Implementing Frequency Selective Fading for OPNET's Wireless LAN Modules

Oscar Puñal, James Gross
Mobile Network Performance Group, UMIC Research Centre
RWTH Aachen University, Germany

Marc Emmelmann
Telecommunications Network Group
TU Berlin, Germany

E-mail: punal@umic.rwth-aachen.de

Abstract

In OFDM-based IEEE 802.11 Wireless LANs, adapting the modulation type per sub-carrier to the frequency selectivity of the wireless channel leads to significant performance improvements. To integrate adaptive modulation into IEEE 802.11a/g networks, some minor modifications to the standard are needed in order to acquire the subcarrier gains and signalize the modulation assignments (on a per-packet basis), amongst others. However, to implement adaptive modulation in OPNET some changes have to be performed at the physical layer processing to include a frequency selective fading profile and to accurately compute packet error events based on the frequency-selective channel profile. We take an analytical approach for this and demonstrate a much more precise packet error behavior than with standard OPNET modules.

Introduction

A lot of research in the wireless networking community targets at the further improvement of wireless local area networking as described by the IEEE 802.11 standard. In this context, evaluating new approaches by means of simulation is always subject to criticism as the accuracy of simulation models in the wireless domain is much more an issue than regarding wired networking systems. A particular problem arises from the channel behavior in wireless local area networks compliant to IEEE 802.11 a,g or n. Due to the transmission scheme, namely orthogonal frequency division multiplexing, frequency-selectivity of the channel (caused by multi-path propagation environments) leads to a varying channel gain per sub-carrier. The transmission over a single sub-carrier can safely be assumed to be subject to flat-fading. However, the transmission over the entire WLAN bandwidth (of about 16 MHz) cannot be assumed to be subject to frequency-flat fading (even in indoor scenarios). This frequency-selectivity is usually not taken into account, however, it has been shown recently to have a very strong impact on the performance of WLANs [1].

In this paper we discuss an implementation for taking these issues into consideration based on the network simulation tool OPNET version 14.0. Our main motivation for implementing such an extension was to study the integration of adaptive modulation into WLAN. It is not the intention to discuss this issue in this paper, the interested reader can refer to [2] for this subject. Instead, we present here the implementation approach for OPNET and highlight the performance differences seen if considering our approach instead of the one provided by OPNETs standard models. The remaining paper is structured as follows: First, we discuss the packet error rate model of OPNET for WLANs. Afterwards, we present our approach and we finally discuss some performance issues.

Overview of OPNETs WLAN Packet Error Model

The purpose of this section is to summarize briefly how packet error events are generated with OPNET. OPNET's PHY layer is modeled through a set of pipelined stages associated partially with the transmitter while the most of them are associated with the receiver of the ongoing communication. The error model is implemented in the thirteenth stage („error“ stage) of the transceiver pipeline and gets as input previous calculations that take place at the „received power“ stage, the „signal-to-noise ratio“ stage and the „bit-error rate“ stage, amongst others. The error stage calculates, based on the inputs, the number of bit errors present in the packet or packet segment. This value, which is read by the last stage of the pipeline („error correction“ stage), decides ultimately about the correct packet reception together with a pre-selected error correction threshold. In detail, the error event generation operates as follows:

1. Initially, the current SNR at the receiver (for the corresponding packet transmission) is generated. The SNR is determined by considering a certain path loss model, the transmit power, the noise level as well as antenna effects. However, there is no default support for frequency-selective fading. This is a problem especially in OFDM systems as shown for example in [1]. Large deviations in the simulated performance have been observed which can lead to very different performance results for example when considering several interfering WLAN cells. The major difference between considering frequency-selective fading and not doing so is a significant change of the packet error behavior.

2. Once the received power and signal-to-noise ratio (SNR) are obtained, the bit-error-rate (BER) is calculated. For that purpose, predefined SNR-to-BER relationships are available in form of tables per modulation type. The input SNR for that curves should be the bit-SNR or Eb/No. However, instead of applying Eb/No, OPNETs standard model does not consider the effects of the different modulation types and uses the symbol-SNR or Es/No.

3. Next, the instantaneous BER per packet is generated, based on the previously determined average BER per SNR. The probability that a certain number of errors occur in a packet with a certain length can be computed by the binomial probability function. A loop within this stages’ computer code increases the number of wrong bits per packet until the summed probability is larger than a previously generated random number. This generates the total amount of wrong bits per packet. This value is then passed to the next pipeline stage which determines if the packet can be corrected or not.

4. The last stage of the transceiver pipeline („error correction“ stage) is responsible for the decision of accepting or rejecting a received packet based on the number of bit errors, the length of the packet and an error correction threshold selected at the receiver. This threshold is defined as the highest proportion of bit errors allowed in a packet to still be considered as a correctly decoded one. The error correction threshold can be set differently for different error correction types. While the corresponding setting is important for the observed system performance, it is neither documented nor somehow standardized and therefore this is an unclear aspect of OPNET's error model.

New Implementation of WLAN Packet Error Model

Our main motivation for improving the packet error event generation originate from the question how to model (and implement) frequency-selective fading in an appropriate way. The goal of our investigations was to study bit loading and adaptive modulation in WLANs, taking the full protocol modifications described in [2] into account. The shortcomings of the OPNET model in this context are first of all that no frequency-selective channel model is provided and second, that for the packet error model based on the error correction thresholds no reasonable values are known for setting these thresholds (especially in the context of frequency-selective fading as well as if adaptive modulation per sub-carrier is considered). Hence, we are interested in a new channel model and also in an error model which can handle either the same modulation type over all sub-carrier or adaptive modulation per sub-carrier while taking a frequency-selective channel profile into account. A further constraint was to make the error model applicable to different coding schemes.

The general, new architecture for accomplishing these goals is as follows: Instead of considering a single SNR for the link of the current packet transmission, this base SNR is split into 48 SNR values and a fading component is added to each sub-carrier SNR. The fading can either be read from a file (if for example a channel matrix file is considered as in the case for 802.11n, where the channel matrix (potentially for MIMO transmission) is obtained from Laurent Schumachers MATLAB tool [3]. Alternatively, exponentially distributed fading taps can be generated if the correlation of the fading in frequency is not taken into consideration. Next, for each SNR per sub-carrier the BER is determined, depending on the modulation type applied per sub-carrier. Given these values, an overall BER is determined by averaging the individual sub-carrier BERs (also considering in this averaging step the chosen constellation size: the sub-carriers with a higher modulation type have a larger impact on the average BER than sub-carrier with low modulation types). This yields the overall instantaneous BER which now has to be considered in combination with the respective error correction scheme. We refer to this BER in the following as β.

The chosen approach for the packet error rate generation is based on an analytical model. It generates a tight upper bound for the packet error probability, which takes the average instantaneous BER β as input. In [4,5] an upper bound of the bit error probability is derived for binary convolutional coded transmission with hard-decision Viterbi decoding and independent bit errors. This assumption can be done, since the interleaving block of the OFDM transceiver chain reduces the error's correlation. The resulting bit error probability is given by:

$$P_b \leq \frac{1}{k} \sum_{d=d_{free}}^{\infty} c_d \cdot P_b \qquad (1)$$

In this equation, k is the number of input bits to the register of the convolutional encoder, d_{free} is the free distance of the convolutional code, P_d is the probability that an incorrect path of distance d is chosen and c_d is the number of bits in error in that case. The values for c_d can be obtained by derivations; we have used the values from [6] for the rate 1/2 coder with generator (133,171). For the punctured rates with 3/4 and 2/3 we have used the corresponding values given in [7]. P_d can be upper bounded as

$$P_d \leq 2 \cdot \beta \cdot (1-\beta) \qquad (2)$$

where β is the instantaneous BER as described above.

Given the bound on the resulting bit error probability Pb, we can obtain the packet error probability for a packet of size ς bits (including MAC header's length but excluding redundancy bits introduced by the convolutional coder) by:

$$P_{packet} \leq 1-(1-P_b)^{\varsigma} \qquad (3)$$

This upper bound is accurate and considerably tight for low input BER, however it loses precision under higher uncoded bit error probabilities (about 10^{-3} and larger), which is illustrated in Figure 1 for a rate 2/3 encoder. In that case, the bound of Equation 1 overestimates the resulting coded bit error probability and hence a too high packet error probability is obtained. We correct this by introducing a scaling factor to the resulting coded bit error rate of Equation 1. Figure 1 illustrates this manipulation. The figure shows the resulting coded bit error rate vs. SNR (i.e. versus input BER β). The curve on top (in green) is the direct result from Equation 1. In comparison, we show results from simulating the transmission chain and the real encoder/decoder with standard MATLAB tools (dashed, blue curve below). As stated, there is a gap for lower SNR while the bound is tight for high SNR. In order to overcome this problem, we calculate from the two curves correction factors by applying Lagrange interpolation (applied according to the outcome of the bound). The resulting curve that we obtain from this operation is shown also (solid, red line). As we see, the manipulation generates directly the curve of the MATLAB simulations. Finally, we obtain a precise packet error probability model which allows evaluating different packet sizes, different coding schemes and different physical layer approaches. In Figure 2 we show the resulting (direct) relationship between input (uncoded) BER – i.e. β – and the corresponding output (coded) BER for convolutional coding with rate 3/4. Note that the OPNET packet error rate model assumes a certain error correction code to always decode up to a certain ratio of bits in error per packet (the precise number depending on the respective ECC threshold). The presented curve shows that this is not valid: while the slope of the curve stays constant for an area between 5 * 10^{-3} and 4 * 10^{-2}, the relationship changes constantly below and above these input (uncoded) BER values. Hence, assuming a constant relationship does lead to wrong performance results.

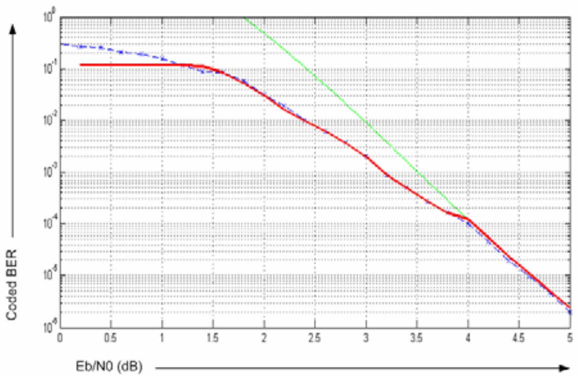

Figure 1: Coded bit-error rate vs. channel Eb/No (i.e. versus uncoded bit error rate): Upper bound (green), simulated curve (blue) and corrected curve (red) considering a (133,171) convolutional code with rate 2/3.

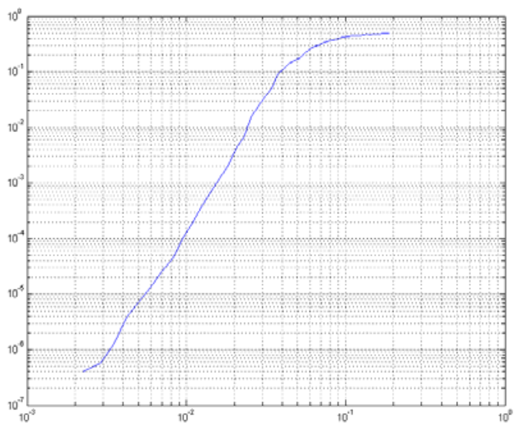

Figure 2: Uncoded (input) BER on the x-axis vs. coded (output) BER on the y-axis for a rate ¾ convolutional coding scheme with hard-decision Viterbi decoding.

System Performance Consequences

In this section, we first compare OPNET's default model with the extended one under a flat-fading AWGN channel. Afterwards (and only for the new model) frequency selective fading is switched on to demonstrate the effects of a more realistic channel model. The packet error rate and the goodput are the metrics studied. The system considered corresponds to OFDM-based IEEE 802.11a, working at the 5 GHz band. We focus on the PHY modes 5, 6 and 7 to illustrate the performance impact for all three convolutional coding schemes applied in 802.11a (mode 5: 16-QAM with convolutional coding rate 1/2; mode 6: 16- QAM with rate 3/4; mode 7: 64-QAM with rate 2/3). The simulation scenario consists of an access point that communicates with a single station in saturation mode, meaning the AP has always packets in its buffer ready to be transmitted. A packet length of 228 byte is considered while we deactivate RTS/CTS handshake. We use for all results the OPNET built-in tables for determining BER from SNR.

Let us first focus on the direct difference between the packet error rate model from OPNET and the one proposed in this paper. In order to come up with a comparison, the obvious question is how to set the error correction threshold for OPNET. For the setting of this ECC threshold, we first obtain for each considered PHY mode and a certain reference SNR the PER generated by our proposed model. Next, we determine the corresponding ECC threshold that yields the same PER for the corresponding SNR and the same PHY mode. The reference points are the following: mode 5 at 16 dB generates a PER of 0.0028, mode 6 at 16 dB generates a PER of 0.0835 and mode 7 at 22 dB generates a PER of 0.0253. As result, we come up with an ECC threshold of 1.3 % for mode 5, 1.02% for mode 6 and 0.79% for mode 7. However, we notice an extreme sensibility of the packet error rate depending on the chosen ECC threshold. Table 1 illustrates this by varying the ECC threshold and showing the corresponding packet error rates. While an ECC threshold of 0.013 yields a PER of 0.0028, reducing the ECC threshold to 0.01 already leads to an increase of the PER to about 0.1. Hence, if OPNETs ECC threshold model is used, much care should be spent on how to choose the ECC threshold.

ECC (in percent)	PER (in percent)
0.5	93
1	9.45
1.15	2.52
1.2	1.57
1.3	0.28
1.4	0.06
2	0

Table 1: Packet error rate dependency on the ECC threshold setting

In Figure 3, we show the resulting packet error rates (vs. SNR) derived from OPNETs model and our proposed one for frequency flat fading. Notice that the above determined ECC thresholds are kept constant for the full SNR range considered. The main difference between both models relates to the slope of changing from a high packet error rate to a low one. Notice that in general for frequency-flat fading channels this transition is rather sharp. However, we can observe from the figure that the OPNET models feature an extremely sharp transition. Essentially, from a certain SNR point on transmission is possible by the respective PHY mode without even any performance reduction by packet errors. In contrast, our model (which is not considering frequency-selective fading in this case) is able to provide a smoother transition from high to low packet error rates as it can be expected from real transmission. Notice that our model does not depend on a certain parameterization which has to be determined for the respective case first. Instead, once the relationship between the upper bound and the real coded bit error rate is determined, this model can be applied flexibly (and with a large credibility) to other situations like considering frequency-selective fading or considering adaptive modulation per sub-carrier.

In the following we focus on the impact if frequency-selective fading is considered. In Figure 4 we show the resulting goodput of a single station in a WLAN cell that is constantly served by the AP. In this case, we assume a frequency-flat channel. Notice the relatively step rate increase (which is even steeper if considering the model of OPNET).

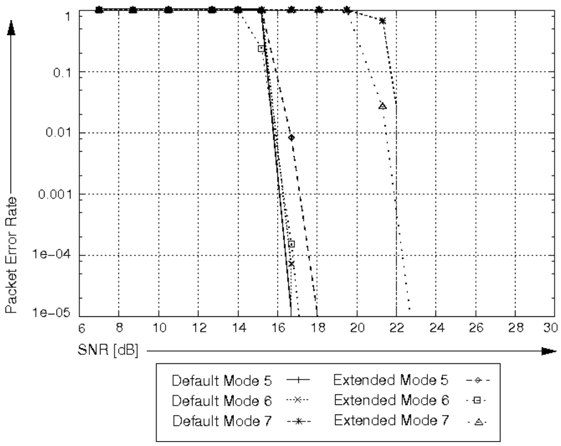

Figure 3: Average packet error rate comparison of OPNETs WLAN packet error rate model and our proposed model. We show the PER curves for the three different PHY modes 5,6 and 7 .

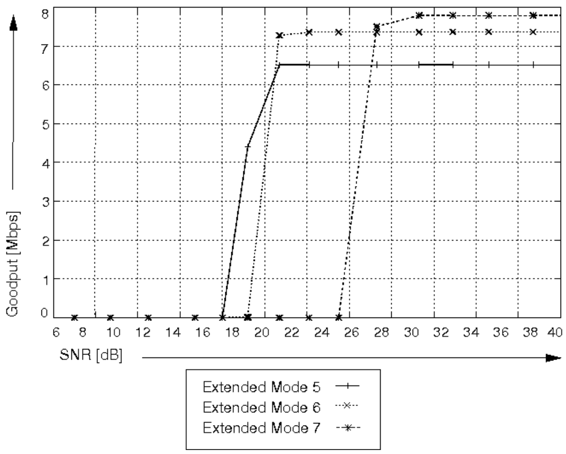

Figure 4: Goodput curves for the new error model under frequency-flat fading for modes 5, 6 and 7.

Next, we show in Figure 5 the corresponding values if frequency-flat fading is considered (using an RMS delay spread of 100 ns). Notice the longer transition from the areas of almost zero goodput up to the areas with a high goodput. The most important point to notice here is that this can be considered without a dedicated parameterization of the model. The same error model used for frequency-flat fading can be applied to frequency-flat fading. Even using this model in more complex situation (bit loading the sub-carriers), the proposed model can be employed, producing highly credible results.

Conclusions

In this paper we propose a new model for OPNETs packet error generation for WLANs. The model is based on an analytical approach and provides a more flexible, credible way to generate packet error events when simulating WLAN performance. We demonstrate its flexibility by applying the model to frequency-selective fading and show how the model is implemented in OPNETs transmission chain.

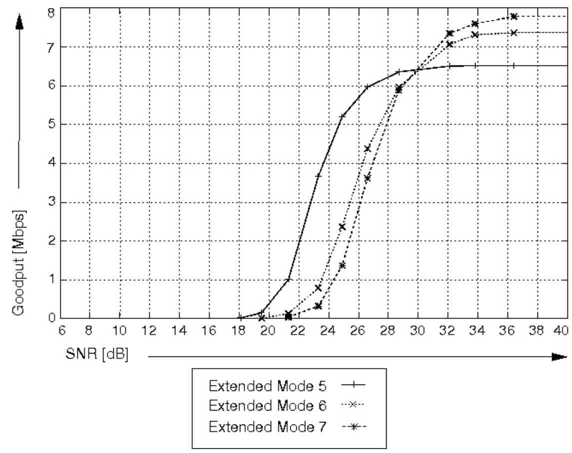

Figure 5: Goodput curves for the new error model and for modes 5, 6 and 7 under the presence of Rayleigh frequency selective fading

Acknowledgements

This work has been done while all authors were with the Telecommunication Networks Group, TU Berlin.

References

[1] O. Awoniyi and F.A. Tobagi, “Packet Error Rate in OFDM-based Wireless LANs Operating in Frequency Selective Channels”, INFOCOM, 2006.

[2] J. Gross, M. Emmelmann, O. Puñal, A. Wolisz, Dynamic Single-User OFDM Adaptation for IEEE 802.11 Systems, in: Proc. ACM/IEEE International Symposium on Modeling, Analysis and Simulation of Wireless and Mobile Systems (MSWIM 2007), 2007.

[3] L. Schumacher, “Wlan MIMO Channel Matlab Program Tech. Rep. URL:“http://www.info.fundp.ac.be/^{~}lsc/Research/IEEE_80211_HT%SG_CMSC/distribution_terms.html”

[4] M. Bossert, “Channel Coding for Telecommunications”, John Wiley & Sons, Inc., 2000.

[5] J. Proakis, “Digital Communications”, McGraw-Hill, 1995.

[6] P. Frenger, P. Orten, T. Ottoson, „Convolutionl Codes with Optimum Distance Spectrum“, IEEE Trans. Commun. 3 (11) (1999) 317-319

[7] D. Haccoun, G. Begin, “High-Rate Punctured Convolutional Codes for Viterbi and Sequential Decoding”, IEEE Trans. Commun. 37 (11) (1989) 1113-1125