rate from u to v and the probability that the packet is not dropped within K attempts from u to v when it is transmitted at rate $r_{u,v}$, respectively. Note that the minimization in Eq. (19) ensures the selection of the best rate between u and v (at the time of path computation), and thereby the minimum ETM cost from the source to v. The details of the proposed path selection algorithm are represented in Algorithm 1.

Algorithm 1: Path Algorithm

```
Input: Graph $G(V, E)$ and Source $s$
Initialization: $dist[u] \leftarrow \infty$ and $prev[u] \leftarrow \emptyset$ for $\forall\ u \in V/\{s\}$,
$dist[s] \leftarrow 0$, $rate[u, v] \leftarrow 0$, and $\Phi \leftarrow V$;
/* $prev[u]$ is the node that immediately follows $u$,
and $v$ is the neighbor of $u$                          */;
while $\Phi \neq \emptyset$ do
    $u \leftarrow MIN(\Phi)$;
    /* $MIN(\Phi)$ is the element in $\Phi$ that has the
    least path cost                                     */;
    $\Phi \leftarrow \Phi - \{u\}$;
    for each neighbor $v$ of $u$ do
        $tmp \leftarrow$
        $\min_{r_{u,v}\in R}\left\{\frac{dist[u]}{\pi_{r_{u,v}}} + L_{r_{u,v}}T_{r_{u,v}} + B(L_{r_{u,v}}) + \right.$
        $\left. + Q_{u,v}\frac{1-\pi_{r_{u,v}}}{\pi_{r_{u,v}}}\left(KT_{r_{u,v}} + B(K) + Q_{u,v}\right)\right\}$;
        if $tmp < dist[v]$ then
            $dist[v] \leftarrow tmp$;
            $prev[v] \leftarrow u$;
            $rate[u, v] \leftarrow \mathrm{argmin}_{r_{u,v}\in R}\left\{\frac{dist[u]}{\pi_{r_{u,v}}} + \right.$
            $L_{r_{u,v}}T_{r_{u,v}} + B(L_{r_{u,v}}) + Q_{u,v} +$
            $\left.\frac{1-\pi_{r_{u,v}}}{\pi_{r_{u,v}}}\left(KT_{r_{u,v}} + B(K) + Q_{u,v}\right)\right\}$;
        end
    end
end
```

5.2 Our rate selection algorithm

Once a route is determined between a node pair, the route is used for a while (since frequent route changes can be overhead intense; the proposed path selection algorithm is triggered to run at every 20 seconds for the evaluation in Sect. 5) However, the optimal rate for use with each link is time-varying. In other words, the rates determined by the routing algorithm may not be optimal for the duration for which the route is used. To cope with this, we propose a rate adaptation algorithm that tunes the rate dynamically on a short-term basis; our algorithm implicitly accounts for the link positions on the path to ensure that packet drops closer to the destination are less likely. In particular, here we address the ***Optimal Rate Problem (ORP):*** *Given a path* $(v_0, \cdots, v_n)$, *determine the rate set* $\mathbb{R}_n$ *that minimizes the expected transmission cost for the path*, $\mathbf{E}[\mathbf{C}(\mathbb{R}_n)]$.

Lemma 4 *ORP satisfies the overlapping property, i.e., the problem can be broken down into smaller subproblems that retain the same structure.*

Proof Formally, ORP can be expressed as:

$$\begin{aligned} \min_{\mathbb{R}_n\in R^n}\mathbf{E}[\mathbf{C}(\mathbb{R}_n)] = \min_{r_n\in R}\Bigg\{&\frac{\min_{\mathbb{R}_{n-1}\in R^{n-1}}\mathbf{E}[\mathbf{C}_{n-1}(\mathbb{R}_{n-1})]}{\pi_{r_n}} \\ &+ L_{r_n}T_{r_n} + B(L_{r_n}) + Q_n + \frac{1-\pi_{r_n}}{\pi_{r_n}}\left(KT_{r_n} + B(K) + Q_n\right)\Bigg\}, \end{aligned} \tag{20}$$

where R^n is an n-dimensional vector space over the possible set of rates R. The calculation of the right-hand side of Eq. (20) for each rate r_n, requires the calculation of $\min_{\mathrm{I}}$ R$_{n\text{-}1}$ $\in R^{n\text{-}1}\mathbf{E}[\mathbf{C}(\mathrm{I}$ R$_{n\text{-}1})]$. This indicates that ORP for the path (v_0, ..., v_n) can be solved if the sub-problem (again an ORP) for the path (v_0, ..., $v_{n\text{-}1}$) is solved. □

Lemma 5 *ORP satisfies the optimal substructure property: if we have the optimal rate set for a problem, then the associated rate set for each sub-problem is also optimal.*

Proof We prove the Lemma by contradiction. Let I R$_n^*$ = (r_1^*, ..., r_n^*) be the optimal rate set for ORP for the path (v_0, ..., v_n). We now assume that the rate set I R$_{n\text{-}1}^{*}$ = (r_1^*, ..., $r_{n\text{-}1}^*$) for the sub-problem with sub-path (v_0, ..., $v_{n\text{-}1}$) is not optimal i.e., the optimal substructure property does not hold. In other words, we assume that the path cost $\mathbf{E}\left[\mathbf{C}(\mathbb{R}_{n-1}^*)\right]$ is not minimal. Given this assumption, there exists a rate set $\mathbb{R}_{n-1}' = \left(r_1', \cdots, r_{n-1}'\right)$ with its path cost $\mathbf{E}\left[\mathbf{C}(\mathbb{R}_{n-1}')\right]$ satisfying $\mathbf{E}\left[\mathbf{C}(\mathbb{R}_{n-1}')\right] < \mathbf{E}\left[\mathbf{C}(\mathbb{R}_{n-1}^*)\right]$.

$$\mathbf{E}\left[\mathbf{C}(\mathbb{R}_{n-1}')\right] < \mathbf{E}\left[\mathbf{C}(\mathbb{R}_{n-1}^*)\right]. \tag{21}$$

Then, we can find a new rate set I R$_n'$ = (r_1', ..., $r_{n\text{-}1}'$, r_n^*) whose path cost $\mathbf{E}[\mathbf{C}(\mathrm{I}$ R$_n')]$ satisfies the following inequality:

$$\mathbf{E}\left[\mathbf{C}(\mathbb{R}_n')\right] = \frac{\mathbf{E}\left[\mathbf{C}(\mathbb{R}_{n-1}')\right]}{\pi_{r_n^*}} + L_{r_n^*}T_{r_n^*} + B(L_{r_n^*}) + Q_n^* + \frac{1-\pi_{r_n^*}}{\pi_{r_n^*}}\left(KT_{r_n^*} + B(K) + Q_n^*\right) < \mathbf{E}\left[\mathbf{C}(\mathbb{R}_n^*)\right]. \tag{22}$$

However, this contradicts the postulate that I R$_n^*$ is the optimal rate set for the path (v_0, ..., v_n). □

Proposition 2 *ORP can be solved by using dynamic programming.*

Proof An optimization problem can be solved by dynamic programming if the problem satisfies both the overlapping and the optimal substructure properties [30]. The proof is immediate from Lemmas 4 and 5. □

Based on the above properties of ORP, we propose an optimal rate selection algorithm using a dynamic programming technique. The proposed rate selection algorithm chooses a rate for link j (for $j = 1$, ..., n) such that the chosen rate satisfies the following equation:

$$\mathcal{V}_j = \min_{r_j \in R}\left\{\frac{\mathcal{V}_{j-1}}{\pi_{r_j}} + L_{r_j}T_{r_j} + B(L_{r_j}) + Q_j + \frac{1-\pi_{r_j}}{\pi_{r_j}}\left(KT_{r_j} + B(K) + Q_j\right)\right\}, \tag{23}$$

where $\mathcal{V}_0 = 0$ for $j = 0$ and $\mathcal{V}_j$ is the expected transmission cost for the delivery of a packet from node v_0 to node v_j. The details of the proposed rate selection algorithm are provided in Algorithm 2.

Note that given the graph $G(V, E)$, it can be easily shown that the complexity of Algorithm 1 and Algorithm 2 are $O(R|V|^2)$ and $O(R|V|)$, respectively [30].

Algorithm 2: Rate Algorithm

```
Input: $\mathcal{V}_j$
Output: $r_{j+1}$
foreach $v_j$ in the path $(v_0, \cdots, v_j, \cdots, v_n)$ do
    $r_{j+1} \leftarrow \operatorname{argmin}_{r_{j+1} \in R}\left\{\frac{\mathcal{V}_j}{\pi_{r_{j+1}}} + L_{r_{j+1}}T_{r_{j+1}} + B(L_{r_{j+1}}) + Q_{j+1} + \frac{1-\pi_{r_{j+1}}}{\pi_{r_{j+1}}}\left(KT_{r_{j+1}} + B(K) + Q_{j+1}\right)\right\}$;
end
```

V. IMPLEMENTATION AND EVALUATIONS

6 Implementation and evaluations

In this section, we first describe the implementation of the ETM-framework. Then, we compare its performance with that of the ETOP-and ETT-frameworks. The term ETT-framework (ETOP-framework) refers to the combination of ETT-routing (ETOP-routing) and the original SampleRate algorithm. In our first set of experiments, we modify the framework to exclude the queuing delay when computing the ETM metric i.e., we set $Q_j = 0$ (Sects. 5.2, 5.3, 5.4, and 5.5). Our goal is to exclusively capture the retransmission costs. The impact of the queuing delay is subsequently evaluated in Sect. 5.6.

6.1 Testbed and implementation

6.1.1 Testbed description

Our wireless testbed consists of 21 Soekris net5501 nodes, which run a Debian Linux distribution with kernel v2.6.16.19. Each node is equipped with a CM9 802.11 a/b/g miniPCI card [31], which embeds the Atheros AR5213 chipset and an omnidirectional antenna of 5 dBi gain. We use the Roofnet module in the popular Click toolkit [15] to implement our algorithms. We modify the MadWifi-0.9.3 driver [12] so that it can recognize RAW packets generated by Click. Experiments are performed with the 802.11a mode to avoid interference from co-located 802.11b/g networks. In our experiments we disable the RTS/CTS messages (as is commonly done [2]), and the default long retry limit of 7 is used by the wireless cards.

6.1.2 Implementation of ETM-routing

We implement the ETM-routing as a modified version of Srcr [4], a source-based routing protocol proposed for

Roofnet; it is similar to the DSR protocol [32]. With ETM-routing, each node periodically (every one second in the implementation) broadcasts a 1,500-byte probe packet at each possible data rate and a 60-byte packet at the basic rate. The larger probes reflect data transmissions in the forward direction and the smaller probes characterize the ACK transmissions in the reverse direction. Each probe packet contains queuing delay information (measuring the average queuing delay is detailed next) at the broadcasting node in its header. Each node maintains a cache of all the known routes, the average queuing delay at the nodes on the routes, and the delivery rates of the probes over their corresponding links. Whenever the source node needs to send a packet, the source first checks whether the destination is in the cache. If the destination is found, the source runs Algorithm 1, proposed in Sect. 4, on the graph constructed with the topology in the cache to find the route with the minimum ETM cost. When each node on the route forwards a packet, it includes its latest queuing delay and the current delivery rates of the probe packets between itself and the previous node into the packet header; thus, at these times all the nodes on the route update the rate vectors and the queuing delay information in their caches. To avoid route flapping, average queuing delays are only updated if the change is higher than a preset threshold.

If the source cannot find the destination in its cache, it invokes a network-wide query. Every node which receives the query, inserts its own address, its queuing delay information, and probe delivery statistics (between itself and the node from which it received the query) in the packet header and re-broadcasts it. When the destination node receives the query, it responds to the source; this response is sent on the reverse path corresponding to the route via which the packet was received and contains the information that was obtained in the query. The source updates the node cache, the queuing delays of the associated nodes, and the corresponding delivery rates on each link from the information. It then runs Algorithm 1 to compute a new route to the destination.

6.1.3 Estimating the mean queuing delays

To measure the average queuing delay at a node, each packet is time-stamped at the instant it is enqueued and at the instant when it is to be transmitted. The difference between two time-stamps is measured to determine the queuing delay of the packet. Let $x_{j,t}$ be the measured queuing delay of the t^{th} enqueued packet at the transmitter of a link j. Then average queuing delay Q_j for node j is estimated by using the following exponential smoothing formula:

$$Q_j = \alpha x_{j,t} + (1 - \alpha) s_{j,t-1}, \tag{24}$$

where α is the smoothing factor ($0 < \alpha < 1$). $s_{j,t\text{-}1}$ is the weighted mean of the queuing delays of last u enqueued packets; it is computed as $s_{j,t\text{-}1} = \sum_{n=1}^{u} \omega_n x_{j,t\text{-}n}$, where the weighted factors ω_n satisfy the property $\sum_{n=1}^{u} \omega_n = 1$. If $\alpha \approx 1$ there is a lesser extent of smoothing and recent changes are considered more important. In our experiments we choose $\alpha = 0.5$ to balance the two factors and $1/u$ for ω_n.

6.1.4 Implementation of ETM-rate adaptation

We implement the ETM-rate adaptation module on top of SampleRate [4]. SampleRate periodically invokes transmissions of 1,500-byte data packets at a randomly chosen bit-rate; it computes the packet delivery ratio (PDR) and uses this to compute the rate that yields the best throughput. Computation of the PDRs is based on actual data transmissions (much faster time scale) rather than on periodic broadcast probes and can respond to the time-varying channel quality more quicky and accurately. We modify the SampleRate algorithm as follows. Whenever a node v_j wants to transmit a packet to its neighbor v_{j+1}, it runs Algorithm 2, proposed in Sect. 4, using the statistics obtained with the native SampleRate algorithm. It obtains r_{j+1} and $\mathcal{V}_j$ and includes this information in the packet header and sends the packet at rate r_{j+1}. The route remains the same; only the rates that are used on the links of the route change. The rate adapts to cope with temporal changes in link quality in the short term; as discussed, our approach (Algorithm 2) takes into account the position of the link on the path.

6.1.5 Implementation complexity

The use of ETM requires the embedding of additional information (rate vector and the queuing delay for ETM-routing and the information on r_{j+1} and $\mathcal{V}_j$ for ETM-rate adaptation) in the packet header. As compared to ETT or ETOP, this adds an overhead of 2–5 % depending on path length. In particular, the information embedded with ETT is of the order of 16 bytes: two directions (forwards and reverse) between two neighboring nodes requires PDR information for each rate, which requires 1 byte (there are eight rates supported for IEEE 802.11a), while it is 20 bytes with ETM; in addition to the PDR information, ETM requires 4 additional bytes: 2 bytes for the queuing delay, 1 byte for r_{j+1}, and 1 byte for $\mathcal{V}_j$. Compared to a packet size of 1,500 bytes, we believe that this is sufficiently small. This small overhead provides throughput improvements of up to 253 and 368 % as compared with the ETT and ETOP frameworks, as will be shown in Sect. 5.6 Additional processing delays due to ETM are not noticable in our implementation. In other words, packet generation rates and CPU utilization remain almost the same as with ETT and ETOP.

6.2 Performance of TCP flows

We first consider the impact of the considered frameworks on the performance of TCP connections.

6.2.1 Setup

In our first set of experiments, a large number of source-destination pairs, 70, are randomly chosen out of the 420 (=21 × 20) possible pairs; 26 pairs are separated by a single-hop and 44 pairs are separated by multiple hops. At the start of an experiment, Roofnet is allowed to run for 20 s in order to reach stable operations. Then, the source of the pair pings the destination for 20 s at a rate of one packet per second. With this, Roofnet discovers the paths to the destination. Right after the ping session, the source initiates a TCP connection with the destination, and sends data for 3 minutes. During the session, the maximum achieved TCP throughput is measured using Iperf [33]. Immediately after, we repeat the process with the next considered framework. Thus, the results with each framework are obtained within minutes of each other; we expect the slower time-scale channel conditions to have changed little during this time, which is something that we experimentally observe during our experiments. Each run for the 70 pairs takes approximately 13 h. We repeat the experiment seven times and compute an average to reduce the impact of temporal variations.

6.2.2 TCP performance of each framework

In Table 2 we tabulate the median throughputs[2] and the weighted path length (**WPL**) for the 70 considered flows, with each framework. *WPL* is defined as the length of the path weighted by the number of packets sent over that path during a TCP flow. A larger *WPL* value indicates that longer paths are found (possibly supporting higher rates on the component links). Note that path lengths can change during the course of the session. Node pairs are grouped into one of four types according to path length between them (depending on the minimum number of hops observed in the node connectivity graph). The results show that over all pairs the ETM-framework outperforms the ETOP-and ETT-frameworks in terms of the median TCP throughput by 131 and 30 %, respectively.

Table 2 Median TCP throughput versus the weighted average path length for all 70 pairs

Path length	# of pairs	Median TCP Throughput (Kbps)/WPL		
		ETOP	ETT	ETM
All length	70	1275 / 2.03	2255 / 2.18	2940 / 2.22
1-hop	26	18000 / 1.01	16250 / 1.13	17600 / 1.09
2-hops	22	1125 / 2.00	1955 / 2.19	2195 / 2.27
3-hops	18	586 / 3.10	979 / 3.20	1250 / 3.27
4-hops	4	379 / 4.00	575 / 4.31	783 / 4.39

We further examine the CDFs of the achieved throughputs between node pairs in Fig. 2. We see that the CDFs of the TCP throughputs for the pairs that are separated by one hop, are statistically identical with the different frameworks (Fig. 2(a)). This is because with all of the frameworks, the direct link between the considered pair is chosen rather than an alternate multi-hop path; the channel quality on the direct links is sufficiently good and the multi-hop forwarding penalty incurred on longer paths hurts the throughput. In this case the position of the link has little or no bearing and ETM offers little or no improvements over ETOP or ETT. ETM offers significant throughput improvements compared to what is achieved with ETT and ETOP for node-pairs that are separated by two or more hops. In particular, the ETM-framework achieves median throughput improvements up to 113 % (for pairs separated by 4 hops) as compared with ETOP. The improvements over ETT amounts to 36 % for pairs separated by 4 hops.

6.2.3 Performance analysis

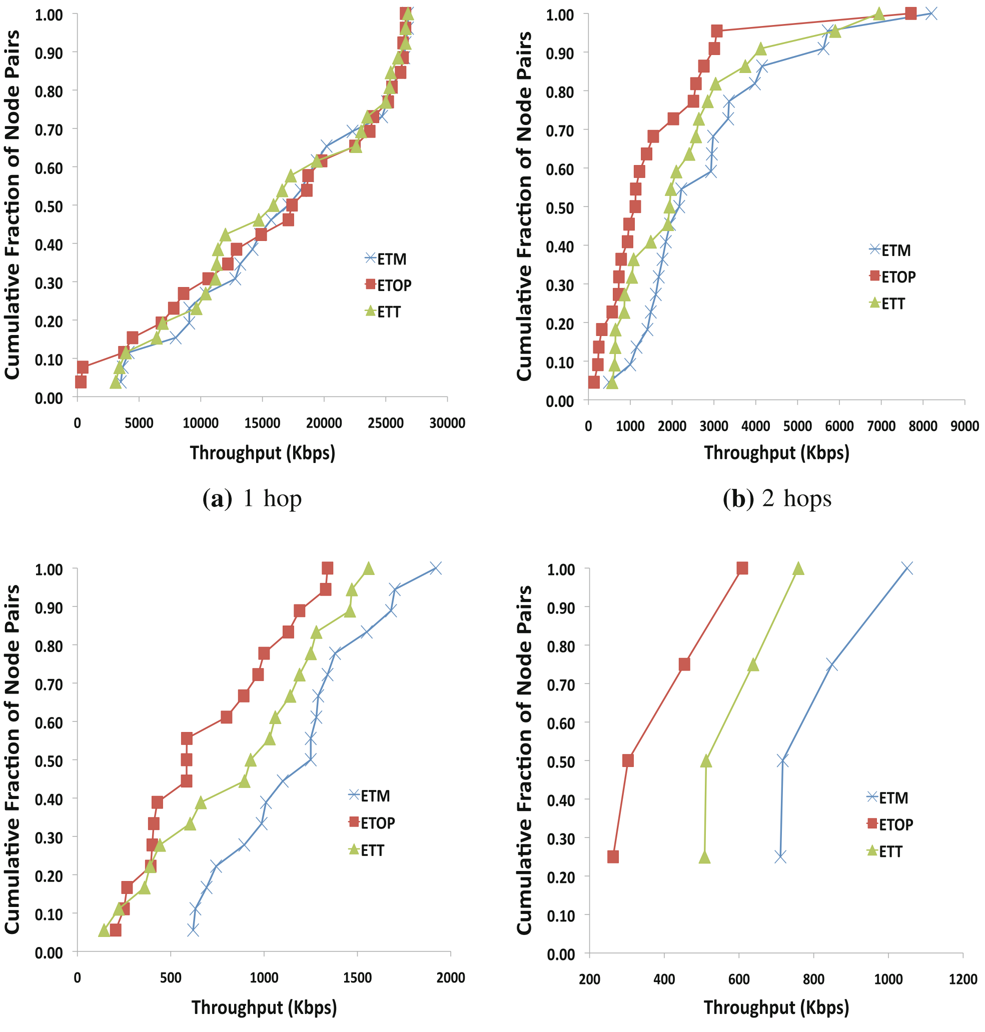
To understand the performance differences between the frameworks, we take a closer look at the performance of each framework over long paths; note that all of the frameworks are designed to primarily provide improvements over such paths. To this end, we present the detailed results with respect to four node pairs ($20 \rightarrow 24, 20 \rightarrow 25, 20 \rightarrow 40,$ and $24 \rightarrow 20$) (see Fig. 3); the paths between these pairs are the longest in our experiments.

Towards performing a comprehensive study, at each node the following statistics are gathered at the MAC layer using the Click-handler [35]: (i) the number of transmissions (including retransmissions), and (ii) the number of packets that are dropped (when the maximum limit on the number of possible retransmissions is exceeded).

We define two new metrics: the Effective number of transmissions and Retransmissions (**EnR**) and the Effective number of Dropped packets (**EnD**). For a TCP flow, *EnR* is computed as the ratio of the packets received at the destination to the total number of transmissions and retransmissions attempted at the MAC layer for that flow (at every

[2] When the distribution of the data is skewed (as it is in our case), the median is more representative of the observed behaviors than the mean [34].

Fig. 2 The CDF of the throughputs for pairs separated by different path lengths with each of the frameworks



(a) 1 hop

(b) 2 hops

(c) 3 hops

(d) 4 hops

node along the path of the flow). This measure reflects the cost of delivering a TCP packet successfully from the source to the destination. The *EnD* of a TCP flow is defined to be the ratio of the total number of MAC layer packet drops associated with a flow to the total number of transport layer transmission attempts performed by the source for that flow. With this measure we estimate the fraction of packets (from those sent from the source) that were dropped en route the destination.

We also record all the paths traversed by a TCP flow and the number of packets sent over each of the paths. Based on the path and packet records, we introduce the metric Weighted Pair Reliability (**WPR**). *WPR* is defined as the path delivery ratio weighted by the number of packets sent over that path during a TCP flow; the path delivery ratio is computed as the product of the delivery ratios of the links on the path. The *WPR* metric captures the overall reliability with each considered framework. A higher *WPR* value is representative of increased reliability.

Table 3 shows the performance results for the four node pairs in terms of the newly defined measures. Our experiments validate our intuition that the ETM-framework reduces the number of transmissions needed for e2e reliable data delivery. As seen in Table 3, the ETM-framework (in typical cases) reduces the *EnR* and *EnD* significantly as compared to the ETOP- and ETT-frameworks; this in turn, leads to higher TCP throughputs. We also observe from the *WPR* and *WPL* metrics that ETM computes more reliable, albeit longer paths than both ETT and ETOP in typical cases. The important consequence is that these longer paths can more reliably carry high rate transmissions.

6.2.4 An in-depth look at the paths generated with each framework

Next, we examine the routes computed with ETT, ETOP, and ETM for one of the four pairs, 20 → 25. From Table 3, for this pair the ETM-framework achieves throughput improvements of about 40 and 137 % as compared with the ETT-framework and the ETOP-framework, respectively. The primary route established with each framework is depicted in Fig. 3.

We make the following observations with regards to the considered flow from the figure:

- ETOP-routing results in a route that does not exploit the multi-rate capability; since small probe packets attain high packet delivery rates over relatively long links, the route computed is short but can only support low rates. Thus the throughput is the poorest.
- ETT-routing chooses longer paths as compared to ETOP-routing (much longer) and ETM-routing (comparable); in particular, the segment between the nodes 22 and 25 is longer. However, the use of SampleRate results in the choice of high rates and consequently more drops on the links composing this segment. In particular, packet drop rates of 7.8 and 5.3 % are observed on the links 22 → 41 and 41 → 40; the corresponding average transmission rates on these links are 39.3 and 37.7 Mbps.
- ETM-routing chooses lower rates closer to the destination. In particular, links 22 → 30 and 30 → 25 experience packet drop rates of just 2.2 and 0.66 %; the corresponding average transmission rates on these links are lower than that with ETT (with our modifications of SampleRate) and equal to 30.4 and 35.4 Mbps.

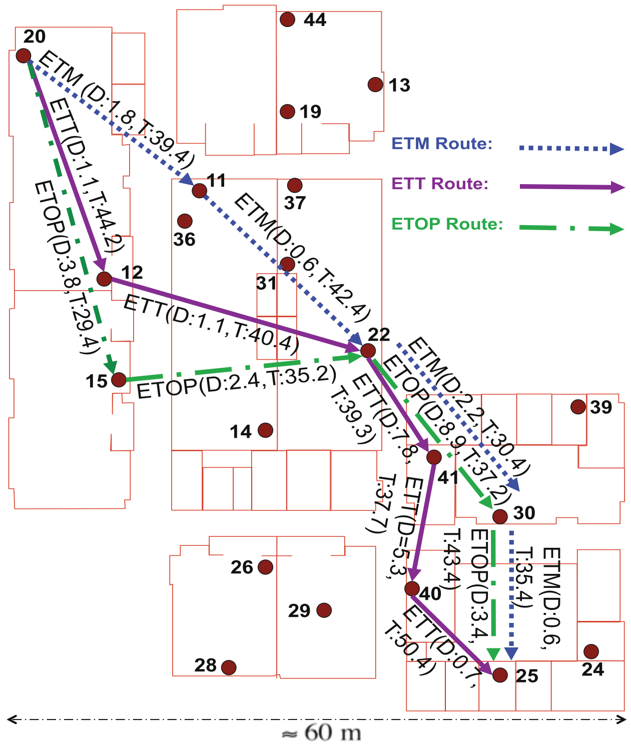


Fig. 3 The primary paths for node pair 20 → 25 that are selected by each framework; The following terms are used: D: the packet drop rate on the link (%), T: average transmission rate on the link (Mbps)

Note that the results with other multi-hop pairs exhibit behaviors consistent with the findings above.

6.3 Isolating the effects of ETM-routing and ETM-rate adaptation

Next, we examine whether each ETM-module is effective if activated in isolation. We find that each component can provide benefits on its own.

6.3.1 Setup

In order to examine the impact of ETM-routing, we select three pairs 20 → 24, 44 → 24, and 44 → 28; again, these nodes are separated by long paths (about four hops). We establish a TCP flow between each pair and perform experiments for 3 minutes as before. During this experiment, the SampleRate algorithm with each framework is turned off, and instead, the data rate that each routing module computes (while finding the route) is used for packet transmissions for the duration of the route. In this experiment, we only compare ETM-routing with ETT-routing since ETOP-routing does not have an associated rate selection.

For evaluating ETM-rate adaptation, we setup an experiment where only a single fixed path is used. In particular, we use the primary route discovered with ETT-routing for each pair; once the route is found, only the

Table 3 TCP flow results for the four node pairs with long paths

	20 → 25			20 → 24			20 → 40			24 → 20		
	ETOP	ETT	ETM	ETOP	ETT	ETM	ETOP	ETT	ETM	ETOP	ETT	ETM
Throughput (Kbps)	303	512	717	263	508	711	454	759	849	609	638	1050
EnR	13.2	11.3	6.9	17.2	10.9	5.7	12.8	10.1	8.9	10.3	11.9	6.3
EnD (%)	7.5	8.1	2.6	11.3	8.9	4.1	10.2	8.8	7.1	6.9	7.4	4.1
WPR	0.14	0.21	0.39	0.19	0.27	0.48	0.26	0.31	0.39	0.38	0.31	0.61
WPL	4.0	4.9	4.4	4.3	4.9	5.2	4.0	4.3	4.7	4.2	4.7	5.0

nodes on the path are kept turned on during the experiment. For the same reason as in the previous experiment, only ETT-routing is considered for performance comparisons. On the computed path we compare the performance with SampleRate and ETM-rate adaptation.

6.3.2 Isolated impact of ETM-routing

In Table 4 a summary of the results from the experiment detailed above is presented. For all the considered source-destination pairs, ETM-routing attains higher reliability in terms of *WPR* as compared to ETT-routing. As indicated by the *EnR* and *EnD* values, the routes computed with ETM-routing experience fewer link layer retransmissions and packet drops. The higher reliability achieved leads to higher throughputs than with ETT-routing.

6.3.3 Isolated impact of ETM-rate adaptation

Table 5 summarizes the results from our rate adaptation experiment. We use the "path reliability" as a measure of the reliability achieved between the source-destination pair (only one path exists now); we also show the average data rate used on each link due to the candidate rate adaptation modules. As seen from the *EnD* values, ETM-rate adaptation results in much fewer packet drops than SampleRate (used with the ETT-framework). This is because, SampleRate only considers retransmissions over a single hop while selecting a data rate. Thus, it selects data rates more aggressively; this leads to increased transmission failures (see table). Packet drops close to the destination are especially expensive.

The TCP throughputs attained in this experiment are higher than those attained for the same pair in the previous experiment (with ETM-routing in isolation). This is primarily because in the previous setting the data rates are changed at slower time-scales (only when routes change). Thus, links fail to adapt to short-term variations in quality; this may result in either overselection or underselection of rates and thus, lower throughputs.

Table 4 Experimental results to capture the effect of ETM-routing

	20 → 24		44 → 24		44 → 28	
	ETT	ETM	ETT	ETM	ETT	ETM
Throughput (Kbps)	324	398	207	273	598	659
EnR	16.9	11.8	19.8	13.7	10.3	8.2
EnD (%)	11.7	9.2	12.4	9.3	7.1	6.3
WPR	0.21	0.35	0.22	0.32	0.39	0.48
WPL	4.5	4.7	4.8	4.1	4.0	4.2

Table 5 Experimental results to capture the effect of ETM-rate adaptation

	20 → 24		44 → 24		44 → 28	
	ETT	ETM	ETT	ETM	ETT	ETM
Throughput (Kbps)	427	714	496	791	637	1042
EnR	16.2	7.3	12.1	8.6	9.9	5.7
EnD (%)	10.1	3.2	7.7	3.8	3.0	1.4
Path Reliability	0.48	0.69	0.64	0.76	0.57	0.79
Ave. Rate (Mbps)	39.2	30.3	37.7	29.1	40.9	33.7

6.4 Effect of interference

In the experiments so far, we only considered one active TCP flow at any given time. However, multiple TCP flows can interfere with each other. In this experiment, we evaluate the performance of the ETM-framework with multiple simultaneous TCP connections.

6.4.1 Setup

For the experiment five nodes that are at the periphery of our network (nodes 14, 20, 24, 39, and 44) are chosen and 10 temporally staggered TCP flows are established between distinct node pairs. Correspondingly, there are 5 × 4 possible pairs of nodes and $20 \times 10 = 200$ distinct TCP flows; each flow lasts for 3 mins. We control the number of simultaneously active connections by varying the times of initiations of the TCP flows. We also randomize the order in which the TCP flows are established between the node pairs.

6.4.2 Results

We use the Multiplied Median Throughput (**MMT**) metric which was proposed in [16], as an estimate of the achieved network-wide TCP throughput. *MMT* is the product of the number of concurrent flows and the median throughput achieved by the flows. Figure 4 shows the *MMT* values with ETOP, ETT, and the ETM-frameworks versus the number of simultaneously active connections. In all cases, the *MMT* value increases as we initially increase the number of concurrent flows; it then starts to drop. The peak *MMT* values with the ETOP, ETT, and ETM-frameworks appear with two, four, and five concurrent flows, respectively. As shown in Table 3 and Fig. 3, the inherent selection of *lossy inefficient* routes causes the capacity with ETOP to saturate with few flows. Beyond four concurrent flows, the *MMT* with the ETT-framework significantly drops. The reduction in link layer retransmissions with ETM leads to lower overhead and thus, reduced inter-flow interference. Thus, ***the ETM-framework achieves the best performance even in interference dominated settings***; as we see, the *MMT* value peaks with five flows with ETM.

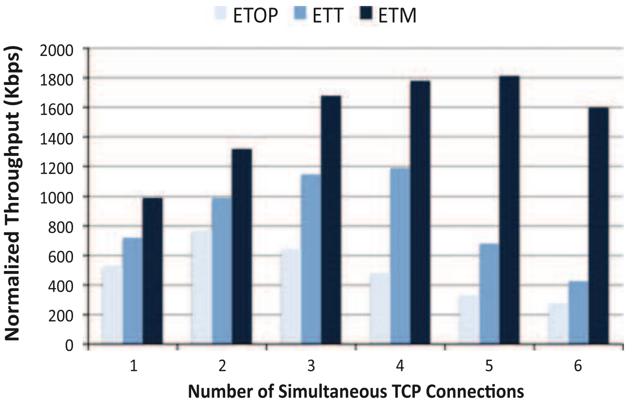


Fig. 4 Capturing the impact of multiple simultaneous TCP connections with the *MMT* metric

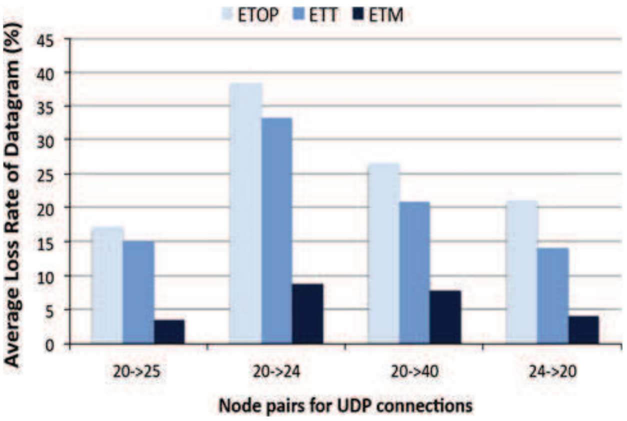


Fig. 5 Performance of frameworks over UDP flows evaluated by datagram loss rate

6.5 Performance of UDP flows

We have so far considered TCP to be the default transport layer protocol in our experimental evaluations. Since UDP is also a popular transport layer protocol, a natural question that arises is "How does ETM work with UDP?" Recall that the ETM cost metric was derived considering e2e retransmissions by a transport layer protocol; UDP does not perform e2e retransmissions. Nevertheless, the cost metric reduces costly and wasteful packet drops close to the destination by choosing "more reliable paths" to begin with, and reducing wasted transmissions by appropriately reducing rates as packets traverse closer to the destination. Furthermore, ETM rate adaptation also dynamically adjusts rates to make the paths more reliable. Given these features, one might expect the framework to provide performance benefits with UDP as well. We demonstrate that this is the case in our next set of experiments.

6.5.1 Setup

For these experiments, we consider the paths with the same source-destination nodes pairs $(20 \rightarrow 24, 20 \rightarrow 25, 20 \rightarrow 40,$ and $24 \rightarrow 20)$ as listed in Table 3; these paths correspond to the longest paths observed in our experiments. We establish and run a UDP flow on each pair for 3 minutes. For each UDP connection, we set the sending rate to be 1 Mbps. Our metric for performance comparison is the percentage average loss rate of datagrams (the ratio of the number of lost datagrams to the total number of datagrams sent, expressed as a percentage) during the connection. It is easy to compute the throughput from this percentage (the throughput in percentage is simply 100 % - percentage loss rate).

6.5.2 Results

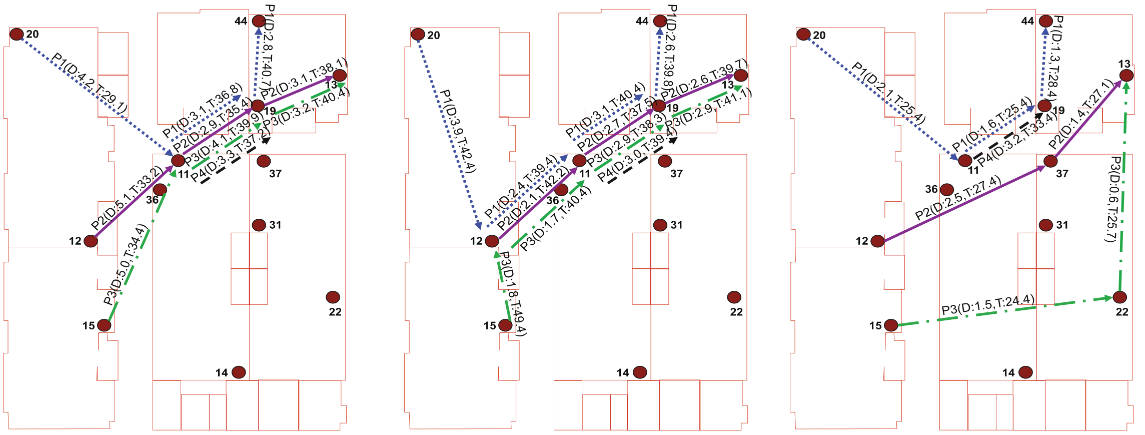
Figure 5 shows the percentage average loss rate of datagrams for each considered source-destination pair. In all the considered cases, the ETM-framework is observed to achieve higher reliability as compared to the other frameworks; in particular, the loss rates are 77 and 70 % lower on average, as compared to the ETOP and ETT-frameworks, respectively. This reduction in loss rate is profound when compared to the 50 and 46 % reduction in *EnD* with ETM as compared to the ETOP and ETT-frameworks (recall Table 3). This is because of the impact of the TCP congestion control mechanism; TCP decreases the sending rate when packet drops occur and this decreases the achievable gains as compared to UDP. The reduction in packet losses increases UDP throughput, reduces congestion and thus, benefits the entire network overall. We wish to point out that the behaviors reported here are consistent with what was observed with other node pairs in the network.

6.6 Effect of congestion

As mentioned earlier, all of the experiments described thus far ignored the queuing delay component in the ETM metric i.e., Q_j was set to zero. In this section, we perform experiments with the queuing delay included in the metric. Clearly, this has an impact primarily in settings with multiple-flows in the network. By incorporating Q_j in the metric, the ETM-framework balances the load across the network in such settings.

Setup: For the experiment, three distinct scenarios are considered. In scenario 1, four pairs $11 \rightarrow 19, 12 \rightarrow 44, 15 \rightarrow 13,$ and $20 \rightarrow 22$ are chosen. Pairs $11 \rightarrow 28, 14 \rightarrow 26, 22 \rightarrow 29,$ and $31 \rightarrow 29$ and pairs $24 \rightarrow 41, 25 \rightarrow 41, 30 \rightarrow 40$ and $39 \rightarrow 40$ are considered in scenario 2 and 3, respectively. In each scenario, using each framework we simultaneously establish TCP connections between each chosen pairs and send traffic between each pair for 3 minutes. Again experiments with different frameworks are conducted back-to-back to minimize the significant channel changes; thus, the results with each framework are obtained within minutes of each other. We set the smoothing factor α to 0.5 while estimating the average queuing delay with the ETM-framework.

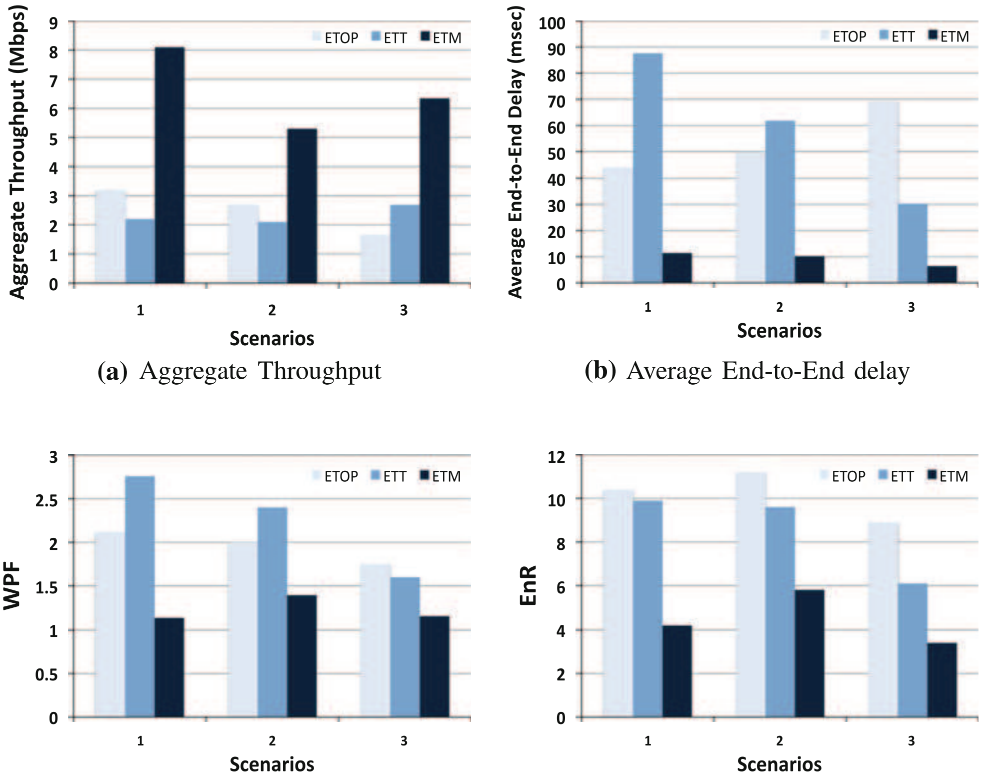
Results: The primary routes observed during the experiment with scenario 1, with each framework are depicted in Fig. 6 (due to the space constraints we only show the routes with scenario 1). As shown in the figure, both ETOP- and ETT-routing produce routes with large *overlaps*. In particular, the routes chosen by all of the flows include link 11 → 19. This causes large queuing delays at node 11 which in turn, increases the end-to-end delay and reduces the throughput of each flow. The overlap is higher with the ETT-framework and thus, the impact is higher. In particular, packets from three of the four flows also traverse node 12 before reaching node 11 as shown in Fig. 6(b).



(a) ETOP **(b)** ETT **(c)** ETM

Fig. 6 The primary paths of the four pairs for scenario 1 ($P1 : 11 \longrightarrow 19, P2 : 12 \longrightarrow 44, P3 : 15 \longrightarrow 13$, and $P4 : 20 \longrightarrow 22$), selected by each framework; the same terms are used as in Fig. 3: D: the packet drop rate on the link, T: average transmit rate on the link

Fig. 7 Performance metrics to capture the effect of ETM-framework in congested network scenarios



(a) Aggregate Throughput **(b)** Average End-to-End delay

(c) WPF **(d)** EnR

In contrast, we observe that ETM-routing results in load-balanced paths for the considered pairs (Fig. 6(c)); in particular, link 11 → 19 is now used by only two of the flows. In order to quantify the extent of load-balancing achieved with each framework, we introduce the metric Weighted number of Processed Flows (**WPF**). In order to define *WPF*, let f_j be the number of flows through node j (node j could either be the source or a relay for a flow) for a set of paths chosen by the candidate framework. The ratio $\frac{\sum_{j \in N} f_j}{|N|}$ is computed for the set of paths. Here $|N|$ is the cardinality of the set N of nodes that are either sources or act on relays for the flows. Note that with time the set of paths and N may change and in that case, a new ratio is computed. *WPF* is defined to be a weighted sum of such ratios where the weight associated with a ratio corresponds to the fraction of packets routed via the set of paths that determine the ratio. This measure reflects the load of each node. Figure 7(c) depicts the *WPF* values achieved by each framework for the three scenarios. With scenario 1, ETM-framework achieves the lowest *WPF*. Load-balanced transmissions reduce the average end-to-end delay per flow as shown in Fig. 7(b) and this leads to throughput improvements of about 253 and 368 % as compared with the ETOP- and ETT-framework, respectively (see Fig. 7(a)). We also observe from the *EnR* metric in Fig. 7(d) that even with the incorporation of queuing delay, reliable transmissions contribute significantly to achieved high throughput with the ETM-framework. Due to similar factors, the superior performance of ETM-framework is also observed in scenarios 2 and 3, as shown in Fig. 7.

7 Conclusions

In this paper, we propose an integrated routing and rate adaptation framework for multi-rate multi-hop wireless networks. Our framework is based on the use of a new metric ETM, and has three interdependent goals: (a) transmit packets with increased reliability as they traverse closer to the destination (b) achieve the best rates while adhering to the first goal, and (c) facilitate the load-balancing on requested traffic. We first analytically compute the ETM metric; the metric not only specifies the optimal path but the rates that are to be used on the path. Second, we design a framework that embeds the ETM metric in both the end-to-end routing and link level rate-adaptation modules. We implement our ETM-framework on an indoor wireless mesh network. We show that the use of the ETM-framework results in significant performance enhancements over the popular ETT-framework (designed for multi-rate networks) and the recently proposed ETOP-framework which accounts for link positions but does not account for multi-rate capabilities.

References

1. Jakllari, G., Eidenbenz, S., Hengartner, N., Krishnamurthy, S. V., & Faloutsos, M. (2008). Link positions matter: A noncommutative routing metric for wireless mesh network. In *INFOCOM 2008. The 27th conference on computer communications. IEEE*, pp. 744–752.
2. De Couto, D. S. J., Aguayo, D., Bicket, J., & Morris, R. (2005). A high-throughput path metric for multi-hop wireless routing. *Wireless Networks, 11*(4), 419–434.
3. Draves, R., Padhye, J., & Zill, B. (2004). Routing in multi-radio, multi-hop wireless mesh networks. In *Proceedings of the 10th annual international conference on mobile computing and networking (MobiCom '04)* (pp. 114–128). New York, NY: ACM.
4. Bicket, J., Aguayo, D., Biswas, S., & Morris, R. (2005). Architecture and evaluation of an unplanned 802.11b mesh network. In *Proceedings of the 11th annual international conference on mobile computing and networking (MobiCom '05)* (pp. 31–42). New York, NY: ACM.
5. Lacage, M., Manshaei, M. H., & Turletti, T. (2004). IEEE 802.11 rate adaptation: A practical approach. In *Proceedings of the 7th ACM international symposium on modeling, analysis and simulation of wireless and mobile systems (MSWiM '04)* (pp. 126–134). New York, NY: ACM.
6. Choi, J., Na, J., Park, K., & Kim, C. K. (2007). Adaptive optimization of rate adaptation algorithms in multi-rate WLANS. In *IEEE international conference on network protocols, 2007 (ICNP 2007).*, pp. 144–153.
7. Qiao, D., & Choi, S. (2005). Fast-responsive link adaptation for IEEE 802.11 WLANs. In *IEEE international conference on communications, 2005 (ICC 2005).*, Vol. 5, pp. 3583–3588.
8. Holland, G., Vaidya, N. H., & Bahl, P. (2001). A rate-adaptive mac protocol for multi-hop wireless networks. In *Proceedings of the 7th annual international conference on mobile computing and networking (MobiCom '01)*, Rome, Italy (pp. 236–251). New York, NY: ACM.
9. Zhu, H., & Cao, G. (2005). rDCF: A relay-enabled medium access control protocol for wireless ad hoc networks. In *INFOCOM 2005. 24th annual joint conference of the IEEE computer and communications societies*, Vol. 1, pp. 12–22.
10. Sadeghi, B., Kanodia, V., Sabharwal, A., & Knightly, E. (2002). Opportunistic media access for multirate ad hoc networks. In *Proceedings of the 8th annual international conference on mobile computing and networking (MobiCom '02)* (pp. 24–35). New York, NY: ACM.
11. Kamerman, A., & Monteban, L. (1997). WaveLAN II: A high-performance wireless LAN for the unlicensed band. *Bell Labs Technical Journal*, 118–133.
12. MadWifi, Multiband Atheros Driver for WiFi. http://www.madwifi.org/.
13. Bejerano, Y., Han, S., & Kumar, A. (2007). Efficient load-balancing routing for wireless mesh networks. *Computer Networks, 51*, 2450–2466.

14. Mhatre, V., Baccelli, F., Lundgren, H., & Diot, C. (2007). Joint MAC-aware routing and load balancing in mesh networks. In *ACM CoNEXT*.
15. Kohler, E., Morris, R., Chen, B., & Jannotti, J. (2000). The click modular router. *ACM Transactions on Computer Systems (TOCS), 18*(3), 263–297.
16. Draves, R., Padhye, J., & Zill, B. (2004). Comparison of routing metrics for static multi-hop wireless networks. In *ACM Sigcomm*.
17. Paris, S., Nita-Rotaru, C., Martignon, F., & Capone, A. (2011). EFW: A cross-layer metric for reliable routing in wireless mesh networks with selfish participants. In *INFOCOM, 2011*, pp. 576–580.
18. Qi, B., Shen, F., & Raza, S. (2012). iBATD: A new routing metric for multi-radio wireless mesh networks. In *2012 ninth international conference on information technology: New generations (ITNG)*, pp. 502–507.
19. Razak, S., Kolar, V., & Abu-Ghazaleh, N. B. (2010). A MAC interaction aware routing metric in wireless network. In *Proceedings of the 13th ACM international conference on modeling, analysis, and simulation of wireless and mobile systems (MSWIM '10)*, (pp. 118–121). New York, NY: ACM.
20. Koksal, C., & Balakrishnan, H. (2006). Quality-aware routing metrics for time-varying wireless mesh networks. *IEEE Journal on Selected Areas in Communications, 24*(11), 1984–1994.
21. Banerjee, S., & Misra, A. (2002). Minimum energy paths for reliable communication in multi-hop wireless networks. In *ACM Mobihoc*.
22. Dong, Q., Banerjee, S., Adler, M., & Misra, A. (2005). Minimum energy reliable paths using unreliable wireless links. In *Proceedings of the 6th ACM international symposium on mobile ad hoc networking and computing (MobiHoc '05)* (pp. 449–459). New York, NY: ACM.
23. Proakis, J. (2000). *Digital communications*. McGrawHill
24. Qiao, D., & Choi, S. (2001). Goodput enhancement of ieee 802.11a wireless lan via link adaptation. In *IEEE ICC*.
25. IEEE Standard for Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications. ISO/IEC 8802-11:2005/Amd.4:2006(E).
26. Tickoo, O., & Sikdar, B. (2004). Queuing analysis and delay mitigation in IEEE 802.11 random access MAC based wireless networks. In *IEEE Infocom*.
27. Engelstad, P., & Østerbø, N. (2006). Queuing delay analysis of IEEE 802.11e EDCA. In *3rd annual conference on wireless on-demand network systems and services*.
28. Sobrinho, J. L. (2001). Algebra and algorithms for qos path computation and hop-by-hop routing in the internet. *IEEE/ACM Transactions on Networking, 10*(4), 541–550.
29. Yang Y., & Wang, J. (2008). Design guidelines for routing metrics in multihop wireless networks. In *INFOCOM 2008. The 27th conference on computer communications. IEEE*, pp. 1615–1623.
30. Cormen, T., Leiserson, C., & Rivest, R. (2000). *Introduction to algorithms*. McGrawHill.
31. CM9 802.11 a/b/g wireless card. http://www.wneweb.com/.
32. Johnson, D., Maltz, D., & Broch, J. (2001). Dsr: The dynamic source routing protocol for multi-hop wireless ad hoc networks, chapter 5. In *Ad hoc networking*, pp. 139–172.
33. Iperf-tool. http://dast.nlanr.net/projects/iperf/.
34. Jain, R. (1991). *The art of computer systems performance analysis*. John Wiley and Sons.
35. The Click Modular Router Project. http://www.read.cs.ucla.edu/click/.

Author Biographies

Tae-Suk Kim received his MS degree and PhD in Industrial Engineering andTelecommunication Engineering Interdisciplinary Program from Korea AdvancedInstitute of science and Technology (KAIST), Daejeon, Korea in 2000 and 2005,respectively. He is currently working as a research staff for Samsung Advanced Institute of Technology (SAIT).From Aug. 2007 to Jul. 2009 he was a postdoctoral research associate in the Department of Computer Science and Engineering, University of California, Riverside (UCR). Priorto joining UCR, he was a postdoctoral research associate in the Illinois Network Designand EXperimentation (INDEX) Group in the Department of Computer Science atUniversity of Illinois at Urbana Champaign (UIUC). His research interests lie in resourcemanagement and cross-layer optimization of wireless networks.

Gentian Jakllari is a faculty member at the Telecommunication Dpt at the University of Toulouse, France. From 2008 to 2011 he was a Scientist at BBN Technologies in Cambridge, MA, USA. He got his bachelor's degree in computer science at the University of Ioannina, Greece and his M.Sc and Ph.D. at the University of California, Riverside. His research interests are in the field of computer networks with emphasis on wireless networks, access layer protocols, routing protocols, cooperative networks, cognitive radio networks and quality of service provisioning.

Srikanth V. Krishnamurthy received his Ph.D degree in electrical and computer engineering from the University of California at San Diego in 1997. From 1998 to 2000, he was a Research Staff Scientist at the Information Sciences Laboratory, HRL Laboratories, LLC, Malibu, CA. Currently, he is a professor of Computer Science at the University of California, Riverside. His research interests are primarily in wireless networks and security. Dr. Krishnamurthy is the recipient of the NSF CAREER Award from ANI in 2003. He was the editor-in-chief for ACM MC2R from 2007 to 2009. He is a Fellow of the IEEE.

Michalis Faloutsos is a faculty member at the Computer Science Dept. at the University of California, Riverside. He got his bachelor’s degree at the National Technical University of Athens and his M.Sc and Ph.D. at the University of Toronto. His interests include, Internet protocols and measurements, peer-to-peer networks, network security, BGP routing, and ad-hoc networks. With his two brothers, he co-authored the paper on power-laws of the Internet topology, which received the ACM SIGCOMM Test of Time award. His work has been supported by many NSF and military grants, for a cumulative total of more than $6 million. Several recent works have been widely cited in popular printed and electronic press such as slashdot, ACM Electronic News, USA Today, and Wired. Most recently he has focused on the classification of traffic and web-security, and co-founded a cyber-security company in 2008, offering services as www.stopthehacker.com, which received two SBIR grants from the National Science Foundation, and institutional funding in Dec 2011.