(Wireless) Internet Routing

What is a wireless “link”? 
A story of link metrics for wireless networks
Learning From Deployments

- Early worked focused traditional routing issues
  - Control plane: topology management, neighbor discovery
  - Data plane: forwarding, avoiding loops
- Emphasis on simulation
- Early wireless hardware: 1 or 2 rates, quite robust and reliable
- And then people started deploying experimental research networks
  - MIT Roofnet
  - Microsoft research
  - ...
Reminder: What is Routing?

Control Plane:
- Topology discovery
  - Neighborhood discovery
- Route discovery and maintenance
  - Fault recovery

Data Plane:
- Forwarding
Topology and Route Discovery: What Does it Mean?

- **Topology:**
  - Nodes and links

- **Route discovery:**
  - How to find routes?
  - How to choose between different routes available? How to choose the best available route?

Links and route metrics
Link and Route Metrics
Components of a Routing Metric

- **Link Metric**: Assign a weight to each link

- **Path/Route Metric**: Combine metrics of links on path
Outline

- Routing metric
- Challenges in design of wireless routing metrics
  - Asymmetric links
  - Temporal variability
  - Different data rates and packet sizes
  - Wireless interference
  - Estimator design
Hop Count Routing Metric

- One possibility is to assign each link a metric of 1 (hop-count based routing)

- Problems:
  - Maximizes the distance traveled by each hop
    - Low signal strength $\rightarrow$ high loss ratio
  - May use a higher TxPower $\rightarrow$ interference
  - Different links have different qualities
Hop Count: Good or Bad Metric?

- Intuition from wired networks is wrong
  - Links share spectrum
  - Capacity penalty for more hops

\[ \text{Throughput} = 1 \]
\[ \text{Throughput} = \frac{1}{2} \]
\[ \text{Throughput} = \frac{1}{3} \]

- What should we optimize?
  - Per-route throughput, network capacity, power?
Same Hop Count, Different Throughput

- Setup: Throughput between one pair of nodes (best 8 routes)
### Performance of Hop Count

**x axis:** throughput

**y axis:** fraction of pairs with less throughput

- **Setup:** 100 random node pairs, throughput CDF, DSDV with min. hop-count vs “best” route

**Run R1:** 1 mW, 134-byte packets

---

A High-Throughput Path Metric for Multi-Hop Wireless Routing, De Couto et al., 2003

11
Motivation for a Better Metric

(a) Pairwise delivery ratios at 1 mW

(b) Pairwise delivery ratios at 30 mW

Distribution of link loss ratio
- Many links are asymmetric, wide range of loss ratios
Link and Route Metrics Example

- **Link metric = delivery radio:** A sends 100 packets to B. Since B received all 100, link quality = 100%.

![Diagram showing delivery ratio]

Delivery ratio = 100%
Route Metric Examples

Delivery ratio = 100%

Bottleneck link:

A-B-C = 50%
A-D-C = 51%

End-to-end delivery ratio:

A-B-C = 100%*50%=50%
A-D-C = 51%*51%=26%
Actual Throughput

Delivery ratio = 100%

Actual throughput:

- A-B-C : ABBABBBABBB = 33%
- A-D-C : AADDAADD = 25%
Link Metric: Straw Men

- Discard links with loss rate above a threshold?
  - Risks disconnecting nodes!

- Product of link delivery ratios as probability of end-to-end delivery?
  - Ignores inter-hop interference: prefers 2-hop route with 0% loss over 1-hop with 10% loss, when latter is nearly double the throughput

- Throughput of highest-loss link on path (bottleneck)?
  - Also ignores inter-hop interference
Can we Use the SNR? One RoofNet Link Over 24 Hours

- Cannot use Prism S/N ratio to predict link quality
**Metric: Expected Transmission Count**

Minimize total transmissions per packet  
(ETX, 'Expected Transmission Count')

Link throughput $\approx \frac{1}{\text{Link ETX}}$

<table>
<thead>
<tr>
<th>Delivery Ratio</th>
<th>Link ETX</th>
<th>Throughput</th>
</tr>
</thead>
<tbody>
<tr>
<td>100%</td>
<td>1</td>
<td>100%</td>
</tr>
<tr>
<td>50%</td>
<td>2</td>
<td>50%</td>
</tr>
<tr>
<td>33%</td>
<td>3</td>
<td>33%</td>
</tr>
</tbody>
</table>
Capturing Bi-directionality of a Link

- Assuming link-layer acknowledgments (e.g.: 802.11) and retransmissions:
  - $P(\text{TX success}) = P(\text{Data success}) \times P(\text{ACK success})$

Bi-directional Link Quality = \[ \frac{1}{p_{\text{forward}} \times p_{\text{reverse}}} \]

De Couto et al, Woo et al
Route ETX

Route ETX = Sum of link ETXs

<table>
<thead>
<tr>
<th>Route ETX</th>
<th>Throughput</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>100%</td>
</tr>
<tr>
<td>2</td>
<td>50%</td>
</tr>
<tr>
<td>2</td>
<td>50%</td>
</tr>
<tr>
<td>3</td>
<td>33%</td>
</tr>
<tr>
<td>5</td>
<td>20%</td>
</tr>
</tbody>
</table>
ETX Comments

- **Advantages:**
  - Better estimate of link quality and interference than hop count.

- **Problems:**
  - ETX assumes that all radios run at the same bit rate and all packets are of the same size.
    - 802.11b rates: \{1, 2, 5.5, 11\} Mbps
  - Assumes also the same power
  - ETX is a worst case measure and assumes that no spatial reuse is possible.
ETX: Summary

- **Link ETX**: predicted number of transmissions
- **Path ETX**: sum of link ETX values on path
- Calculate link ETX using *forward* and *reverse* delivery ratios
- To avoid retry, *data packet* and *ACK* must succeed
- \( \text{ETX} = \frac{1}{(d_f \times d_r)} \)
  - \( d_f \) = forward delivery ratio (data packet)
  - \( d_r \) = reverse delivery ratio (ACK packet)
Metric: Expected Transmission Time (ETT)

- Link loss rate = $p$
  - Expected number of transmissions
    \[ \text{ETX} = \frac{1}{1-p} \]
- Packet size = $S$, Link bandwidth = $B$
  - Each transmission lasts for $S/B$
    \[ \text{ETT} = \left( \frac{S}{B} \right) \times \text{ETX} \]
- Lower ETT implies better link
ETT: Illustration

11 Mbps
5% loss

18 Mbps
50% loss

1000 Byte Packet

ETT : 0.77 ms

ETT : 0.89 ms
Path Metric using ETT

- Add ETTs of all links on the path
- Use the sum as path metric

SETT = Sum of ETTs of links on path

(Lower SETT implies better path)
How Can You Get Diversity (to Decrease Inter-Hop Interference)

- Use multi-radio and/or multi-channel setup
  - 802.11b:
    - 20 MHz channel, 5 MHz spacing
    - 11 channels
    - 3 “orthogonal”
  - 802.11a
    - 20 MHz channels, 20 MHz spacing
    - Around 20 “orthogonal” channels
SETT does not favor channel diversity

<table>
<thead>
<tr>
<th>Path</th>
<th>Throughput</th>
<th>SETT</th>
</tr>
</thead>
<tbody>
<tr>
<td>Red-Blue</td>
<td>6 Mbps</td>
<td>2.66 ms</td>
</tr>
<tr>
<td>Red-Red</td>
<td>3 Mbps</td>
<td>2.66 ms</td>
</tr>
</tbody>
</table>
Capturing Channel Diversity

- Group links on a path according to channel
  - Links on same channel interfere

- Add ETTs of links in each group

- Find the group with largest sum.
  - This is the “bottleneck” group
  - Too many links, or links with high ETT (“poor quality” links)

- Use this largest sum as the path metric
  - Lower value implies better path

“Bottleneck Group ETT” (BG-ETT)
**BG-ETT Example**

<table>
<thead>
<tr>
<th>Path</th>
<th>Throughput</th>
<th>Blue Sum</th>
<th>Red Sum</th>
<th>BG-ETT</th>
</tr>
</thead>
<tbody>
<tr>
<td>All Red</td>
<td>1.5 Mbps</td>
<td>0</td>
<td>5.33 ms</td>
<td>5.33 ms</td>
</tr>
<tr>
<td>1 Blue</td>
<td>2 Mbps</td>
<td>1.33 ms</td>
<td>4 ms</td>
<td>4 ms</td>
</tr>
<tr>
<td>Red-Blue</td>
<td>3 Mbps</td>
<td>2.66 ms</td>
<td>2.66 ms</td>
<td>2.66 ms</td>
</tr>
</tbody>
</table>

BG-ETT favors high-throughput, channel-diverse paths.
BG-ETT does not favor short paths

<table>
<thead>
<tr>
<th>Path</th>
<th>Throughput</th>
<th>Blue Sum</th>
<th>Red Sum</th>
<th>BG-ETT</th>
</tr>
</thead>
<tbody>
<tr>
<td>3-Hop</td>
<td>2 Mbps</td>
<td>0</td>
<td>4 ms</td>
<td>4 ms</td>
</tr>
<tr>
<td>4-Hop</td>
<td>2 Mbps</td>
<td>4 ms</td>
<td>4 ms</td>
<td>4 ms</td>
</tr>
</tbody>
</table>
WCETT: Weighted Cumulative ETT

- SETT favors short paths
- BG-ETT favors channel diverse paths

Weighted Cumulative ETT (WCETT)

\[ WCETT = (1-\beta) \times \text{SETT} + \beta \times \text{BG-ETT} \]

\( \beta \) is a tunable parameter

Higher value: More preference to channel diversity
Lower value: More preference to shorter paths
Example of WCETT

<table>
<thead>
<tr>
<th>Path</th>
<th>Sum</th>
<th>Max</th>
<th>WCETT ((\beta = 0.9))</th>
<th>WCETT ((\beta = 0.1))</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>27</td>
<td>22</td>
<td>22.5</td>
<td>26.5</td>
</tr>
<tr>
<td>2</td>
<td>33</td>
<td>22</td>
<td>23.1</td>
<td>31.9</td>
</tr>
<tr>
<td>3</td>
<td>34</td>
<td>20</td>
<td>21.4</td>
<td>32.6</td>
</tr>
<tr>
<td>4</td>
<td>8</td>
<td>8</td>
<td>8</td>
<td>8</td>
</tr>
</tbody>
</table>
A Good Link Estimator

- Accurate
- Agile yet stable
- Small memory footprint
- Simple

Woo et al
ETX: Measuring Loss Rates

- Periodically send broadcast probe packets of fixed size
- All nodes know sending rate of probes
- All nodes compute loss rate based on how many arrive per measurement interval
- Nodes enclose loss measurements in their probes (B tells A loss from $A \rightarrow B$)
Expected Transmission Count (ETX)

- In theory: 
  \[ ETX = \frac{1}{d_f \times d_r} \]
  - \( d_f \): forward delivery ratio
  - \( d_r \): reverse delivery ratio
  - Assumptions: fixed rate, fixed power, ignore packet size
ETX Original Implementation

- Each node broadcast probe of fixed size at average period $T$
  
  $$r(t) = \frac{\text{count}(t-w,t)}{w/\tau}$$

  - Typical: size 134 bytes, $w = 10, \tau = 1$

- Each probe contains # probes received from each neighbor

- Use 802.11 broadcast frames, 1 Mbps, 1 mW
  - No ack, no retransmission
ETX in OLSR: olsr.org implementation

- Link quality computed from HELLO packets reception
  - Sent every two seconds

- New LQ Hello messages (broadcast link quality computed for each neighbor)
  - Non RFC 3626 compliant

- Extend TC messages (how good links are)
  - Non RFC 3626 compliant
Packet size typically larger than probe size (193 bytes with overhead)
Accuracy of ETX

- ETX overestimates packet delivery (why? Probably underestimate ACK delivery ratio)
Expected Transmission Time (ETT)

- Start with ETX, multiply with link bandwidth

\[ ETT = \frac{S}{B} \times ETX \]

  - S: packet size, B: link bandwidth (raw data rate)

- How to find B?
  - Fix rate / autorate / ...
  - Use measurement: packet pairs
How to Compute the ETT = Compute B

Packet pairs:
- Send two packets back-to-back: a short (137) followed by a long (1137)
- Measure time difference of reception
- Send back to transmitter
- Min of 10 consecutive samples, divide size of long packet by min value (ignore size of first packet)

Implementation
- 802.11 broadcast frame for loss rate
- Unicast packet for bandwidth (because use of autorate)

Why not RTT: $O(n^2)$
Accuracy of Packet Pairs Measurements

![Graphs showing the accuracy of packet pairs measurements for 802.11a and 802.11g.]
ETT: Assumptions

- Path throughput \( t \) given by:
  \[
  t = \frac{1}{\sum_i \frac{1}{t_i}}
  \]
  - where \( t_i \) = throughput of hop \( i \)

- Underestimates throughput for long paths
  - Distant nodes can send simultaneously

- Overestimates throughput for paths with heavy "self-collisions"
Another ETT Implementation

- Measure delivery probability and compute ETT
  - Delivery prob. measurement
    - For each 802.11b rate: send periodic 1500-byte broadcasts
    - Periodic minimum size 60-byte broadcast at 1 Mbps (Ack)
    - Nodes keep track of received broadcasts and report back to neighbors
    - Delivery prob. = fraction rx @ 1500 * fraction rx @ 60
      - Accounts for ACK, for each rate
  - ETT computation (for best delivery prob.)
    - Delivery prob. * Transmission time of 1500-byte frame
ETT: Expected Transmission Time

- ACKs always sent at 1 Mbps
- Data packets typically 1500 bytes
- Nodes send 1500-byte broadcast probes at every bit rate $b$ (delivery ratio: $d_{f,b}$)
- Nodes send 60-byte (min size) broadcast probes at 1 Mbps (delivery ratio: $d_r$)
- At each bit-rate $b$, $ETX_b = 1 / (d_{f,b} \times d_r)$
- For packet of length $S$, $ETT_b = (S/b) \times ETX_b$
- Link ETT = $\min_b (ETT_b)$
Impact of Interference

- Interference reduces throughput

- Throughput of a path is lower if many links are on the same channel
  - Path metric should be worse for non-diverse paths

- Assumption: All links that are on the same channel interfere with one another
  - Pessimistic for long paths
Evaluations

23 nodes running Windows XP.
Two 802.11a/b/g cards per node: Proxim and NetGear (Autorate)
Diameter: 6-7 hops.
Media Throughput
(Baseline, single radio)
Media Throughput
(Baseline, two radios)
Impact of $\beta$ value

Channel diversity is important; especially for shorter paths
ETX vs Hop Count with Mobility

- Scenario: 45 1-min. TCP transmissions with a mobile sender in a static mesh network

Comparison of Routing Metrics for Static Multi-Hop Wireless Networks, Draves et al., 2004
DSR with ETX

- **DSR Modifications**
  - *Link* cache, run Dijkstra to find best route (at source)
  - ETX measurement using periodic broadcast link probes
  - RREQ forwarding: forwarder address + ETX of incoming link
  - IF RREQ with better accumulated metric for given ID: forward again
  - Metrics included in RREP
  - No overhearing, no salvaging, no reply from cache
DSDV with ETX

- Use short (134 bytes payload) probe every second
- Remember probes over last 10 seconds
Do We Gain Anything?

Setup: 100 random node pairs, throughput CDF

A High-Throughput Path Metric for Multi-Hop Wireless Routing, De Couto et al., 2003
DSDV: Hop Count vs ETX

- Setup: 40 pairs from the 100 random node pairs
- Higher tx power increases connectivity
DSR: Hop Count vs ETX

- ETX helps for initial route choice

Run R1: 1 mW, 134-byte packets

Cumulative fraction of node pairs

Packets per second delivered

- Max 4-hop throughput
- 3-hop
- 2-hop

Best static route
DSR ETX (no feedback)
DSR Hop-count (no feedback)
DSR with feedback: Hop Count vs ETX

- Failure feedback removes low throughput routes
Summary

- Link metrics play a crucial role in the performance of a routing algorithm for wireless networks
- They are still an active area of research
- In practice (today), ETX appears to be the most and widely used metric
- Look at the system overall result
Backup Slides
Capturing Temporal Variability

How to distinguish between links with random errors vs links with bursty errors?

RNP: Required Number of Packets

Consider the following packet traces (\(\bigcirc\) : success, \(\times\) : failure)

10 pkts Tx, 5 pkts Rx; RR = 5/10 = 0.5; 1/RR = 2.0; RNP = 2.5

RNP = \[\frac{\Sigma (2 \ 1 \ 2 \ 1 \ 2 \ 1 \ 2 \ 1 \ 1)}{10} = 1.5\]

10 pkts Tx, 5 pkts Rx; RR = 5/10 = 0.5; 1/RR = 2.0; RNP = 1.5

RNP = \[\frac{\Sigma (1 \ 1 \ 1 \ 6 \ 5 \ 4 \ 3 \ 2 \ 1)}{10} = 2.5\]

RNP takes into account the underlying loss distribution.

RR = reception rate
Link Estimator Study

- Study 7 estimators
  - by tuning to yield the same error bound

- Results
  - WMEWMA(T, $\alpha$) Estimator
    - Stable, simple, constant memory footprint
      - Compute success rate over non-overlapping window (T)
      - Average over an EWMA($\alpha$)
  - Key:
    - 10% $|\text{error}|$ requires at least 100 packets to settle
    - Limits rate of adaptation
Why a new routing metrics?
-ETX (Expected transmission count)

- Shortest path routing not suitable
  - Multi rate issue
- ETX
  - $P = 1 - (1 - p_f) \times (1 - p_r)$
    - $p$: probability that the packet transmission is not successful
  - $ETX = 1 / (1-p)$
    - Good in homogeneous single-radio environments than shortest path routing
- Disadvantages
  - ETX only consider loss rates on the links and not their bandwidths.
  - ETX is designed to give preference to shorter paths over longer paths in case of loss rate is similar.
  - Two bad example scenario: a node with 802.11a and 802.11b NICs, even 2 802.11b NICs
### WCETT (Weighted Cumulative ETT)

1. An estimate of the end-to-end delay:

\[ WCETT = \sum_{i=1}^{n} ETT_i \]

2. Considers the impact of channel diversity:

\[ X_j = \sum_{\text{Hop } i \text{ is on channel } j} ETT_i \]

\[ WCETT = \max_{1 \leq j \leq k} X_j \]

1+2 → \[ WCETT = (1 - \beta) \times \sum_{i=1}^{n} ETT_i + \beta \times \max_{1 \leq j \leq k} X_j \]
ETT: Expected Transmission Time

- ACKs always sent at 1 Mbps
- Data packets typically 1500 bytes
- Nodes send 1500-byte broadcast probes at every bit rate $b$ (delivery ratio: $d_{f,b}$)
- Nodes send 60-byte (min size) broadcast probes at 1 Mbps (delivery ratio: $d_r$)
- At each bit-rate $b$, $ETX_b = 1 / (d_{f,b} \times d_r)$
- For packet of length $S$, $ETT_b = (S/b) \times ETX$
- Link ETT = $\min_b (ETT_b)$
ETT (Expected transmission time)

- ETT: “bandwidth-adjusted ETX”
  - \( ETT = ETX \times \left( \frac{S}{B} \right) \)
    - \( S \): the size of packet
    - \( B \): the bandwidth of the link
- The back-off time is counted in and can not significantly affect the result
- \( p_f, p_r \) can be got by probing \( O(n) \)
- Bandwidth of each link (infrequent unicast probing \( O(n^2) \))
  - One way is to restrict the bandwidth manually
  - Measure it empirically
    - Packet pair
- Why not use RTT for ETT?
  - Self interference (load dependent)
  - Per-neighbor probing is \( O(n^2) \)

Figure 4: Accuracy of packet-pair estimations.
Implementation and such

- Measure loss rate and bandwidth
  - loss rate measured using broadcast probes similar to ETX
    - updated every second
  - bandwidth estimated using periodic packet-pairs
    - updated every 5 minutes

- Implemented in a source-routed, link-state protocol, Multi-Radio Link Quality Source Routing (MR-LQSR)
  - nodes discover links to its neighbors, measure quality of those links
  - link information floods through the network
    - each node has “full knowledge” of the topology
    - sender selects “best path”
    - packets are source routed using this path