Analysis of Link Break Detection using HELLO Messages

Zainab R. Zaidi  Marius Portmann  Wee Lum Tan
National ICT Australia (NICTA)  National ICT Australia (NICTA)  National ICT Australia (NICTA)
zainab.zaidi@nicta.com.au  marius.portmann@nicta.com.au  weelum.tan@nicta.com.au

ABSTRACT
HELLO messages are widely used for neighbor discovery in routing protocols for wireless multihop networks. In this paper, we provide mathematical and experimental proofs that the current strategy of declaring a link down based on observations that a specific number of HELLO messages are lost, is not a correct strategy. In our mathematical analysis, we characterized the error bursts over wireless links by extending the Gilbert bit error model, and validate our model using HELLO loss data from our wireless mesh testbed. Our analysis shows that error burst lengths follow a geometric distribution, i.e. the probability of additional losses in the burst does not depend on the observed losses. We propose an alternative link break detection strategy where a link break is declared when the observed mean error burst length for the link is longer than the mean route recovery time. Our testbed results show that running the AODV routing protocol with our proposed strategy yields better throughput and lower control message overhead, compared to the case where AODV relies on its default settings of declaring link breaks when two consecutive HELLO messages are lost.

Categories and Subject Descriptors
C.2.2 [Network Protocols]: Routing protocols; G.3 [Probability and Statistics]: Distribution functions

General Terms
Experimentation; Performance

1. INTRODUCTION
Although a lot of work has been done to develop routing protocols for wireless multihop networks that could yield reasonable performance over uncertain wireless media, they are generally observed to be under-performing in practical scenarios. This is mostly because various protocol parameters are not tuned to suit the operational settings. A classic example is the HELLO messaging used for neighborhood discovery in various routing protocols, e.g. AODV (Ad hoc On-demand Distance Vector) [13], OLSR (Optimized Link State Routing) [9], etc. Two parameters are associated with the HELLO protocol: $U$: the number of HELLO messages received before a link is declared UP, and $D$: the number of HELLO messages lost before a link is declared DOWN [1].

While the parameter $U$ is usually selected as 1, it is not immediately clear what should be the right value for $D$. In practical multihop networks, wireless link quality may fluctuate randomly (due to interference, fading, etc.). As such, setting $D$ to a low value may lead to poor performance since it will lead to unnecessary declarations of link breaks and subsequent new route discoveries. On the other hand, setting $D$ to a large value is also not a good solution as valuable time would be lost if the link is under a long error burst. These practical issues usually do not arise in simulation experiments where simplistic error models are used for wireless channels.

Not much work has been reported in the literature to optimize the parameters of the HELLO protocol. A default value of $D = 2$ is selected in the original AODV routing protocol based on simulation results [13]. $D$ is termed as the allowed-hello-loss in [13]. In a follow-up work [3], field experiments were carried out with three nodes equipped with 802.11b radios for $D = 2$ and $D = 3$. The experiment results show that $D = 2$ gives better throughput in a mobile setting while $D = 3$ is better in a static setting. A detailed literature survey is presented in Section 5.1.

In this paper, our goal is to come up with a method to select an optimum value of $D$ which would maximize link utilization and throughput. We first develop an analytical model for describing error bursts over a wireless link and validate it using the HELLO loss data from our wireless mesh testbed. The error burst distribution is found by extending the well-known Gilbert bit error model for bursty channels [7]. Our results show that error burst lengths are geometrically distributed. As a consequence, the probability of observing additional losses does not depend on the observed losses. This indicates that the $D$ parameter (or allowed-hello-loss) should not be used as an indicator for broken links. In this paper, we propose an alternative strategy of characterizing each link in terms of its observed mean error burst length, and declaring a link as broken only when it is greater than the average route recovery time of the network. Using this approach, results from AODV routing experiments on our wireless mesh testbed show throughput improvements and lower control message overhead compared to the case with AODV using the default value of $D = 2$. 
The basic Gilbert model is shown in Fig. 1. The system (channel) could be in either of two states: G (Good) or B (Bad). States G and B have different error probabilities, with the error probability in B much larger than that in G. We have simplified the model by assuming that there are no error in G and when in B, there is 100% error. This assumption greatly simplifies our analysis and our validation study shows that the simplified model is sufficient to model packet errors over a bursty channel. Using the conditional probabilities in Eqs. (2) to (5), we can actually build a two-state model similar to the Gilbert model to represent packet errors as shown in Fig. 2. The two states consists of state E, representing the event $E_i$ defined above, and state C, representing a correctly delivered packet defined as the event $C_i$ previously. It is important to note that the transition probabilities for the Gilbert model and the packet error model are not necessarily stationary for practical channels.

### 2.2 Error burst length distribution

Let $BL$ represent the error burst length in terms of number of packets, so we have,

$$P(BL = k) = P(E_i \cdots E_{i+k-1} C_{i+k})$$

$$= P(E_i) P(E_{i+1} | E_i) \cdots P(E_{i+k-1} | E_{i+k-2}) \cdot P(C_{i+k} | E_{i+k-1}),$$

$$= H P_n [1 - H P_n]^k$$

Thus, Eqn. (7) shows that the error burst length follows a geometric distribution. The mean error burst length can then be found analytically as follows:

$$E[BL] = \frac{1 - H P_n}{H P_n}$$

Throughout this paper, we use $E[BL]$ to denote the mean error burst length in terms of consecutive losses of HELLO messages. We interchangeably use the units of $E[BL]$ as
either number of HELLO packets or seconds, as one HELLO message is typically transmitted every second. The mean error burst length is very useful in estimating the geometric distribution parameters for real data, i.e.,

\[ HP_n = \frac{1}{1 + E[BL]} \]  

(9)

Section 4.2 presents model validation results using the HELLO loss data from our wireless testbed links where we have very strong evidence that the above geometric distribution model of packet burst errors is indeed a practical and realistic model. In our experimental study, we have calculated the burst lengths in terms of consecutive lost HELLOs for various wireless links in our testbed in both directions. Using the error burst data, we estimated the mean error burst length and the geometric distribution parameters using (9). We have used statistical tools to match the empirical distribution of the error burst length with a geometric distribution with the same mean.

3. LINK BREAK DETECTION STRATEGY

3.1 Link characterization

Before discussing the implications of our packet error model on the selection of optimal values for the parameter \( D \), an interesting issue is to look at various link quality parameters and their ability to sufficiently characterize the packet loss behavior of a link. The BER (Bit Error Rate) and PER (Packet Error Rate) parameters are widely used to characterize the quality or strength of a wireless link. However, we will show in this section that they are not sufficient measures, especially when used for routing purposes, as there may be very different mean error burst lengths associated with the same BER or PER, depending on the parameters in the error model. Furthermore, wireless links with longer error bursts, on the average, should be treated differently from the ones with shorter error bursts, when selecting a route.

BER is the steady-state probability of being in state B in the Gilbert model, i.e.

\[ BER = \frac{1 - p}{2 - p - q} \]  

(10)

For a specific BER, there could be multiple solutions for the pair of parameters \( p \) and \( q \). Besides conforming to the above equation, \( p \) and \( q \) should also be legitimate probability values, i.e., \( 0 \leq p, q \leq 1 \). A range of feasible solutions is shown by straight lines in Fig. 3 for a specific BER, shown as labels on the lines. For a link with BER = 0.01, \( p \) has a higher value, while it does not matter much what the value of \( q \) is. Similarly, for a very bad link, \( q \) has a higher value irrespective of the value for \( p \).

As shown in Fig. 3, the highest variability in values for \( p \) and \( q \) is observed for middle-range values of BER. This variability has huge effects on the error burst length distribution. Two links with the same BER can have drastically different distributions of error burst length, from close to single packet errors to very long mean error burst length as shown in Fig. 4. Fig. 4 shows the mean error burst lengths for combinations of \( p \) and \( q \) in Fig. 3. Each curve is labeled with the specific BER value. For all values of BER shown in Fig. 4, the mean error burst length seems to be zero or very small for values of \( p \geq 0.97 \).

Re-writing the expected value of error burst length in
terms of BER, i.e. replacing

\[ HP_n = [p^N (1 - q)p^{N-1}] = \left[ \frac{1-p}{1-p^N} \right] \]

in Eqn. (8), we get:

\[ E[BL] = \frac{1}{(1 - BER)p^{N-1}} - 1 \]  

From the expression above, we can see that for the same BER, if \( p \) is larger, the mean error burst length will be smaller as shown in Fig. 4. The mean error burst length increases with BER and also when \( p \) becomes smaller. Thus, we can have links with the same BER but different mean error burst lengths, depending on the probability \( p \) of remaining in the \( G \) (Good) state in Fig. 1.

PER is the steady-state probability of being in state E of the packet error model given in Fig. 2, i.e.

\[ PER = \frac{1 - p^N}{1 + HP_n - p^N} = \frac{(1 - p^N)(1 + E[BL])}{1 + (1 - p^N)(1 + E[BL])}. \]

Similarly as above, we can re-write this expression to show the mean error burst length in terms of the parameters PER and \( p \). As before, we can show that two links with the same PER can have different mean error burst lengths, depending on the parameter \( p \). As such, links specified in terms of BER or PER alone do not give complete information about the packet burst errors.

3.2 Link break detection

Ideally, an optimum value of \( D \) in the HELLO protocol would be one that maximizes the probability of observing longer error bursts given that \( D \) losses have been observed. However, using Eqn. (7), the conditional probability of observing \( k \) additional losses when \( D \) losses are already observed, is found to be independent of \( D \), i.e.

\[ P(\text{BL} \geq D + k | \text{BL} \geq D) = (1 - HP_n)^k = P(\text{BL} \geq k) \]  

This result is a consequence of the memoryless property of a geometric distribution and is in complete contradiction to the way we handle link break declarations in HELLO protocol. Observing some HELLO losses does not change the uncertainty of future losses over a wireless link.

It makes sense to leave a wireless link under fade to look for a better alternative in order to minimize the time of inactivity or the lost time. If the link enters the error state or state E of Fig. 2 and is not declared broken, the average lost time \( E[T_L] \) will be equal to \( E[BL] \), i.e. the mean error burst length. On the other hand, if a link is declared broken as soon as it enters the error state, the average lost time will be \( t_r \), where \( t_r \) is the recovery time in seconds. Recovery time includes the time to notify the source of the traffic, as well as initiating and completing a new route discovery process.

In this paper, we propose to declare a link break if its mean error burst length is longer than the average time required to recover from a link failure, i.e. if \( E[BL] > t_r \), the link is declared down. \( E[BL] \) is considered here in units of seconds. The periodic HELLO messages will be used to estimate \( E[BL] \) within a time window for every link. A very short time window will not allow us to observe all probable error burst lengths, while a very long time window will not allow \( E[BL] \) to adapt quickly to the changes in the wireless medium. We selected the averaging time window to be 100 seconds in our testbed experiments; as the typical mean \( t_r \) is usually under 2 seconds for our network, a window of 100 seconds is more than enough to have error burst lengths greater than \( t_r \) if there is any. Our experiment results show that this strategy works better in terms of yielding higher throughput than the default strategy of using \( D = 2 \) in the AODV routing protocol. Details are given in Section 4.3. We also note that estimating \( E[BL] \) requires an initial setup time in order to compute the first value of \( E[BL] \) over the entire time window. The rest of the values can be updated using a sliding window approach and hence our strategy does not induce any delays in declaring link breaks when compared to the default strategy of using \( D = 2 \).

4. EVALUATION

This section discusses our experimental results in validating the packet loss model, and the performance comparison of the AODV routing protocol in delivering packets with our new strategy of link break detection and with the default scheme of using \( D = 2 \).

4.1 Wireless mesh testbed

We have used NICTA’s wireless mesh testbed for our experiments (http://mytestbed.net/). The NICTA wireless mesh testbed is composed of 40 static nodes installed across three floors at the NICTA office building in Sydney. A layout of a section of the testbed is shown in Fig. 5. In Fig. 5, the yellow (or light grey in greyscale) squares denote the location of the mesh nodes and the numbers show their respective IDs. The black squares and straight lines are irrelevant for our purpose. They show the power outlets in the building with which the mesh boxes are attached to.

Each node consists of a Ubiquity SR2 wireless card supporting the IEEE 802.11a/b/g standard. We have installed
the AODV-UU implementation\(^2\) of the AODV routing protocol on our testbed nodes. In all our experiments, we have used the default configuration of MadWiFi driver for the wireless interfaces and configured the nodes to use 802.11a channel 56.

### 4.2 Model validation

Model validation using real HELLO loss data is the critical component of this study. We have performed experiments over several links in the NICTA wireless mesh testbed spanning multiple days. The links included in our study ranged from very good, where only a small fraction of HELLO messages are lost, to very bad, where only a few HELLO messages are received over a long period of time.

In these experiments, the AODV-UU routing daemons run on various combinations of two or more nodes, and statistics about reception of HELLO messages are collected. An example is shown in Fig. 6 where empirical and analytical distributions are drawn for error bursts over link 16-7, i.e. the link between node 16 and node 7 in Fig. 5. The analytical distribution is given in Eqn. (7) where the parameter \( HP_n \) is estimated from the data, i.e. \( HP_n = (1 + \bar{BL})^{-1} \), where \( \bar{BL} = \sum_i BL_i/N_o \), and \( N_o \) is the number of observed error bursts. The empirical distribution can be found as

\[
P_{emp}(BL = k) = \frac{1}{N_o} \sum_{i=1}^{N_o} 1_{(BL_i = k)}
\]

There are several statistical goodness-of-fit tests defined to estimate the quality of fit of the samples to a particular distribution. We start with Pearson’s \( \chi^2 \) test. This test is very sensitive to the number of observations, and for links with longer error bursts, we need many more observations to have a reasonable test result. Two other goodness-of-fit tests which can work with smaller sample sizes are also used in our experiments. The test statistics are given as follows:

\(^2\)http://aodvuu.sourceforge.net

<table>
<thead>
<tr>
<th>Link</th>
<th>PER (%)</th>
<th>( E[BL] )</th>
<th>Trial length (days)</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>2-3</td>
<td>0.8</td>
<td>0.0081</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>3-2</td>
<td>0.9</td>
<td>0.0091</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>4-2</td>
<td>21.8</td>
<td>0.2791</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>2-4</td>
<td>29.24</td>
<td>0.4137</td>
<td>5</td>
<td>All</td>
</tr>
<tr>
<td>17-9</td>
<td>0.48</td>
<td>0.0048</td>
<td>5</td>
<td>tests</td>
</tr>
<tr>
<td>17-5</td>
<td>65.55</td>
<td>1.9025</td>
<td>1</td>
<td>at</td>
</tr>
<tr>
<td>15-17</td>
<td>0.42</td>
<td>0.0042</td>
<td>1</td>
<td>( \alpha = 0.2 )</td>
</tr>
<tr>
<td>17-15</td>
<td>0.31</td>
<td>0.0031</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>16-17</td>
<td>0.15</td>
<td>0.0015</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>17-16</td>
<td>0.34</td>
<td>0.0034</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>7-4</td>
<td>51.61</td>
<td>1.1429</td>
<td>5</td>
<td></td>
</tr>
<tr>
<td>2-17</td>
<td>91.23</td>
<td>10.4444</td>
<td>1.5</td>
<td></td>
</tr>
<tr>
<td>9-4</td>
<td>53.94</td>
<td>1.7099</td>
<td>5</td>
<td></td>
</tr>
<tr>
<td>4-9</td>
<td>58.18</td>
<td>1.3915</td>
<td>5</td>
<td>CVM</td>
</tr>
<tr>
<td>9-17</td>
<td>41.14</td>
<td>0.699</td>
<td>5</td>
<td>and</td>
</tr>
<tr>
<td>5-17</td>
<td>86.72</td>
<td>6.5295</td>
<td>1</td>
<td>AD</td>
</tr>
<tr>
<td>7-16</td>
<td>68.44</td>
<td>2.1686</td>
<td>1</td>
<td>tests</td>
</tr>
<tr>
<td>16-7</td>
<td>42.71</td>
<td>0.7454</td>
<td>1</td>
<td>at</td>
</tr>
<tr>
<td>7-17</td>
<td>70.86</td>
<td>2.4323</td>
<td>1</td>
<td>( \alpha = 0.2 )</td>
</tr>
<tr>
<td>17-17</td>
<td>57.26</td>
<td>1.3396</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>17-2</td>
<td>99.71</td>
<td>461.667</td>
<td>1.5</td>
<td>AD at ( \alpha = 0.2 )</td>
</tr>
</tbody>
</table>

1. \( \chi^2 \) test:

\[
\chi^2 = \sum_k \frac{(\text{expected}_k - \text{observed}_k)^2}{\text{expected}_k}
\]

where \( \text{expected}_k = N_o \times P(BL = k) \) and \( \text{observed}_k = \sum_{i=1}^{N_o} 1_{(BL_i = k)} \).

2. Cramér-von Mises (CVM) test [2]:

\[
W^2_{N_o} = N_o \sum_k \left( \frac{[F_{emp}(BL = k) - F(BL = k)]^2 \times P(BL = k)}{P(BL = k)} \right)
\]

where \( F(BL = k) = P(BL \leq k) \).

3. Anderson-Darling (AD) test [2]:

\[
W^2_{N_o} = N_o \sum_k \left( \frac{[F_{emp}(BL = k) - F(BL = k)]^2 \times P(BL = k)}{F(BL = k)(1 - F(BL = k))} \right)
\]

We also use truncation with the CVM and AD tests where the statistics is summed for the values of \( k \) in the interval \([1, \min(k \geq \text{max observed } BL; (1 - F(BL = k))^3 \leq 10^{-4}/N_o)]\). The bootstrap method is used to calculate the
critical values for the statistics. The bootstrap method uses 1000 trials and generates \( N_o \) samples in each trial \([2]\). The hypothesis test is defined as
\[
H_0 : P_{emp}(BL = k) = P(BL = k) \\
H_1 : P_{emp}(BL = k) \neq P(BL = k)
\] (13)

The summary of the validation results is shown in Table 1. For 25 links, out of a total of 27 links in our validation study, \( H_0 \) was accepted at \( \alpha = 0.2 \) by the CVM and AD tests with proper truncation to avoid tail behavior. For links from node 4 to node 7, and from node 5 to node 4, \( H_0 \) was accepted by the CVM test at \( \alpha = 0.05 \) and the AD test at \( \alpha = 0.005 \).

For 10 out of the 27 links, we are able to accept \( H_0 \) at \( \alpha = 0.2 \) by the \( \chi^2 \) test as well. In general, these links have smaller mean error burst length and our experiment duration was long enough to get sufficient samples suitable for a \( \chi^2 \) test. Moreover for 23 links, we are also able to accept \( H_0 \) by the CVM and AD tests without the use of truncation defined above. The links in our experiment varied in quality. The minimum average error burst length observed was 0.001, in terms of the number of consecutive HELLO losses, with PER = 0.2% for the link from node 16 to node 17. The maximum average error burst length observed was 461.67 with PER of 99.7% for the link from node 17 to node 2. The HELLO loss data over these 27 links were collected for different durations ranging from 1 to 5 days, as shown by the \( \text{Trial length} \) column in Table 1.

4.3 Performance evaluation

To compare the default setting of \( D = 2 \) in the AODV routing protocol with our new strategy of declaring a link break when \( E[BL] > t_r \), where \( t_r \) is the mean route recovery time, we selected a few groups of nodes in our testbed which have at least two paths between the source node and the destination node. For example, Fig. 7 shows the available paths between the source node 1 and the destination node 16, via nodes 2, 5, 7, and 14. In our experiments, these paths consist of a different number of hops, ranging from a single hop to five hops across the different groups of nodes. A 54 Mbit/s UDP flow is transmitted between the source node and the destination node within each group of nodes for 10 minutes using the Iperf tool. We selected UDP traffic to avoid the artifacts of the congestion control mechanism of TCP flows.

![Figure 7: Available paths between node 1 and node 16 in one of our experiments.](image)

We ran AODV-UU with its default parameters on all the nodes for our baseline experiment. For the alternate strategy of link break detection, we modified the AODV-UU code to save the history of HELLO messages reception within the specified averaging window, which is set to 100 seconds in these experiments. Whenever a new HELLO message is received from a neighbor node, the receiver recalculates \( E[BL] \) using the received HELLO messages history. If \( E[BL] > t_r \), the value of \( D \) is set to 1 so that the next time the link is in error or enters the E state of Fig. 2, the link will be declared down. On the other hand, if \( E[BL] \leq t_r \), we keep \( D \) at its initial value, which is 100 in our experiments. This implementation maintains the ‘link break’ mechanism of AODV-UU which is triggered by the parameter \( D \), although it is not a very accurate implementation of our strategy. This is because by updating \( E[BL] \) only when a HELLO message is received, we may miss an error burst with length less than the initial value of \( D \) until it finishes. Furthermore, the link is assumed to be active throughout such a burst. Nevertheless, we remark that this inaccuracy degrades the performance of our strategy and an accurate implementation can only enhance the throughput performance reported in our results. For all of our experiments, we calculated \( t_r \) from the log files of our baseline experiments. The alternate strategy of link break detection requires a startup time to estimate \( E[BL] \), so we started UDP flows in all the experiments after 2 minutes of starting the AODV daemon on the nodes.

![Figure 8: Throughput comparison using the default setting of \( D = 2 \) and the new strategy of declaring a link broken when \( E[BL] > t_r \).](image)
### Table 2: Throughput comparison.

<table>
<thead>
<tr>
<th>Source-Destination</th>
<th>Nodes in the experiment</th>
<th>Hops</th>
<th>Average throughput (Mbits/s)</th>
<th>Average RREQ/RREP overhead (No. of packets)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>2-4</td>
<td>(D = 2 ) (E[BL] &gt; t_r)</td>
<td>(D = 2 ) (E[BL] &gt; t_r)</td>
</tr>
<tr>
<td>1-16</td>
<td>1, 2, 5, 7, 14, 16</td>
<td>1-3</td>
<td>3.4</td>
<td>4.68</td>
</tr>
<tr>
<td>16-5</td>
<td>5, 7, 12, 16, 17</td>
<td>2-3</td>
<td>2.92</td>
<td>3.04</td>
</tr>
<tr>
<td>28-32</td>
<td>28, \cdots, 32</td>
<td>1-3</td>
<td>11.02</td>
<td>12.9</td>
</tr>
<tr>
<td>2-4</td>
<td>2, 4, 5</td>
<td>1</td>
<td>26.32</td>
<td>28.48</td>
</tr>
</tbody>
</table>

always consist of one hop as shown in the bottom graph, although an alternate path of two hops via node 5 could also be selected. The middle graph shows the average packet error rate (PER) calculated from the loss of HELLO messages for the active path. PER is averaged over the interval when a path is active. Interestingly, the more stable plot of PER for \(E[BL] > t_r\) shows paths remain active for a longer duration with our strategy of link break detection compared to that of \(D = 2\). The top graph shows the throughput performance. Averaged over 10 minutes, the mean throughput using our strategy is 29.96 Mbits/s, while with the default setting of \(D = 2\), the mean throughput is 27.65 Mbits/s.

Table 2 shows the average throughput for all of our experiments, with the averaging performed over the length of the experiment (which is set at 10 minutes) and also over the number of experiments. We conducted three trials for each experiment. Table 2 also shows the average control message overhead of Route-Request (RREQ) and Route-Reply (RREP) packets transmitted in the experiments, with the averaging performed over the number of trials. We have counted all RREQ and RREP packets transmitted from all nodes, from the log files. Since our strategy, in general, yields more stable paths, the control message overhead of route discovery is also considerably low. The more stable paths, resulting from the use of our strategy, are also responsible for the more stable PER plot in the middle graph of Fig. 8. Interestingly, for the last experiment listed in Table 2 where a one hop path was always selected, \(D = 2\) was responsible for reporting a link break so many times that a huge number of RREQ broadcasts were needed to find the destination which is only a hop away.

### 5. RELATED WORK

#### 5.1 Selection of D or allowed-HELLO-loss

Not much work has been reported in the literature to optimize the parameters of the HELLO protocol. The original AODV paper [13] presented a simple simulation based results to show that \(D = 2\) is the best choice. The paper presented route acquisition latency and goodput results for \(D = \{1, \cdots, 4\}\) for networks with 50 and 100 nodes. \(D = 2\) is selected as the best choice despite it having the worst latency for the 50-nodes network and not achieving the best goodput for both 50-nodes and 100-nodes networks. In a follow-up work [3], field experiments were carried out with three 802.11b radios for \(D = 2\) and \(D = 3\). The experiments were performed in two modes: static (all nodes are stationary) and mobile (one node moves further away). The results show that \(D = 2\) gives better throughput in a mobile setting while \(D = 3\) is better in a static setting.

Some other recent works include the performance analysis of neighborhood discovery via HELLO messaging [1, 12]. Their approach utilizes Markov chain modeling to derive the steady-state probabilities of link detection. However, the studies do not consider the drawbacks of using a larger \(D\), i.e., lower throughput when the link remains inactive for long period.

#### 5.2 Packet and bit error models

While a lot of work has been done in characterizing errors in bursty channels in order to evaluate the performance of error correction and detection coding schemes, not much has been done to validate these models for wireless multihop networks using 802.11 radios. In the context of wireless mesh networks, a lot of research is focused on modeling the effects of hidden terminals on packet delivery while ignoring all other causes of errors [4].

The classic approach of modeling bit errors using a 2-state Markov chain is given by Gilbert [7], where the Good state has zero error rate and the Bad state has non-zero error probability. Gilbert’s model is generalized by Elliott to introduce a smaller error rate in the Good state compared to that of the Bad state [5]. This model is further generalized to include \(N\) states partitioned into \(k\) Good states and \(N - k\) Bad states in [6]. In [6], expressions are also given for error free run lengths and error burst lengths as products of state transition probabilities in the Gilbert-Elliott model.

Several enhancements have also been suggested for the basic Gilbert model which are more realistic for specific scenarios, e.g. in [11], the Gilbert-Elliott model is enhanced and compared for 802.11 wireless links. The enhancement involves making the transition probability between Good and Bad states to be dependent on the current bit [11]. An alternate model is presented in [14] where independent errors are considered separately from correlated errors for general packet transmission systems.

While the enhanced versions of the Gilbert model promise more accuracy, they are mathematically more complicated and it is difficult to derive closed-form and simple expressions for required statistics such as probability of error burst length, etc. We started with the simplified form of the Gilbert model in this paper and our strategy was to use more complicated versions of the model only when validation with real data fails. The simplified Gilbert model, used in our paper, has also been successfully used to describe both the packet loss characteristics in Internet video transmission [8], and the cell discard model for MPEG video transmission in ATM networks [15]. Other efforts include the characterization of packet or block errors by extending the Gilbert model [10, 16] in a similar fashion as in [6]. In [16], mutual information arguments are used to show that a first-order time-homogeneous Markov chain is sufficient to rep-
resent packet errors in a bursty channel. However, in [10],
it is shown that this can be achieved only by using non-
stationary, time-inhomogeneous Markov chain. Simulations
based on the Gilbert model are used to verify the accuracy
of packet characterization in [16] and [10].

The packet loss model developed in this paper is in ac-
cordance with the previous results [6, 10, 16]. The use
of the simplified Gilbert model helped us to come up with
the closed-form expressions for the distribution of error burst
length and the transition probabilities for the packet level
model, as functions of the parameters in the Gilbert model.
We also stepped forward and validated the model using real
HELLO loss data from our wireless mesh testbed, thus mak-
ing a genuine case for its application in practical scenarios.

6. CONCLUSION

In this paper, we have shown that declaring a link break
when a specific number of HELLO message losses are ob-
served is not a correct strategy. Our mathematical anal-
ysis yields a geometric distribution for error burst length for
HELLO message losses. This result is also validated using
real HELLO loss data from our wireless mesh testbed. For
a geometric distribution, the probability of observing addi-
tional losses does not depend on the observed losses, and
hence D or allowed-hello-loss is an irrelevant parameter for
the HELLO protocol. Instead, we propose to characterize a
link based on its observed mean error burst length and when
it is found to be more than the mean route recovery time
for the network, it is better to declare a link break in order
to improve link utilization and minimize loss of time. Our
testbed experiments show that improved throughput and
lower control message overhead performance are achievable
using our new strategy, compared to the default setting of
HELLO messaging in the AODV routing protocol.

As a future extension to the current study, we intend to
look into the discrepancy in HELLO message reception rate
and data delivery rate over a wireless link and how it affects
the performance of our new strategy for link break detection.

Acknowledgement

The authors like to acknowledge the assistance of Ms. Shachi
Shastry, Dr. Thierry Rakotoarivelo, and Mr. Christoph
Dwertmann in setting up the experiments over the NICTA
wireless mesh testbed.

7. REFERENCES

[1] BARAS, J. S., TABATABAEE, V., PURKAYASTHA, P.,
AND SOMASUNDARAM, K. Component based
performance modeling of the wireless routing
protocols. In Proceedings IEEE ICC Ad Hoc and
Sensor Networking Symposium 2009 (June 2009),
p. 1–6.

comparative study of goodness-of-fit tests for the
geometric distribution and application to discrete time
reliability.
[Online; accessed 11-May-2011].

of hello messages for determining link connectivity. In

The 5th International Symposium on (2002), vol. 2,
p. 504 – 508 vol.2.

loss of hello-messages in a wireless mesh network. In
Communications (ICC), 2010 IEEE International
Conference on (May 2010), pp. 1 –6.

[5] ELLIOTT, E. Estimates of error rates for codes on
burst-noise channels. Bell Systems Technical Journal

[6] FRITCHMAN, B. A binary channel characterization
using partitioned markov chains. Information Theory,

Systems Technical Journal 39 (September 1960),
1253–1265.

[8] GIBOD, B., STUHLMÜLLER, K., LINK, M., AND HORN,
U. Packet loss resilient internet video streaming. In
Proceedings of SPIE Visual Communications and

[9] JACQUET, P., MUHLETHALER, P., CLAUSEN, T.,
LAOUTI, A., QAYYUM, A., AND VIENNOT, L.
Optimized link state routing protocol for ad hoc
(August 2001), pp. 62–68.

the packet error statistics in bursty channels. In Local
27th Annual IEEE Conference on (November 2002),
p. 534 – 541.

as parsimonious bit error models of wireless channels.
In INFOCOM 2003, Twenty-Second Annual Joint
Conference of the IEEE Computer and
Communications, IEEE Societies (March-3 April

of neighbor discovery in proactive routing protocols.
In Proceedings of the 7th ACM workshop on
Performance evaluation of wireless ad hoc, sensor,
and ubiquitous networks (New York, NY, USA,
October 2010), PE-WASUN ’10, ACM, pp. 66–70.

on-demand distance vector routing. In Proceedings
of the 2nd IEEE Workshop on Mobile Computing
Systems and Applications (February 1999),
p. 90–100.

[14] SARKER, J., HASSAN, M., AND HOSSAIN, A.
Statistical modelling of correlated and independent
errors in packet data transmission systems. In
Personal Wireless Communications, 1997 IEEE
International Conference on (December 1997), pp. 499
–504.

on qos for mpeg video transmission in atm networks.
International Conference on (June 1999), vol. 1,
p. 147 –151 vol.1.

errors in bursty channels. Communications, IEEE
Transactions on 45, 6 (June 1997), 660 –667.