PERFORMANCE EVALUATION OF MULTI-HOP AD-HOC WLANS
Farshad Eshghi, Ahmed K. Elhakeem, and Yousef R. Shayan
Dept. of Electrical and Computer Engineering
Concordia University
Montreal, QC, Canada

ABSTRACT
Ongoing technological advances in portable devices, coupled with the need for continuous connectivity while mobile, have made ad-hoc networks a compelling research and development topic. Performance evaluation of the IEEE 802.11’s ad-hoc mode of operation, as a dominating WLAN protocol, requires a thorough investigation, particularly in a multi-hop scenario. Throughout this work, a modular simulation environment comprising traffic generator, mobility, wireless channel, and IEEE 802.11 protocol modules was developed to evaluate system performance measures, such as delay and packet failure rate. Simulation results show that the optimal selection of the system parameters can lead to considerable improvement in various performance measures which are of special interest to time-sensitive applications.

KEY WORDS
Ad-hoc networks, IEEE 802.11, performance evaluation, wireless local area network (WLAN), multi-hop.

1 Introduction
Mobile computing continues to enjoy rapid growth thanks to ongoing technological advances in portable devices which led to lowering the power consumption, and lengthening the battery life-span. Simultaneously, the need for continuous connectivity while roaming, has put the ad-hoc networks at the center of attention of researchers and the industry over the past few years. Portability and mobility are in great consistence with the infrastructure-less nature of ad-hoc networks. This has made mobile ad-hoc networks (MANET) ([1],[2]) a compelling research topic. On the other hand, viewed either as a stand-alone network or as a last-mile wireless connection to the internet, ad-hoc networks have the capability of expansion through multi-hopping. Multi-hop MANETs, in situations where they are supposed to support multimedia, present a challenge in many aspects. This is attributed to the fact that applications with different natures demand different treatments. Multi-hop MANETs [3] have features that distinguish them from wired and infrastructured wireless networks. These features can be enumerated as: unpredictable link properties, node mobility, limited battery life, route maintenance, security [4], hidden terminal [5], exposed terminal, and capture effect [6].

In multihop ad-hoc networks due to the numerous variables involved, system multi-dimensionality grows significantly thus making analytical modeling a harder task. Turning to simulation enables us to investigate a greater number of phenomena and possibly more involved models regarding traffic, channel, etc. In such a complex system, optimal selection of the system parameters can lead to considerable improvement in performance especially for time-sensitive applications.

The motivation for our work herein is to compensate for the fact that all the past simulation efforts have adopted too many restrictive assumptions, and to find out if the 802.11’s ad-hoc mode can serve as the core building block of a multi-hop, mobile, ad-hoc network. We have tried to develop a thorough simulation environment spanning both MAC and PHY layers which considers mobility and varying wireless channel as well as FEC using turbo coding. Aiming at time-sensitive applications, our goal is to evaluate the performance measures of the system such as end-to-end delay and packet failure rate in different situations (e.g. traffic, mobility, etc) and against different choices of system parameters (e.g. fragmentation factor, buffer length, retransmission limit, etc). The paper is organized as follows. In Section 2, a topological description of the system is presented. Traffic generation, mobility, channel, and 802.11 protocol modules are introduced in Section 3. Key assumptions made throughout the simulation are intermitently mentioned when discussing the modules. Computer simulation results are presented afterwards in section 4. Concluding remarks appear in section 5.

2 System Description
A number of \( n_{tot} \) nodes are uniform-randomly located inside a rectangular area. As the first basic assumption, all nodes are aware of others’ geographical locations. Our main emphasis in this paper is the interaction of physical and MAC layers, so we simplify the simulation of routing. We assume the information exchange among all nodes to be GPS-based and certainly to result in some bandwidth cost. Each node with a packet ready to be sent picks another node as its next (intermediate) destination provided that: it is inside the source node’s coverage area, and has the least Euclidean distance to the packet’s final destination. While the latter may not be the best routing strategy, in most cases it leads to the least number of hops traversed.
and is of course easy to implement. However, any other routing algorithm [7] can easily be incorporated in our simulation environment. In the initialization phase, some nodes are randomly selected as mobile and the rest are left as fixed nodes. The mobility status will prevail for the whole simulation time. The parameter $P_{\text{mobile}} = \frac{v_{\text{mobile}}}{v_{\text{next}}}$ along with $v_{\text{mobile}}$: mobile node’s speed, represent the mobility volume of the system. In a similar way, randomly selected nodes are responsible for generating traffic while all the nodes might serve as forwarding and final destination nodes at any time during the simulation. A parameter representing the population of traffic generating (active) nodes is defined as $P_{\text{active}} = \frac{v_{\text{active}}}{v_{\text{next}}}$. Each active node is pre-assigned a node as its final destination which remains unchanged throughout the course of simulation.

3 Simulation Modules

3.1 Traffic Generation

Considering voice connection as a representative of real-time applications, the packet generation model employed is an on-off keying model. Figure 1-a illustrates the traffic generation model. While in the active state (ACT in the figure), the model acts as a constant bit rate (CBR) source generating r bit/s and while in idle state (IDL in the figure), no packet is generated. Parameters $\alpha$ and $\beta$ are self-loop transition probabilities in IDL and ACT states respectively.

The slot time of the model ($t_{\text{slot}}$) is considered to be long enough to accommodate the longest packet size generated above the user’s link layer. The average idle and active states dwelling times are then:

$$T_{\text{IDL}} = t_{\text{slot}}(1 - \alpha)^{-1} \quad T_{\text{ACT}} = t_{\text{slot}}(1 - \beta)^{-1}$$  

(1)

Throughout the simulation, the two scalar functions source(IDL,NO) and status(IDL,NO) determine whether the node #IDL,NO is a traffic generator and presently active respectively. Upon packet generation at source node, vector functions pkt(SEQ,NO,CP,VER,HOP,VER) and ver(SEQ,NO) are responsible for carrying over all the information pertinent to the generated packet. Each packet is uniquely identified by its 3-tuple identifier (SEQ,NO,CP,VER,HOP,VER) as will be explained shortly in Section 3.4.1.

3.2 Mobility

All nodes designated as mobile, periodically (each $\Delta/v$ second where the input parameters $\Delta$ and $v$ denote displacement step length in meters and mobile’s speed in m/s respectively) update their location. This moves the mobile node as much as $\Delta$ along either x or y axes. The two functions that deal with mobility are mobile(IDL,NO) and location(IDL,NO) which respectively specify the initial mobility status and updated location of the nodes throughout the simulation. Large scale fading and shadowing phenomena are directly related to the macro-movements of the nodes, and so, they are handled in the mobility module. Using two-slope curve model [8] and log-normal distribution for the path loss and shadowing respectively, we will have:

$$P_L(d) = P_L(d_0) + 10n \log_{10}\left(\frac{d}{d_0}\right) + X(\nu, \sigma); \quad d \geq d_0$$

$$P_L(d) = 20\log_{10}\left(\frac{\Delta}{\lambda}\right); \quad d < d_0$$

(2)

In (2), $n_{PL}$ and $\lambda$ represent terrain-dependent path loss exponent and wavelength respectively, and $X$ is a normal random variable in dB. Parameter $d_0$ denotes the break-point distance below which the environment has free space characteristics.

3.3 Channel Model

In this module, the small scale behavior of the channel, in particular the multi-path fading as its major contributor is simulated. The two important parameters that quantify the channel behavior with regard to fading classes are coherence bandwidth ($B_c$) and coherence time ($T_c$). Less conservative expressions for the coherence bandwidth and coherence time are [9]:

$$B_c \approx \frac{1}{5\sigma_f} \quad T_c \approx 0.423C \frac{v_m f_c}{\nu_m f_c}$$

(3)

In (3) $\sigma_f$ denotes the delay spread of the wireless channel and $f_m$, $v_m$, $f_c$, and $C$ denote maximum Doppler frequency, maximum mobile’s speed, carrier frequency, and light speed respectively which are related through $f_m = \frac{v_m f_c}{c}$. The power spectral density of the DSSS signal in 802.11 is limited to around 20-22 MHz using a transmit spectrum mask [10]. This bandwidth translates to a 9-10 nsec delay spread through applying (3). While this value of delay spread may fall into the rural area range, suburban and urban areas show higher delay spreads ([11],[12]) and subsequently lower coherence bandwidths. To avoid frequency band variations of the signal strength, at this point we adopt the flat fading channel assumption which may not be valid for all terrain situations. Regarding the coherence bandwidth and for the system under study, values of $C = 10^6$ m/s, $f_c = 2.4$ GHz, and $v = 10$ m/s give a coherence time of $T_c = 5.29$ msec. Compared to the
symbol period of 1μsec, the wireless channel can conventionally be considered as slow fading. However, since we are concerned about packet rather than symbol (bit herein), depending on the packet length, a packet may experience several channel conditions. So at any time we consider the channel as a time-invariant stationary process with different statistics during different coherence-time units. In order to model the time variations of the channel, a two-state Markov model (Gilbert model) is adopted. We assume that a mobile node along its movement alternatively experiences either a line of sight (LOS) or a Rayleigh fading channel along its target node. Figure 1-b shows the embedded channel model in our simulation with α′ and β′ denoting the self-loop probabilities of bad (B) and good(G) states. By equating $T_C$ in (3) to the model’s slot time ($t'_\text{slot}$), and the average dwelling time of the bad and good states with $T_B$ and $T_G$ respectively, we obtain:

$$T_B = t'_\text{slot}(1 - \alpha')^{-1}, \quad T_G = t'_\text{slot}(1 - \beta')^{-1} \quad (4)$$

In the good state, the received power ($P_{RG}$) is calculated from the transmitted power ($P_T$) as $P_{RG} = P_T - P_L(d)$ where $P_L(d)$ is defined in (2) with all powers in dB. In the bad state, the above obtained $P_{RG}$ is used to find the Rayleigh distributed envelope of the received signal ($r$) through:

$$f_R(r) = \frac{2r}{P_{RG}}e^{-r^2/P_{RG}} \quad (5)$$

and in turn the bad state received power as $P_{R_B} = r^2$.

3.4 802.11 Protocol Implementation

IEEE 802.11 protocol implementation in the ad-hoc mode is discussed here. More emphasis is put on the MAC layer while the PHY layer implementation is realized through adopting parameters related to the specific choice of modulation, spreading, and FEC coding.

3.4.1 MAC Layer

The random back-off algorithm is the contention resolution strategy in the IEEE 802.11 WLAN protocol operating in ad-hoc (DCF) mode. For a more detailed discussion of the protocol, the reader is referred to [13]. Real-time traffic, depending on the application, requires a compromise between delay and loss. Late fragments are not worth more than lost ones. To take care of this, and as an amendment to the protocol toward real-time adaptability, unacknowledged fragments are retransmitted for a limited number of times ($R_{\text{max}}$) before releasing the channel. Moreover, $R_{\text{max}}$ unsuccessful retransmissions are followed by a fragment drop to avoid too much latency of the late fragments. Moreover, we assume that the fragmentation is done at layers higher than MAC so we only need to deal with different MPDU-length packets. In terms of performance evaluation, the latter does not seem to produce considerable discrepancy with respect to the MAC fragmentation scheme. Due to the ad-hoc nature of the network, the MAC protocol is implemented in a distributed manner (concurrently in all nodes) and thus there are some peculiarities which will be highlighted herein. Throughout the initialization phase, three functions are created that reflect the complete status of the node (other than those provided by source(), status(), and mobile()) during the simulation process. Vector function node(ID_NO) gives information such as the node’s location, destination (in case it is generating any traffic), number of packets in the queue, back-off counter’s content, contention window size, mode of activity, etc. Vector function Q(ID_NO,CELL_NO)) on the other hand, provides the specifics about the packets currently in the queue of node #ID_NO. The sequence numbers of all the packets that already have visited a particular node are stored in archive(ID_NO). At each slot, shown for instance in Fig. 2, each node checks whether its channel is idle. If so, and if NAV=0, proper actions are taken according to its most recently completed activity (e.g. transmission, reception, listening, or being idle). These actions include updating node(), pkt(), Q(), and archive() and/or initiating RTS, CTS, DATA, or ACK transmissions. Mechanisms that result in packet elimination are: blocking due to the target node’s full buffer (or at the generation time), dropping due to reaching some constraints such as time-to-live, maximum hopping, retransmission limit, duplication (explained below), route unavailability, and successful reception. The only verification mechanism regarding a healthy DATA transmission is through ACK exchange (Fig. 2), which itself may or may not fail. Thus, situations may arise in

![Figure 2: Random access and data transmission in IEEE 802.11 ad-hoc mode of operation using channel reservation scheme RTS/CTS: Successful transmission of MPDU.](image-url)
which several copies of the same packet are hopping to the destination. By considering the 3-tuple packet identifier, (SEQ_NO,CP_VER,HOP_VER), the multi-version packet phenomenon is well simulated. SEQ_NO is set by the originating node. CP_VER is manipulated by the transmitting node at the time of retransmission, and HOP_VER is updated by the receiving node. Since all versions of a packet have the same unique sequence number assigned at the generation time, flooding can be partially prevented. As introduced earlier, the function archive(ID_NO) encompasses the sequence number of all the packets which have already visited node #ID_NO. As such, a correctly received packet’s sequence number is compared against the list in archive(). The new version of the packet is either discarded if a match is found, or forwarded if not. Upon a healthy reception, a copy of the received packet is generated with an incremented hop version and an acknowledgment is sent back to the transmitter. From this point on, this newly generated packet (new version) is processed by the receiving node. First, whether the packet is a duplicate is checked, and then it is checked for time-to-live violation where in either case it is discarded. In the case of survival, a successful packet delivery is recorded if the packet is at its final destination; otherwise it is considered for further forwarding. In this stage, it may again get discarded due to a full buffer or to a maximum hop limit situation; otherwise it will be queued. Figure 3 illustrates the whole simulation process and how different earlier mentioned modules are embedded.

### 3.4.2 PHY Layer

Spreading, modulation, and coding schemes are considered here. The DSSS system uses baseband modulations of BPSK (DBPSK) to provide 1 Mbit/s data rate. Corresponding physical layer parameters used are $L_{RTS}/L_{CTS}/L_{ACK} = 20/14/14$ octets, $OH_{MAC}/OH_{PHY} = 34/24$ octets, and $T_{DIFS}/T_{SIFS}/Slot time = 50/10/20$ μsec, where $L_{RTS}/L_{CTS}/L_{ACK}$ and $OH_{MAC}/OH_{PHY}$ denote RTS/CTS/ACK bit lengths, and MACPHY layer overhead bits respectively. As an illustrative example of turbo code application to WLANs, our turbo coding scheme features two identical recursive systematic convolutional (RSC) encoders with parameters $n = 2$, $k = 1$, $K = 3$, $G_0 = 7$, and $G_1 = 5$, with half of the parity bits punctured. The decoder is Log-MAP with 8 iterations. However, no effort has been made to seek the optimum scheme. Regarding the interleaver, we consider two types: one, which shuffles fixed short size control packets (RTS, CTS, and ACK), is a block interleaver; and the other one, which deals with data packets, is a random separated interleaver. For the short packets (control packets), depending on the received SNR, corresponding BER is extracted from BER vs. SNR curve in AWGN/BPSK.
case with the above employed coding scheme [14, Fig. 5.26]. Since short packets experience almost the same channel quality, except at boundary situations, packet error rate (PER) is then:

\[ \text{PER}_{\text{CTRL}} = 1 - (1 - \text{BER})^{L_i} \]  

where \( i = \text{RTS}, \text{CTS}, \text{ACK}. \) Data packets, depending on their length and time coherence of the channel (\( L_c \) in (3)), experience several channel qualities represented by \( \text{BER}_1 \), ..., \( \text{BER}_x \) in \( \text{pk}() \) function (Fig. 3). The PER is accordingly calculated as:

\[ \text{PER}_{\text{DATA}} = 1 - (1 - \text{BER}_z)^{\text{mod}(L_{\text{DATA}}, L_c)} \times \prod_{m=1}^{z-1} (1 - \text{BER}_m)^{L_{Te}} \]  

where \( L_{Te} \) denotes the equivalent bit length of \( T_c \), \( \text{mod}(L_{\text{DATA}}, L_c) \) accounts for the remainder of dividing \( L_{\text{DATA}} \) (number of bits in DATA) by \( L_{Te} \), and \( z \) is the number of coherence times within one packet. There is an implicit approximation in (7), however, which stems from the fact that in turbo coding the equivalent bit error rate corresponds to a frame length as big as the interleaver size rather than \( L_{Te} \). Finally, in the simulation program, the above calculated PER is compared against a uniform-randomly selected number between zero and one to result in a failed or successful transmission.

4 Simulation Results

In this section, the effects of different choices of MAC layer parameters as well as other external network parameters (e.g. mobility, traffic load, etc.) on the system performance are simulated. In particular, MAC layer packet length, maximum life time of a packet (time-to-live, TTL), maximum number of times a packet is retransmitted (\( RcTX_{\text{max}} \)), and a node’s maximum buffer size (\( Q_{\text{max}} \)) are examined. This is done in different network situations by varying the percentage of active nodes (\( P_{\text{active}} \)), the percentage of mobile nodes (\( P_{\text{mobile}} \)), and mobile node speed (\( v \)). However, due to space limitations, mobility/traffic-related results are not presented here. All the fixed parameters of the course of simulation are reported in Table 1. Two major measures considered herein as performance indicatives are end-to-end (e.t.e) delay and packet failure rate (PFR), wherein such a delay has been presented as normalized with respect to subject DATA packet time length. The delay includes the time duration from the moment of packet generation until its healthy reception. Not included are packetization delay, interleaving delay in turbo encoding, propagation delay, delays related to turbo decoder iterations, and other packet processing times at intermediate and destination nodes. However, these delays are both fixed and negligible compared to random transmission and queuing delays. In each simulation run, e.t.e delay is calculated by averaging over all healthy received packets. Packet failure rate is the ratio of the number of packets dropped to the total number of generated packets throughout the duration of simulation. Figures 4-6 illus-

<table>
<thead>
<tr>
<th>Table 1: Parameters used in the simulation.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Network parameters</td>
</tr>
<tr>
<td>( \eta_{\text{tot}} )</td>
</tr>
<tr>
<td>Node’s coverage radius</td>
</tr>
<tr>
<td>Network area</td>
</tr>
<tr>
<td>Displacement step (( \Delta ))</td>
</tr>
<tr>
<td>Simulation time</td>
</tr>
<tr>
<td>Source model parameters</td>
</tr>
<tr>
<td>Node’s traffic rate</td>
</tr>
<tr>
<td>Avg. talk spurt period</td>
</tr>
<tr>
<td>Avg. silence period</td>
</tr>
<tr>
<td>Max. generated packet length</td>
</tr>
<tr>
<td>delivered to the link layer</td>
</tr>
<tr>
<td>Channel parameters</td>
</tr>
<tr>
<td>Path loss exponent (( \eta_{\text{PL}} ))</td>
</tr>
<tr>
<td>Break point distance (( d_0 ))</td>
</tr>
<tr>
<td>Standard deviation of shadowing (( \sigma ))</td>
</tr>
<tr>
<td>( \beta )</td>
</tr>
<tr>
<td>( \alpha )</td>
</tr>
<tr>
<td>Miscellaneous parameters</td>
</tr>
<tr>
<td>Transmitter power (( P_T ))</td>
</tr>
<tr>
<td>Noise floor (( NF ))</td>
</tr>
<tr>
<td>Simulation slot time</td>
</tr>
<tr>
<td>Maximum hopping</td>
</tr>
</tbody>
</table>
Figure 4: Performance measures of the system versus DATA packet length with time-to-live (TTL) as a parameter: a) Normalized delay b) Packet failure rate

Figure 5: Performance measures of the system versus DATA packet length with maximum number of retransmissions (ReTX\text{max}) as a parameter: a) Normalized delay b) Packet failure rate

descent by a climbing packet length. The descents become less significant and particularly indistinguishable beginning at the 4000-bit packet length. On the other hand, in the PFR plot and for the same range of \(L_{DATA}\) (4000-bit onward), \(ReTX_{max} = 2\) undoubtedly is the best choice. Figure 6 shows the effect of varying queue size and DATA packet length on the system performance measures. The left plot, delay, gives almost the same trend and conclusion as the corresponding plot in Fig. 5. Lengthening the queue has conflicting secondary drop effects such as less packet drop due to queue overflow and higher packet drop due to time-to-live violation. Depending on which one has the dominating effect, a better or worse PFR is observed. The PFR plot suggests that (\(Q_{\text{max}} = 2, L_{DATA} = 6000\) bits), (\(Q_{\text{max}} = 5, L_{DATA} = 4000\) bits), and (\(Q_{\text{max}} = 5, L_{DATA} = 8000\) bits) are the preferable choices. In interpreting the quantities of the performance measures, it should be noted that the results pertain to pure 802.11 protocol and no extra efforts such as revising the protocol, clustering, etc have been made.

5 Conclusion

In order to investigate the performance of the DCF in the multi-hop scenario, a simulation environment was developed. The simulation comprising traffic generator, mobility, wireless channel, and IEEE 802.11 protocol modules, was intended to evaluate system performance measures such as delay and packet failure rate. Here we are not considering throughput, since the maximum bandwidth available can not be clearly defined due to the multi-channel nature of the multi-hop. Most of the performance measures above do not demonstrate a monotonic trend against the parameters considered, and this shows the importance and necessity of the optimum values sought.

An important observation is that apart from the max-
im minimum delay, which was constrained to the TTL (chosen suchwise to fulfill many of the requirements of the time-sensitive applications), the other measures results do not seem promising in view of real-time applications. However, one should consider that the access method under study is the pure 802.11 DCF mode of operation in the multi-hop situation, without any possible priority, coordination, route reservation, and clustering considerations. Hence, it can be regarded as almost the worst case performance. In view of this, IEEE 802.11’s DCF mode of operation can be eligible to serve as the core medium access technique of a multihop mobile ad-hoc WLAN, certainly with the proper aforementioned boosts.

References


Figure 6: Performance measures of the system versus DATA packet length with maximum queue length ($Q_{\text{max}}$) as a parameter: a) Normalized delay b) Packet failure rate

![Figure 6: Performance measures of the system versus DATA packet length with maximum queue length ($Q_{\text{max}}$) as a parameter: a) Normalized delay b) Packet failure rate](image-url)