1

Quality-Aware Routing Metrics in Wireless Mesh Networks

Can Emre Koksal

Department of Electrical and Computer Engineering, The Ohio State University, Columbus, OH (email: koksal.2@osu.edu).

1.1 Introduction

In this chapter we address the problem of selecting good paths in networks made up of multiple wireless links[1], such as wireless mesh networks. By "good paths", we mean paths that both benefit individual data transfers (in terms of TCP connection throughput, for example), and which lead to high aggregate network capacity.

Finding good paths between nodes in a wireless network involves two steps:

1. *Assigning cost metrics to links and paths.*
2. *Disseminating routing information.*

The second step, route dissemination, has received much attention over the past decade. The link and/or path metrics need to be disseminated to the nodes in the network using a routing protocol, to help nodes select best paths in a distributed fashion. There are two types of protocols in how the route dissemination is done: proactive and reactive protocols.

Proactive protocols determine paths before there is any demand for communication. They calculate the routing tables ahead of time and maintain them through periodic update messages. Examples include Destination-Sequenced Distance Vector Routing (DSDV, [1]), Fisheye State Routing (FSR, [2]), and Optimized Link State Routing (OLSR, [3]).

Reactive protocols, on the other hand, do not calculate routes ahead of time. Route discovery follows the communication request. Examples of reactive protocols include Ad Hoc On Demand Distance Vector (AODV, [4]), Temporarily Ordered Routing Algorithm (TORA, [5]) and Dynamic Source Routing (DSR, [6]).

In this chapter we address the first issue, assigning cost metrics to links. Regardless of whether a protocol is proactive or reactive, it requires a mecha-

[1] We use the term "link" to refer to the communication channel between a pair of nodes.

nism to differentiate between different paths. This differentiation is done using cost metrics.

The cost metric of a link is the cost of forwarding a packet along the link. The problem of defining a cost metric is considerably harder in wireless networks than in traditional wired networks, because the notion of a "link" between nodes is not well-defined. The properties of the radio channel between any pair of nodes vary with time, and the reliable radio communication range is often unpredictable. The communication quality of a radio channel depends on background noise, obstacles and channel fading, as well as on other transmissions occurring simultaneously in the network. The appropriate cost metric must take into account a number of factors due to the vagaries of radio channels, which in turn makes the task of assigning metrics non-trivial. Moreover, it is desirable that the metrics for the links along a path be *composable*, so that the end-to-end cost of a path can be easily derived from the individual metrics of the links along the path.

We observe that the type of quality aware routing metric to be chosen depends on the physical layer being used. Designing and implementing a physical layer that can fully "hide" the vagaries of the radio channel from higher layers has proven to be difficult for a number of reasons. It requires the physical layer to be able to accurately estimate and adapt several parameters (e.g., transmit power, modulation, error control coding, etc.) to cope with channel conditions that vary rapidly in time. In fact, we know of no current or next-generation radios that propose to employ sophisticated techniques to fully handle channel quality issues at the physical layer, because of implementation complexity and the absence of practically useful codes that can perform well (especially in the non-asymptotic limit of finite packet sizes) across the large range of channel conditions that are observed in practice.

Indeed, practical wireless radios such as the ones based on the various IEEE 802 standards (e.g., 802.11, 802.15, etc.) employ only a simple coding strategy, mostly for error detection. Nodes transmit at one of a discrete set of power levels, and rely on a small number of link-layer packet retransmissions to overcome errors. All other packet losses are visible to higher layers, where they may be recovered using end-to-end mechanisms (such as TCP retransmissions or packet-level forward error correction implemented by applications). Most wireless mesh networks are radio networks comprised of radios similar to 802.11.

Another way modern radios (e.g., 802.11 chip-sets) cope with channel variations is the use of adaptive modulation schemes, allowing higher layers to set one of several possible bit rates. If frame loss rates at a particular bit rate rise, reducing the bit rate can reduce the observed frame loss ratio and improve throughput. Several bit rate adaptation schemes have been proposed (see [7] for a detailed treatment), and the topic remains an active area of work. We view bit rate selection as being complementary to quality-aware routing, in the sense that once the routing protocol picks the best neighbor to use for a destination using measured cost metrics, the link layer picks the best bit rate (modulation scheme) to use for that neighbor.

In the presence of bit rate adaptation, some routing metrics may need to be readjusted, and properly normalized with respect to the transmission rate. For instance, for many applications, a packet loss rate of 10% at 10 Mbps may be preferable over a 5% loss rate at 1 Mbps. Hence, a metric based solely on the loss rate should be modified to take the variety of available rates into account.

1.2 Routing Metrics for Wireless Mesh Networks

In this section we study seven cost metrics, discuss their relative benefits and shortcomings, and whether they would be appropriate for wireless mesh networks. These metrics are Hop Count, Per-hop Round Trip Time (RTT, [8]), Per-hop Packet Pair Delay (PktPair), quantized loss rate [9], Expected Transmission Count (ETX, [10]), modified ETX (mETX, [11]) and Effective Number of Transmissions (ENT, [11]).

1.2.1 Hop Count

The traditional approach to routing in ad hoc wireless networks is minimum-hop (shortest-path) routing (e.g., [1; 5]). The hop count is the simplest cost metric, and the simplicity of it may be attractive for networks for which mobility is high. Indeed, all other cost metrics require a link-level quantity to be measured or estimated and this process takes time, during which the same quantity may alter significantly. Consequently in mobile networks one may be forced to use the simple hop count, which requires minimal amount of measurement.

Although simple, minimum-hop routing inherently "quantizes" the state of a link into one of the two states, "up" or "down." In reality, the state of a wireless link is not in any one of the two states at any point in time. For instance, Fig. 1.1 illustrates the packet delivery ratio taken from a certain link in the Roofnet wireless mesh network [12]. Each node in the network has an 802.11b wireless card and an antenna. The transmission rate is set to a constant 11 Mbps. The delivery ratio was obtained by sending a sequence of 1500-byte broadcast packets, with the receiver keeping track of which packets were received successfully. The successful receipt or loss of a packet defines a binary random variable; each sample delivery ratio in the graph is the average of a window of 40 successive binary random variables. The window advances by 1 for each reported sample. Clearly, the loss rate is almost never 0 or 1, but most of the time it is in the "gray" area in between these two extremes.

It has also been illustrated in different platforms (e.g., [13]) that in the presence of link variability, which is a common phenomenon in wireless mesh networks, minimum hop fails to have a satisfactory performance.

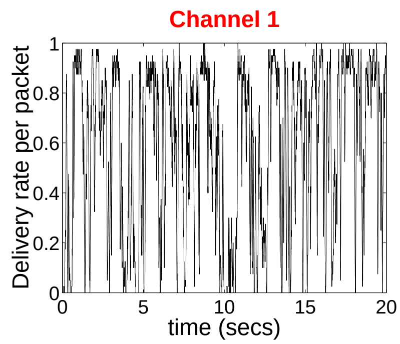


Fig. 1.1. Packet delivery rate for a link of Roofnet.

1.2.2 Per-hop Round Trip Time

[8] propose a delay-based link cost metric. This metric uses the measured average round trip time (RTT) seen by unicast probes between neighboring nodes. It is originally built as a part of a Multi-Radio Unification Protocol (MUP) -a channel assignment protocol for community networks-. Its application as a routing cost metric is implemented in [13].

To measure the channel, a probe packet is broadcast every 500 ms. Upon receiving a probe packet, each neighbor responds immediately, but in a non-preemptive manner. The acknowledgment contains a time-stamp so that the RTT can be calculated. The node keeps an exponentially weighted moving average (EWMA) of the RTT samples for each neighbor:

$$\text{RTT estimate}[n+1] = 0.1 \times \text{RTT}[n] + 0.9 \times \text{RTT estimate}[n],$$

which is a low pass filter with a bandwidth of a few packets. The RTT estimate of the link is then assigned as the cost for the link. This metric is composable, since the sum of the RTT estimates over two links in cascade is the RTT estimate for the two hop path.

The RTT cost metric contains several components contributing to the delay at a link.

- Queueing delay: Since the neighbors reply to the probe packets in a non-preemptive manner, the instantaneous RTT incorporates the time it takes for the existing jobs to be processed at a node.
- Channel quality: A packet may not be correctly decoded due to channel issues caused by fading or interference by other nodes not directly contending with our node. In this case, the packet is retransmitted up to a certain maximum number of times, contributing to the RTT calculation.
- Channel contention: If there are other nodes in the vicinity of one of the neighbors, the probe packet or the acknowledgment can get delayed due to direct contention. Contention can also be viewed as a channel issue

(an outage) caused by a nearby node causing an intolerable amount of interference.

All of the above factors are legitimate factors that should be taken into account when considering the cost of a link. Indeed, it is illustrated in [8] that in a 12 node network simulation with a real world web traffic model, the RTT metric is a reasonably well representative of the actual load at the nodes. Another set of simulations were run for a relatively lightly loaded network of 35 nodes, a small subset of which generates web traffic. When the RTT metric is used for channel assignment to pick the cleaner frequency for each hop, the network throughput increases by up to 70% and the average delay reduces by 50%.

However, there is a fundamental problem associated with using a routing metric, such as the RTT, which varies with varying load. It leads to either a highly oscillatory behavior or even instability. Specifically, suppose the delay at a certain node decreases due to reduced load at that node. Then, more and more of the paths tend to pass through this node, which will pull the delay, and hence the RTT metric back to a high value. The way the protocol is designed, such oscillations leading to route instability cannot be suppressed. The factors causing this kind of route instability is referred to as "self interference."

In [13], the RTT metric is experimentally analyzed in a 23 node network in which every node pair initiates a long TCP session. The median of the average throughputs of all the sessions may be 75% lower when RTT is used instead of the simple hop count (which achieves around 1100 Kbps). The authors also illustrate that this reduction is indeed due to self interference, since the optimal path assignments change about 20 times more frequently with RTT, compared to the hop count.

One needs to be careful in using delay related quantities as a cost metric because of the self interference phenomenon. One solution proposed is to use another metric, per hop packet pair delay (PktPair), which is based on a simple modification to the per-hop RTT metric. We study the PktPair metric in the next section. Some other issues associated with the RTT metric can be listed as:

- The overhead associated with measuring the RTT may be high.
- This metric implicitly accounts for the link rate (the transmission time is inversely proportional to the link rate), but when the queueing delay is large relative to the transmission time, the link rate becomes an insignificant portion of the metric. However, in a dense network, increased link rate is a much more important component of the system performance since the interference and duration of contention is reduced by an increased data rate. Hence the amount of RTT spent on air should be a more important portion of the link cost compared to that spent in a queue at a node. Any throughput based metric can be modified simply to take the link rate into consideration, but it is not as easy for a delay based metric.
- This metric does not respond to the channel variability at time scales shorter than tens of packets. Indeed, the instantaneous RTT is sampled

once every 500 msec and the resulting sequence is further low pass filtered with an EMWA filter. Thus, for a certain change to be effective in the route calculation, it should be sustained for an extended amount of time (5-6 seconds). The system is not responsive to the variations or bursty losses at time scales lower than that.

1.2.3 Per-hop Packet Pair Delay (PktPair)

PktPair is built by [13] in an effort to modify per hop RTT, which is shown to be problematic due to two issues. First one is the self interference and the second one is the relative significance of the queueing delay compared to the transmission time in the overall cost.

The idea of PktPair is based on sending a short probe packet ahead of a long one and using the short one to set a time reference. A small packet (of size 137 bytes) and a large one (1000 bytes) is sent in succession and each neighboring node keeps the time difference between the reception of these two packets. This value is fed back to the sender, which keeps an EWMA. This average is assigned as the cost metric for the link.

The measured difference between the times of reception of two successive packets includes potential delays due to contention for the medium with other nodes and the possible retransmissions due to channel issues caused by fading and other nodes communicating in the vicinity. Unlike the per hop RTT, PktPair does not have any component for the queueing and processing delay in it. This suppresses the route instability due to self interference to some extent. Indeed, the queueing and processing portion of an increase in delay do not contribute to an increase in the metric. However, an increase in contention still causes the metric to increase. Consequently, in a dense network with long term TCP flows, the average throughput increases (to 600 Kbps) by more than 100% and frequency of the change in the optimal path assignments reduce by about 50% compared to the RTT. Nevertheless the improvement is still not good enough for PktPair to outperform even the simple hop count metric.

Another issue associated with PktPair is the overhead, which is even higher than the overhead with per hop RTT.

1.2.4 Quantized Loss Rate

In [9] Yarvis *et al.* propose a routing metric that estimates the per-link frame delivery ratios and uses the end-to-end path loss probability as the cost of routing over a path. Since the increase in the load affects the metric only through the increased contention, the effects leading to self interference are suppressed as much as the PktPair. The implementation is done for the sensor network platform, therefore a large number of simplifications have been made to make it practical in the presence of limited computational power.

To measure the link quality, each node keeps track of the number of correctly received packets from each of its neighbors. In particular, a window of the most recent 32 packets is considered for each downlink and an average number of correctly decoded packets is calculated. This value is then quantized depending on the region it lies: Q_0: 53-100% loss, Q_1: 21-53% loss, Q_2: 10-21% loss and Q_3: 0-10% loss. The midpoint of each region (i.e., 75%, 35%, 15% and 5%) is assigned as the representative of the region. Each node keeps track of its uplink to every neighbor as well and records the higher one of the two quantized loss rates as the (bi-directional) cost of the link.

The quantized loss rate metric is composable. Even though the end to end loss rate is not equal to the sum of the individual loss rates, one can simply use the log function. Indeed, we can add the -log of the estimated delivery rate ($R_e = 1 -$ loss rate) of each link to get the log of the end-to-end delivery rate for the path. This simple modification is used in the actual algorithm. The following table summarizes the metric assignment process.

Quality	delivery rate	R_e	$-\log(R_e)$	cost metric
Q_3	90-100%	0.95	0.05	1
Q_2	79-90%	0.85	0.16	3
Q_1	47-79%	0.65	0.43	8
Q_0	0-47%	0.25	1.39	28

This metric is tested over DSDV in a sensor network platform, and its performance is compared with that of the plain DSDV, for which the hop count is the cost metric. For 28 nodes, the quantized loss rate metric reduced the network wide loss rate by a percentage between 24-32%. For increased number of nodes, the amount of improvement decreases (e.g., for a 48 node network, percent improvement is between 6-20% and for a 91 node network it is between 2-4%). The authors argue that a good portion of this reduction in improvement might be due to the limitation of computational resources in the sensor nodes. Specifically, an increased number of nodes may be leading to an overflow in the neighbor lists, causing them to become ineffective. Note that in wireless mesh networks, the lack of resources is less of an issue and the reduced improvement may be less significant.

Another issue about this metric is that it does not account for the total bandwidth consumed, because it will prefer two links of low loss rate over a single link with higher loss-rate. When link-layer retransmissions are used, the one-hop path may be able to deliver the packet without as many total transmissions as the two-hop path. In fact ETX is motivated by this observation.

1.2.5 The Expected Transmission Count (ETX)

ETX is a metric proposed by [10] for 802.11-based radios employing link-layer retransmissions to recover from frame losses. Basically, the ETX of a radio link is the estimated average number of {data frame, ACK frame} transmissions necessary to transfer a packet successfully over the wireless link. In ETX, each node estimates the frame loss ratio p_f to each of its neighbors over a recent time window, and obtains an estimate p_r of the reverse direction from its neighbor. These loss estimates are obtained using broadcast probe packets (that are not retransmitted) at the link layer once every second. The estimate for p_f and the p_r is respectively the fraction of the probes and the acknowledgments correctly decoded in the last ten seconds. The node then calculates the *expected transmission (ETX) count* for the link between the neighbor as $\frac{1}{(1-p_f)(1-p_r)}$. The ETX metric is composable, since the expected value of total number of transmissions over a path is the sum of the individual expected number of transmissions of the links along the path. In the presence of bit rate adaptation, the only modification required for ETX is to use the Expected Transmission *Time* (rather than Count) as the metric [14], because a lower bit rate ends up using the channel for a longer period of time.

The number of transmissions of a packet on a radio link is an appealing cost metric because minimizing the total number of transmissions maximizes the overall throughput. It was shown in [13] that the ETX metric improves the average throughput of the TCP flows in the 23 node network (to 1357 Kbps) by 23.1% over the hop count metric. Also the frequency of the changes in the calculated optimal paths is only 3 times as much as the hop count, which implies that the effects leading to self interference are mostly suppressed. This is expected since the link level retransmissions depend only on the link level packet errors caused by channel issues. The channel issues are almost completely independent of the load at a node.

Although the experimental results show that ETX performs better that traditional shortest path routing under static network conditions, it may perform poorly under highly variable channel conditions and burst-loss situations. Indeed, the ETX of the link is the reciprocal of the (estimated) probability of correct packet delivery. This definition implies that the probability of delivery of distinct packets is assumed to be an independent and identically distributed process, and hence the number of transmissions per packet has a geometric distribution. If successive packets were lost independently with probability equal to the average packet error rate of the channel, the assumption would be accurate. Packet losses generally occur in bursts, however, and the packet loss probability is usually variable and correlated.

Consider for example the traces in Fig. 1.2 taken from four distinct links in the Roofnet (the first trace was already given in Fig. 1.1 and the method of obtaining the traces was explained back there). Each of these four links has an ETX of approximately 2 during the testing period. Therefore, if ETX is taken as the metric for quality, these four links are identical. On the other hand, the sample variances of the delivery ratios are quite different for these links, i.e., these wireless links have similar long term average behaviors, even though their short-term behaviors are quite different. Indeed, the sample coefficient of variation for the binary packet error sequences are 7.92, 2.16, 1.20 and 0.61.

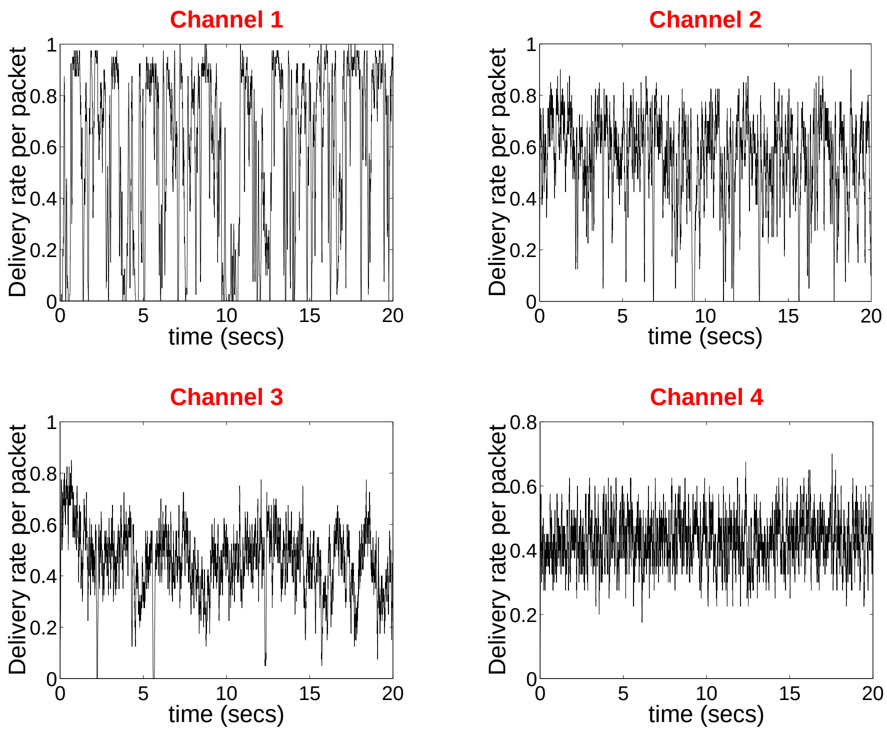


Fig. 1.2. Packet delivery rate for four distinct links of Roofnet.

One may ask whether it is possible to increase the frequency of ETX measurements and change the optimum paths accordingly more and more frequently until the "remaining" variability between updates is somewhat insignificant. Unfortunately, the update procedure involves significant amount of overhead in the network. If repeated frequently, it causes inefficient use of resources, extra interference and even instability of the routing algorithm. Therefore, the time-scale over which path-selection decisions are made is typically no less than tens or hundreds of packets; i.e., once a path between two nodes has been selected, it is likely to remain for several seconds. As shown in Fig. 1.2, there may be a huge channel variability over that time-scale and the ETX has to live with that.

In [15] Koksal *et al.* showed that the variability in short, as well as the longer time scales has a significant impact on the expected number of transmissions. It is further shown in [11] that, given two links in Roofnet, it not uncommon that the link with a lower ETX metric may in fact lead to a *higher* observed loss rate at the transport layer. The main reason for this is that good link-layer protocols do not try to retransmit lost packets forever but give up after a threshold number of attempts. When losses occur in bursts, picking the link in the middle of a burst-error situation would be bad *even* if it had a lower ETX.

To summarize, ETX can improve the throughput of a wireless mesh network by a significant amount compared to the hop count cost metric. However, ETX metric cannot track the variability of the channel at short time scales due to potential route instability.

1.3 The Modified Expected Number of Transmissions (mETX)

This metric is built to overcome the shortcomings of ETX in the presence of channel variability. The development is based on a certain characterization of the channel given in [15]. There, authors develop tools to analyze the channels with non-iid losses and quantify the impact of channel variability on the number of transmissions. This leads to the mETX metric, proposed in [11].

The model assumes that the bit error probability on a link is a (non-iid) stationary stochastic process. The variability of the link is modeled using the statistics of this stochastic process. Then, the mean number of transmissions is analytically calculated and the results show that it can be closely approximated with the first two order statistics of the bit error probability, summed over a packet duration. For mETX, the critical time scale for the link variability is the transmission time of a single packet including all its retransmissions.

The mETX metric is a function of the mean, μ_Σ and the variance, σ_Σ^2 of Σ, the bit error probability summed over a packet duration:

$$\text{mETX} = \exp\left(\mu_\Sigma + \frac{1}{2}\sigma_\Sigma^2\right) \tag{1.1}$$

The μ_Σ term represents the impact of slowly varying and static components in the channel (e.g., shadowing, slow fading), while the σ_Σ^2 represents the impact of relatively rapid channel variations (e.g., flat fading, interference) that the μ_Σ term (and hence the ETX) cannot track.

To estimate these two parameters, bit level information is necessary. Counting only the packet losses is not sufficient; thus, probe packets with a known content are used for estimation. The parameters μ_Σ and σ_Σ^2 are estimated by considering the number of erred bits in each probe packet. As in the ETX metric, each node sends probe packets periodically to calculate a loss rate sample and this information is passed to a moving average filter. Alternatively an EWMA filter can be used.

In [11], results of link measurements taken from 57 links that belong to distinct pairs of 12 different nodes of the Roofnet testbed are illustrated at a transmission rate of 11 Mbps. Based on the measurements, it was shown that the packet loss probability has a higher correlation coefficient ($\rho = 0.85$) with σ_Σ^2 than it has with μ_Σ ($\rho = 0.59$). Consequently the link variability can be even more relevant than the ETX for packet losses. Also, by combining the impact of variability and the average loss rate, mETX achieves a drop of between 7%-50% in the average network loss rate (corresponds to an improvement of up to 60% in TCP throughput). The amount of reduction varies with the number of nodes and the node density.

The main drawback of the mETX metric is the complexity of the channel estimation. Firstly, the probe packets need to be processed at the bit level. This may not necessarily be an issue for the mesh networks due to the relative abundance of processing power, however may be problematic for other platforms such as sensor networks. Secondly, the variance component, σ_{Σ}^2 increases with increased estimation error. Namely, a link may have a high mETX metric due to not only the high channel variability, but also the estimation error. Consequently, a better link with a high estimation error may end up having a higher metric than a worse link. On the other hand, one can justify the fact that a links with more degraded information are less preferable, using the famous quote: "the shortest way home is the way you know."

In the same way as ETX, the mETX can be adapted easily for radios that provide bit rate adaptation by normalizing the metric with respect to the transmission rate.

1.4 The Effective Number of Transmissions (ENT)

The motivation for the ENT metric is to find routes that satisfy certain higher-layer protocol requirements. The challenge is finding a path that achieves high network capacity while ensuring that the end-to-end packet loss rate visible to higher layers (such as TCP) does not exceed a specified value. Given a loss constraint, picking the path that maximizes the link layer throughput may not be sufficient, because it may involve links with high loss rates. Because link-layer protocols give up after a certain threshold number of retransmissions (M), ETX and mETX may pick links that violate the loss rate requirement visible to higher layers. The ENT metric is designed to meet the desired goal.

Similar to the mETX metric, the ENT metric also characterizes the probability of bit error as a stationary stochastic process. Using a large deviations approach, it is shown in [11] that the probability of a packet loss (i.e., number of transmissions exceeding M) can be well approximated with

$$P_{\text{loss}} \approx \exp\left[-\frac{1}{2}\left(\frac{\log M - \mu_{\Sigma}}{\sigma_{\Sigma}}\right)^2\right], \tag{1.2}$$

for large packet sizes and large values of M. Now suppose the desired loss probability is P_{desired} and let $\delta = -\log P_{\text{desired}}/\log M$. There is a one-to-one correspondence between the desired loss rate and δ. Thus the parameter δ uniquely specifies P_{desired}. Note that δ is referred to as the *temporal diversity* *gain* in wireless communication. For a given P_{desired} (i.e., δ) to be met, $P_{\mathrm{loss}} \leqslant P_{\mathrm{desired}}$ and consequently,

$$\mu_{\Sigma} + 2\delta\sigma_{\Sigma}^{2} \leqslant \log M. \tag{1.3}$$

The sum in on the left side of Condition (1.3) is defined as the log effective number of transmissions (i.e., $\log \mathrm{ENT}$) of the link.

One way to interpret Condition (1.3) is as follows. Suppose the higher layer does not specify any loss probability constraint, i.e., $\delta = 0$. Condition (1.3) turns into a comparison of μ_{Σ} (hence the average bit error probability of the channel) with M. Thus, the higher-layer requirement turns into a condition involving average link parameters only, as is the case with ETX. Now suppose the higher-layer has a loss rate requirement, i.e., $\delta > 0$. In that case one needs to *underbook* the resources to meet the loss probability target. The amount of spare ETX that has to be put aside in order to accommodate channel fluctuations is $2\delta\sigma_{\Sigma}^{2}$. This margin allows the packet loss probability target to be met. As expected, this amount is directly related to the variability, σ_{Σ}^{2}, of the channel and the strictness, δ, of the loss rate requirement. This interpretation of ENT is analogous to the notion of *effective bandwidth*, which was developed to model variable traffic sources in queueing networks. Indeed, ENT can be interpreted as the effective bandwidth of the discrete stochastic process, the number of transmissions.

ENT has a structure similar to mETX. The main difference is the extra degree of freedom due to the factor 2δ. Indeed, the mETX is the ENT evaluated at $\delta = 1/4$. Similar to the mETX, a by-product of ENT is to reduce the packet loss ratio observed by higher-layer protocols, *after* any link-layer retransmissions are done. Also, since exactly the same parameters are used in the ENT as in the mETX, the channel estimation procedure is identical.

On the other hand, the ENT metric is not additive as the ETX or the mETX. The metric is composed over successive links using minimax type routing algorithms. More precisely, among all the paths between two nodes, the path along which the links minimizes the maximum ENT is selected as the best route. Another algorithm that combines the ETX and the ENT metrics is proposed in [11]:

"For each link, compute its log ENT. Compare against $\log M$. Assign a cost of ∞ to the links that have log ENT $> \log M$ and assign a cost of ETX to the others. Between any pair of nodes use the path that minimizes the total cost." This algorithm focuses only on the feasible links, i.e., the ones that satisfy the application loss requirement, P_{loss}. It picks those with the minimum ETX among those.

The average network loss rate is also simulated with the link-level data acquired from the Roofnet. The set of feasible links are defined to be those that have an ENT of less than 16 for the δ parameter varied between 1 and 2.5. There were some interesting trends. First, the observed loss rates can be controlled by merely adjusting the "space parameter" δ, which acts as a knob to control the performance. Not only it is guaranteed that each link has no more than a certain desired loss rate, but also the average network loss rate can be reduced by an amount between 7-55% depending on the network size and the control parameter δ.

There is a catch, though. The loss rate does not decrease monotonically with increasing δ. Beyond a certain threshold, the loss rate starts to increase. The reason for this transition is that too many links are eliminated for violating the loss constraint. Consequently, even many "decent" links are gone and no feasible paths remain between some node pairs and the network becomes disconnected.

Another benefit of ENT is that it can be calibrated. A network architect can adjust the δ parameter until the desired network performance is achieved. Indeed, the derivations in [11] are based on certain assumptions, which can be partly violated in different platforms and environments. It is useful to have a degree of freedom for the necessary adjustments.

The main drawback of the mETX metric is valid for the ENT as well. Since the same channel estimation procedure is followed, the estimation error affects the ENT metric similar to the mETX metric.

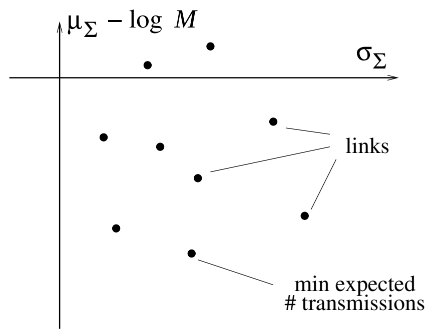
1.5 Geometric Interpretation of Routing Cost Metrics

This section provides a unified geometric framework that combines the mean and standard deviation of the bit error rate process to visually compare the (quantized) loss rate, ETX, mETX and the ENT metrics.

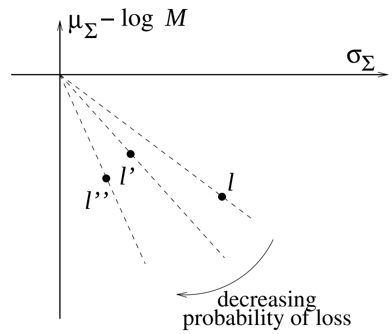
Let us represent a wireless link by two parameters, μ_Σ and σ_Σ. Each link corresponds to a point in the coordinate space $(\sigma_\Sigma, \mu_\Sigma)$ as illustrated in Fig. 3(a). In these graphs we use $\mu_\Sigma - \log M$ instead of μ_Σ as the y-axis. This only introduces a linear shift, but simplifies our discussions. In this space, the point with the lowest ordinate value is the one that minimizes the ETX. Such links will be preferred by routing algorithms that employ μ_Σ as the link cost metric (e.g., ETX).

For any given point, the slope of the line connecting the origin to that point is $(\mu_\Sigma - \log M)/\sigma_\Sigma$. Combining this with Eq. (1.2), points with smaller slopes, i.e., points with larger $|(\mu_\Sigma - \log M)/\sigma_\Sigma|$ have lower loss probabilities. For instance, in Fig. 3(b), link l has a higher loss probability (and hence a higher Quantized Loss Rate) than link l'. If the objective is to minimize the probability of loss, then the path selection algorithm should choose points with large $|(\mu_\Sigma - \log M)/\sigma_\Sigma|$ ratios.

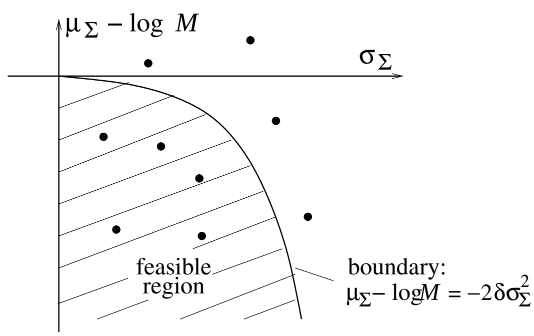
The set of points with a certain diversity gain, i.e., for a given δ, the links that satisfy $\delta = -\log P_{\text{loss}}/\log M$ lie on a parabola as shown in Fig. 3(c). Thus, the points outside the shaded region have a lower diversity gain and fail to satisfy the required constraint for P_{loss}. The shaded region can therefore be regarded as a *feasible region*. Notice that for $\delta = 0$ (i.e., no loss-rate requirement) the feasible region is the entire fourth quadrant. The region shrinks as δ is increased since the boundary of the region becomes more and



(a) Each point in $(\sigma_\Sigma, \mu_\Sigma)$ coordinate space represents a link.



(b) The slopes of the dashed lines are representatives of the loss probability.



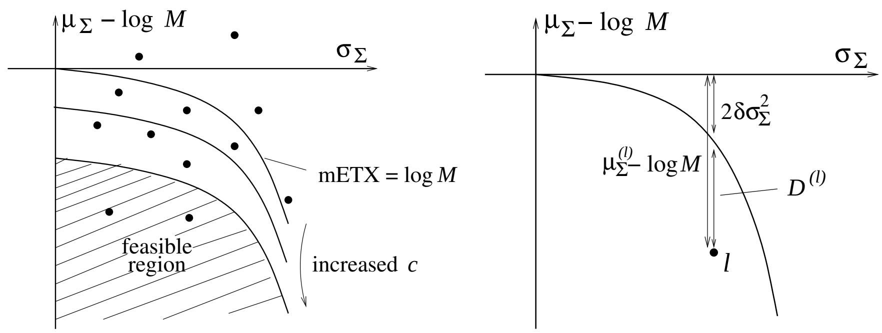
(c) The points in the feasible region satisfy $P_{\text{loss}} \leqslant \exp(-\delta(\mu_\Sigma - \log M))$.

Fig. 1.3. The geometry of the channel parameters and the loss probability.

more concave. Hence, a smaller number of links become feasible. For instance consider the routing algorithm, which minimizes the ETX subject to an ENT constraint. It should pick the links with small ordinate values among the points in the feasible region.

Similarly, the set of links with a constant mETX constitute a parabola in the coordinate space $(\sigma_\Sigma, \mu_\Sigma)$. Indeed, the set of points with mETX equal to the constant c lie on the parabola specified by

$$\mu_\Sigma + \frac{1}{2}\sigma_\Sigma^2 = c.$$



(a) The set of points with mETX=c form the parabola $\mu_\Sigma + \frac{1}{2}\sigma_\Sigma^2 = c$. The points with smaller mETX lie inside the parabola.

(b) The vertical distance between the link l and the boundary of the feasible region is $D^{(l)} = \log M - \log$ ENT.

Fig. 1.4. The geometry of the mETX and the ENT.

These parabolas can also be viewed as the boundaries for a feasible region, where the feasible links are those with mETX less than some given c value. Constant mETX curves are illustrated in Fig. 4(a). As c is reduced, the boundary moves farther away from the x-axis and consequently the set of points with smaller mETX shrink.

Finally, consider the vertical distance, $D^{(l)}$, between any admissible link $l : (\sigma_\Sigma, \mu_\Sigma)$ and the boundary of a feasible region of links. As illustrated in Fig. 4(b),

$$\begin{aligned} D^{(l)} &= -(\mu_\Sigma - \log M) - 2\delta\sigma_\Sigma^2 \\ &= \log M - \mathrm{ENT}^{(l)}(\delta). \end{aligned} \tag{1.4}$$

Hence, the feasible link that maximizes the vertical distance to the boundary of the feasible region is the one that minimizes the ENT. This means that, given an increase in the expected number of transmissions, the link with a small ENT is more likely to remain in the admissible region. Thus, if the objective is robustness with respect to the uncertainty in the measured parameters and to changes in the expected number of transmissions, the routing algorithm should choose points with smaller ENT.

Conclusion

In this chapter we studied seven routing cost metrics to be used for selecting good paths in wireless mesh networks. The following table summarizes these metrics, their benefits and drawbacks.

Metric	**Definition**	**Benefit**	**Drawback**
Hop Count	# hops	simplicity	chooses poor links
Per hop RTT	delay/hop	incorp. multiple factors	self interference
PktPair	transmit delay/hop	reduces self interference	high overhead
Loss Rate	packet loss rate	eliminates lossy links	low bandwidth paths
ETX	# transmissions	improves throughput	fails under variability
mETX	ETX w/variability	works w/variable links	sensitive to est. errors
ENT	eff. bandw. of link	provides controlled QoS	not composable solely

There are a couple of directions along which the routing metrics can be studied further. One paradigm for routing in wireless networks is cooperative diversity. Cooperative diversity takes advantage of broadcast transmission to send information through multiple relays concurrently. Similar to the traditional routing protocols, cooperative schemes also require a differentiation mechanism among different links, which makes the cost metrics necessary. For instance in ExOR [16], once a packet is transmitted over a hop, it may be decoded correctly by a number of other nodes as well as the intended next hop. After the transmission, a priority ordering of such nodes is made to decide who will relay the packet next. This ordering is based on the total cost of different paths from each node that has a copy of the packet to the ultimate destination.

New metrics can be engineered specifically for cooperative communication as well as the multipath routing setting, in which data between a pair of nodes can be carried over multiple paths simultaneously. In the multipath scenario, the composability of a metric becomes critical not only along each path, but also over parallel paths.

Another extension can be to build metrics based on physical layer parameters such as the signal-to-noise ratio (SNR). This can shed further light on the impact of physical layer on optimal routing decisions. It can also bring the channel estimation in line with the common practice of using physical layer pilot symbols to estimate the channel gain.

Finally, studying the impact of adaptive coding and power control on routing is critical for WiFi and 802.11 based networks, since rate adaptation is an integral part of these standards.

References

[1] C. E. Perkins and P. Bhagwat, "Destination Sequenced Distance Vector Routing," in *Proceedings of SIGCOMM*, 1994.
[2] M. Gerla, X. Hong, and G. Pei, "Fisheye State Routing Protocol for Ad Hoc Networks," in *Internet Draft, draft-ietf-manet-fsr-03.txt*, 2002.
[3] P. Jacquet, P. Muhlethaler, A. Qayyum, Laouiti A, L. Viennot, and T. Clausen, "Optimized Link State Routing Protocol," in *Internet Draft, draft-ietf-manet-olsr-07.txt*, 2002.

[4] C. E. Perkins and E. M. Royer, "Ad Hoc On Demand Distance Vector Routing," in *Proceedings of IEEE WMCSA*, 1999.

[5] V. D. Park and M. S. Corson, "A Highly Adaptive Distributed Routing Algorithm for Mobile Wireless Networks," in *Proc. of the IEEE Infocom*, 1997.

[6] D. B. Johnson, D. A. Maltz, Y. Hu, and J. G. Jetcheva, "The Dynamic Source Routing Protocol for Mobile Ad Hoc Networks (DSR)," in *Internet Draft, draft-ietf-manet-dsr-07.txt*, 2002.

[7] John C. Bicket, "Bit-rate selection in wireless networks," M.S. thesis, MIT, Feb 2005.

[8] A. Adya, Paramvir Bahl, Jitendra Padhye, Alec Wolman, and Lidong Zhou, "A Multi-Radio Unification Protocol for IEEE 802.11 Wireless Networks," Tech. Rep. MSR-TR-2003-44, Microsoft Research, July 2003.

[9] M. Yarvis, W. Conner, L. Krishnamurthy, J. Chhabra, B. Elliott, and A. Mainwaring, "Real World Experiences with an Interactive Ad-hoc Sensor Network," in *Proceedings of the International Workshop on Ad Hoc Networking*, August 2002.

[10] D. DeCouto, D. Aguayo, J. Bicket, and R. Morris, "A High Throughput Path Metric for Multi-Hop Wireless Routing," in *Proceedings of MobiCom*, 2003.

[11] C. E. Koksal and H. Balakrishnan, "Quality Aware Routing Metrics for Time-Varying Wireless Mesh Networks," *IEEE Journal on Selected Areas in Communications*, vol. 24, pp. 1984–1994, 2006.

[12] D. Aguayo, J. Bicket, S. Biswas, G. Judd, and R. Morris, "Link-level Measurements from an 802.11b Mesh Network," in *Proceedings of ACM SIGCOMM*, September 2004.

[13] R. Draves, J. Padhye, and R. Patra, "Comparison of Routing Metrics for Static Multi-Hop Wireless Networks," in *Proceedings of SIGCOMM*, 2004.

[14] D. DeCouto, *High-Throughput Routing for Multi-Hop Wireless Networks*, Ph.D. thesis, MIT, June 2004.

[15] C. E. Koksal, K. Jamieson, E. Telatar, and P. Thiran, "Impacts of Channel Variability on Link-Level Throughput in Wireless Networks," in *Proceedings of ACM SIGMETRICS/Performance*, 2006.

[16] S. Biswas and R. Morris, "ExOR: Opportunistic Multi-Hop Routing for Wireless Networks," in *Proceedings of ACM SIGCOMM*, August 2005.