A new ns-3 WLAN error rate model – Definition, validation of the ns-3 implementation and comparison to physical layer measurements with AWGN channel

[Extended Abstract – Poster Presentation]

M.Eng. Christopher Hepner
Institute of Communication Technology
University of Applied Sciences Ulm
Ulm, Germany
hepner@hs-ulm.de

B.Eng. Arthur Witt
Institute of Communication Technology
University of Applied Sciences Ulm
Ulm, Germany
awitt@hs-ulm.de

Prof. Dr. Roland Muenzner
Institute of Communication Technology
University of Applied Sciences Ulm
Ulm, Germany
muenzner@hs-ulm.de

ABSTRACT

In network simulations a correct representation of the physical layer is essential in order to achieve reliable results which are comparable to real hardware performance. The network simulator ns-3 specifies two error rate models for the calculation of the bit error rates and corresponding packet error rates for orthogonal frequency division multiplexing, the YANS [3] and NIST [4] error rate models. In [4] both models are validated and the NIST model is recommended for the calculations because of the overly optimistic results in the YANS model. Still some inaccuracies are present in both of the models which are discussed in this work. A new model is presented in this work and changes to both models present are proposed, which lead to results comparable to real hardware performance. The upper bounds calculated in the models are compared with measurements over an AWGN channel with a typical IEEE 802.11 a/b/g/n wireless LAN module.

1. NEW NS-3 ERROR RATE MODEL

The ns-3 physical layer model is based on the calculation of bit error rates (BER) taking into account the forward error correction present in IEEE 802.11a/g/n. The model calculates the received signal–to–noise ratio (SNR) based on parameters used in the simulation model and calculates a packet error rate (PER) based on the mode of operation (e.g. modulation, coding rate) to determine the probability of successfully receiving a frame (packet success rate - PSR). Assuming an Additive White Gaussian Noise channel (AWGN), binary convolutional coding and hard decision viterbi decoding, the PER can be upper bounded by using the equations in [3], [4] and the following section.

1.1 New model for abstraction of the physical layer

The relation between the available energy per bit E_b and transmitted energy per OFDM symbol E_{OFDM} is described in [1]. This is the energy distributed along the whole symbol, including signaling overheads (see equation (1)).

$$E_{\mathrm{b}} = E_{\mathrm{OFDM}} \left(\frac{N_{\mathrm{FFT}}}{N_{\mathrm{FFT}}+N_{\mathrm{CP}}}\right) \left(\frac{N_{\mathrm{data}}}{N_{\mathrm{data}}+N_{\mathrm{pilot}}}\right) \cdot \left(\frac{1}{N_{\mathrm{data}} N_{\mathrm{BPSCS}} R}\right) \tag{1}$$

N_{FFT} is the FFT length. N_{CP} is the length of the cycling prefix (CP). N_{data} is the number of data subcarriers and N_{pilot} the number of pilot carriers respectively. N_{BPSCS} is the number of coded bits per symbol in each OFDM subcarrier and R the code rate of the Forward Error Correction (FEC). Based on the relation in equation (1) the SNR (Signal-to-Noise Ratio) and $\frac{E_{\mathrm{b}}}{N_0}$ (SNR per bit or the ratio of energy per bit to the one-side noise spectral density N_0) can be determined. The SNR is defined as the ratio of signal power to noise power. The signal power is the energy (variance) per time sample. [1]

$$P_{\mathrm{signal}} = \frac{E_{\mathrm{OFDM}}}{N_{\mathrm{FFT}}+N_{\mathrm{CP}}} \tag{2}$$

N_0 is the noise power. Therefore using (1) and (2) the SNR can be calculated as:

$$SNR = \frac{E_{\mathrm{b}}}{N_0} \left(\frac{N_{\mathrm{data}}+N_{\mathrm{pilot}}}{N_{\mathrm{FFT}}}\right) (N_{\mathrm{BPSCS}} R) \tag{3}$$

The representation in YANS [3], equation (4) doesn't take into account the energy per bit to noise power spectral density ratio $\frac{E_b}{N_0}$ correctly which is used for the BER calculations. $\frac{E_b}{N_0}$ in YANS [3] is calculated by the fraction of the noise bandwidth B_t over the raw bitrate R_b (calculated by the number of bits per OFDM symbol over the symbol interval time) times the SNR. Interference is assumed to be zero.

$$\frac{E_b}{N_0} = \mathrm{SNR}\frac{\mathrm{B}_t}{\mathrm{R}_b} \tag{4}$$

Workshop on ns-3 (WNS3) '2015 Barcelona, Spain

This formulation doesn't take into account the reduction of energy due to the cycling prefix (1st term in eq. (1)) and the reduction of the net energy due to the pilot carriers which do not transport information (2nd term in eq. (1)) [1]. On the other hand equation (5) of the NIST model [4] doesn't account for the ratio of used subcarriers in the OFDM system and CP at all.

$$\frac{E_b}{N_0} = \frac{\mathrm{SNR}}{\log_2(\mathrm{M})} \tag{5}$$

The bit error probability p_{BPSK} for BPSK (Binary Phase-Shift Keying) is calculated as shown in [3]. However in order to calculate the raw bit error probability, the raw $\frac{E_b}{N_0}$ (equation (6)) has to be used.

$$\left(\frac{E_{\mathrm{b}}}{N_0}\right)_{\mathrm{raw}} = SNR\left(\frac{N_{\mathrm{FFT}}}{N_{\mathrm{data}}+N_{\mathrm{pilot}}}\right)\left(\frac{1}{N_{\mathrm{BPSCS}}}\right) \tag{6}$$

The performance achieved by the Viterbi decoding algorithm in the receiver then can be estimated by calculating the bit error probability P_b^u. The probability of incorrectly selecting a path when the Hamming distance d is even or odd $P_2(d)$ is shown in YANS [3]. The error probability can be overbounded by the sum of the pairwise error probabilities $P_2(d)$ over all possible paths that merge with the all-zero path at the given node [5] while NIST [4] is using an approximation also called Chernoff Bound which yields to a loser upper bound on the probability of a bit error. The multiplication factors α_d used for the calculation of the union bound in YANS correspond to the number of pahts of the set of distances d. Instead the multiplication factors β_d which corresponds to the number of nonzero information bits that are in error when an incorrect path is selected for the specified hamming distance d have to be used to obtain the upper bound on the probability of a bit error. The coefficients and d_{free} for punctured Codes can be taken from [2]. We thus obtain the upper bound for the error probability according to equation (7).[1] [6]

$$P_b^u < \frac{1}{b}\sum_{d=d_{free}}^{\infty} \beta_d P_2(d) \tag{7}$$

The packet success rate (PSR) is calculated as shown in the YANS and NIST model by using equation (8) with the number of data bits L instead of raw bits[2] L.

$$PSR \leq (1 - P_b^u)^L \tag{8}$$

1.2 Validation of the error rate models

The upper bounds are validated using a Matlab implementation of the equations shown in [3], [4] and section 1.1 for BPSK (R=1/2). Figure 1 shows the results of the ns-3 implementations for the YANS and NIST model and the calculation of the union bound, upper bound and Chernoff bound for [3]K=1 and K=10 in Matlab compared to the new model. The SNR in the Matlab implementation is calculated by using equation (6). The union bound is calculated as shown in the YANS model [3], the upper bound is calculated by equation (7) and $P_2(d)$ as shown in [3]. The Chernoff bound is calculated by equation (7) and $P_2(d)$ as shown in [4] respectively. The packet success rate (PSR) for the union, upper and Chernoff bound is then calculated by using equation (8) with the number of data bits $L = 8000$ bits while the ns-3 YANS and NIST simulation are using the number of raw bits.

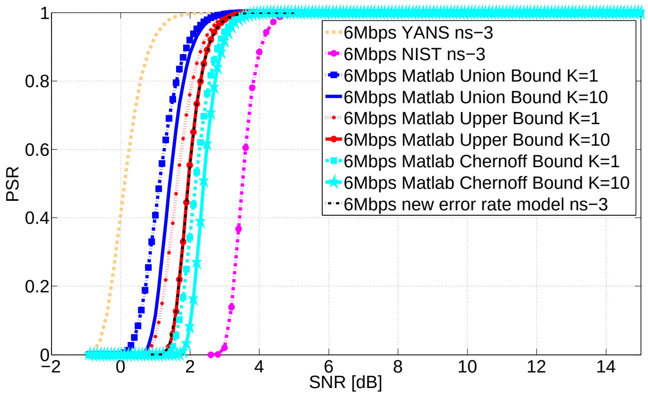

Figure 1: Comparison of ns-3 and Matlab implementation.

The Matlab implementation of the upper bound with K=10 (red solid line) corresponds to the ns-3 implementation of the new error rate model (black dash-dot line). The upper and Chernoff bound vary less than 1dB. The ns-3 YANS model is using the union bound with K=1 and the NIST model is using the Chernoff bound with K=10. Because of the inaccurate SNR calculation and the wrong number of bits for the calculation of the PSR (raw bits instead of data bits which are double the size for R=1/2) the YANS model is overly optimistic (more than 1dB) whereas the NIST model is pessimistic (more than 1dB) when compared to the Matlab implementations using the respective bounds used in the YANS and NIST implementations. The results outlined above indicate the following changes to the ns-3 implementation of the YANS and the NIST model: First the number of raw bits used in equation (8) must be multiplied by the code rate R. Second the calculation of the SNR must be changed to the representation in equation (6). Third it is recommended to use the upper bound for the calculation instead of the Chernoff or the union bound.

2. WIRELESS TESTBED MEASUREMENTS WITH AWGN CHANNEL

Figure 2 shows the result of the measurement over an AWGN channel compared to the upper bound and Chernoff bound for K=1 and K=10 as well as the ns-3 implementation of the model proposed in this paper. The results show a good correspondence of the measurement results with the upper and Chernoff bound for K=10 as well as with the ns-3 implementation of the new model. With the changes described in the previous sections and including the noise figure of the WLAN Hardware module in the network simulation, the YANS and NIST model then using a correct SNR calculation and correct number of bits in ns-3 could

---

[1] $b = (1, 2, 3)$ for code rates (1/2, 2/3, 3/4). The additional factor of 1/2 which is presented in [4] but not in [6] is not taken into account here.

[2] raw bits = data bits / code rate.

[3] K is the number of coefficients α_d or β_d used for the calculation of the error bound.

be used as well to estimate the performance of the physical layer of a typical WLAN module. In this case YANS would correspond to the union bound with K=1 and NIST would correspond to the Chernoff bound with K=10. With the hint that as seen in figure 1 the YANS model changed would still show too optimistic results of about 1dB and the NIST model would show pessimistic results by less than 1dB compared to the new model.

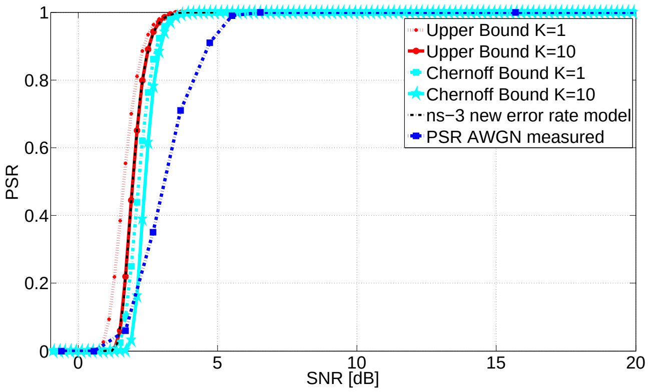

Figure 2: Result of the ns-3 implementation of the new model when compared to the Upper Bounds and PSR of BPSK for R=1/2 over an AWGN channel.

The flattening of the measured AWGN graph might be due to inaccuracies in the measurement setup. Only the WLAN receiver module is placed in a shielded box. Part of the signal from the transmitting module might be propagating over the air and feeding back on the screen of the cable, finally entering at low amplitude the screened box where the receiver is placed. This results in a slightly frequency selective fading characteristic. To prevent this the transmitting module needs to be placed in a shielded box as well and cable lengths should be as small as possible. Therefore essential for the validation is the region of low packet success rates.

3. CONCLUSION

The simulation results of the new ns-3 error rate model show a good correspondence with the measurement results of a typical WLAN module. Key elements of the new model are the usage of the upper bound (equation (7)) with K>=10, correctly taking into account the SNR at the receiver input (equation (6)) and correctly taking into account the number of data bits used for the calculation of the PSR in equation (8). Because of the inaccurate SNR and PSR calculations in YANS and NIST the YANS model is overly optimistic (about 2dB) whereas the NIST model is too pessimistic (about 2dB) when compared to the new error rate model. The calculation of the SNR should be changed in both, the ns-3 NIST and YANS error rate models in order to correctly take into account the error rate in the OFDM system. The number of data bits used in the calculation of the packet success rate should be changed as well in order to take the convolutional coding into account correctly. In order to get reliable results in the ns-3 simulations, it is crucial to take into account the noise figure of the RF-Front-End.

4. REFERENCES

[1] A. M. Camara and R. P. F. Hoefel. On the Performance of IEEE 802.11n: Analytical and Simulation Results. *XXIX Simposio Brasileiro de Telecomunicacoes - SBrt'11*, 2011.

[2] D. HACCOUN and G. BEGIN. High-Rate Punctured Convolutional Codes for Viterbi and Sequential Decoding. *IEEE TRANSACTIONS ON COMMUNICATIONS, VOL. 37, NO. 11*, 1989.

[3] M. Lacage and T. R. Henderson. Yet Another Network Simulator. *Proceeding from the 2006 workshop on ns-2: the IP network Simulator*, 2006.

[4] G. Pei and T. R. Henderson. Validation of OFDM error rate model in ns-3. *Boeing Research & Technology, https://www.nsnam.org/ pei/80211ofdm.pdf*, 2010.

[5] J. G. Proakis. *Digital Communications*. McGraw-Hill, New York, 2001.

[6] A. J. Viterbi. Convolutional Codes and Their Performance in Communication Systems. *IEEE Transactions on Communications Technology, Vol. Com-19, No.5*, 1971.