

Link-to-System Mapping for ns-3 Wi-Fi OFDM Error Models

Rohan Patidar
University of Washington
Seattle, USA
rpatidar@uw.edu

Sumit Roy
University of Washington
Seattle, USA
sroy@uw.edu

Thomas R. Henderson
University of Washington
Seattle, USA
tomhend@uw.edu

Amrutha Chandramohan
University of Washington
Seattle, USA
amruthac@uw.edu

ABSTRACT

The ns-3 simulator contains detailed models of the Wi-Fi MAC layer, including beaconing, rate control, collision avoidance, block acknowledgments, and many other features. However, it relies on abstraction at the physical layer to scale well; Wi-Fi frames are evaluated by specialized interference trackers and analytical error models to arrive at frame reception decisions on a frame-by-frame basis, rather than symbol-by-symbol. Analytical models can provide fairly tight bounds for simple scenarios (additive white Gaussian noise (AWGN) channels with single antennas and limited interference), but the industry relies on detailed link-level simulations to understand more complicated scenarios. This paper reports on an extensive campaign to conduct link simulations of Wi-Fi OFDM performance over AWGN and fading channels, using a commercial link simulator with Wi-Fi support, with results validated against published references. Next, we describe a specific implementation of a technique generally known as link-to-system-mapping, to allow a vector of per-subcarrier signal-to-noise ratios to be distilled into a single “effective SNR” value that can be used to determine performance using link simulation results of the AWGN channel. Finally, we report on the support of our link simulation results in a new ns-3 ErrorRateModel based on tables compiled from link simulation results. Our broader contributions are the link simulation programs themselves which allow others to reproduce and extend the basic tables that we provide, and flexibility in the ns-3 implementation to allow additional tables to be added over time.

CCS CONCEPTS

• **Networks** → Network simulations; • **Computing methodologies** → Discrete-event simulation; • **Computer systems organization** → Robotics;

KEYWORDS

Wi-Fi, Network Simulator 3 (ns-3)

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

WNS3 2017, June 2017, Porto, Portugal

© 2017 ACM. 978-1-4503-5219-2/17/06...\$15.00

DOI: <http://dx.doi.org/10.1145/3067665.3067671>

ACM Reference format:

Rohan Patidar, Sumit Roy, Thomas R. Henderson, and Amrutha Chandramohan. 2017. Link-to-System Mapping for ns-3 Wi-Fi OFDM Error Models. In *Proceedings of the 2017 Workshop on ns-3, Porto, Portugal, June 2017 (WNS3 2017)*, 8 pages.

DOI: <http://dx.doi.org/10.1145/3067665.3067671>

1 INTRODUCTION

Wi-Fi (IEEE 802.11 WLANs) continues to be the most popular technology for broadband wireless access as a result of its integration into a large number of consumer devices. As a result, innovations in Wi-Fi technology to enhance data rates (as well as adaptation to different use cases) have continued based on enhancements to OFDM based link design initiated in the 802.11a standard, via 802.11n and 802.11ac [5]. The latter standards have incorporated a) multi-input multi-output (MIMO) concepts based on the ability of both ends (Access Point and client devices) with regard to support (a limited number) of multiple RF front-ends, b) channel aggregation (i.e. whereby multiple 20 MHz channels can be bonded to provided larger channel widths), and c) higher order modulation schemes in order to achieve Gigabit WLAN link speeds. With the rapid evolution of 802.11 OFDM based standards, open source network simulators such as ns-3 have been unable to keep pace in terms of providing appropriate link abstractions for use in network level simulations. This work is a continuation of our prior effort in [16] to close this gap. Notably, all current link abstractions for OFDM in ns-3 are limited to additive white Gaussian noise channel (i.e. includes no results for any frequency selective fading channels).

Currently ns-3 offers two physical layer abstractions [11] in terms of achieved packet error rates (PER) for an OFDM physical layer over additive white Gaussian channels¹. The original YANS [6] PER model has been shown to be very optimistic [4], that prompted an effort for improving the Wi-Fi physical layer abstraction [16] with the intent to integrate it into a new spectrum-aware physical layer model known as *WifiSpectrumPhy*. The *WifiSpectrumPhy* enables the modeling of signal decomposition in the frequency domain and also allows signals of different technologies to coexist on the same channel. This new capability for frequency-selective channels and Wi-Fi coexistence is the basis work reported herein.

The goal of this work is to develop a full suite of simulation results for 802.11n and .11ac link layers (all modulation and coding

¹No models for any frequency selective channels are available, a situation whose amelioration is a key objective.

schemes, various frequency selective channels and channel bandwidth combinations) for use in ns-3². Accordingly, [16] developed a custom link simulator conforming to the 802.11n standard and the PERs produced were validated against IEEE Task Group-n generated simulations [10] as well as the YANS model for 802.11a. In this effort, we have used a different link simulator - based on MATLAB's 2016 release (and hence publicly available) WLAN System Toolbox [8] for 802.11n/ac. It provides options to simulate MIMO link configurations over standard AWGN and frequency selective fading channels (such as TGn [18] specified channel models). We expect this will result in upgrades to ns-3 Spectrum Wi-Fi link abstraction using the validated results presented here for 802.11n/ac. Of particular significance is enabling of network simulation for various frequency selective channels, for which we develop link-to-system mapping techniques (described in Section 3).

2 NS-3 OFDM ERROR MODELS FOR AWGN: BRIEF REVIEW

Two error models currently exist for OFDM modulation in ns-3. The original model developed for 802.11a is the YANS model [6] based on *analytical bounds* for bit/symbol error rate for coded BPSK, QPSK, and QAM modulation in AWGN channels. Another key aspect of the YANS model is the manner in which it handles packet interference. Figure 1 illustrates a sample scenario in which two frames collide in time, and the Wi-Fi receiver attempts to decode the first arriving frame. An ns-3 data structure known as the InterferenceHelper keeps track of all frame arrivals and their time durations, and at decision points during the reception process, consults an error model to calculate the packet error rate (PER). The InterferenceHelper arrives at a frame PER by subdividing the frame into different temporal regions ('chunks') and evaluating the PER on each chunk, and settles on a final frame success decision if no chunk is found to be in error. This ns-3 abstraction does not correspond to how a real physical hardware implements frame decoding, and was introduced largely for instances of time-varying SNR over a frame without sophisticated error modeling. In 2010, the YANS error model was replaced by the NIST error model (Section 3 in [9] and [13]) due to the latter's alignment with experimental data at the time³; however, the NIST PER results were significantly more pessimistic compared to YANS.

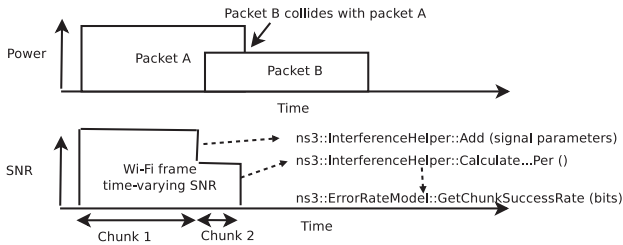


Figure 1: ns-3 error model (figure derived from ns-3 documentation)

²The current state of PER analysis of 802.11n typically provides (either upper or lower) bound on FER for complex receiver systems of interest such as Soft Viterbi decoding.

³The NIST error model also uses analytical bounds and reuses the same InterferenceHelper framework.

As a result, there have been lingering questions regarding the accuracy of both error models, as these have not aligned with recent link simulations [4], [16]. The inadequacies of the YANS model was pointed out in [4]; further differences in performance of ns-3 YANS, NIST error model and MATLAB based link simulation results of 802.11a physical layer were reported in [16]. There are (at least) two causes for the deviation of the NIST PER models as compared to definitive link simulation results: a) both rely on *analytical upper bounds* on the BER - which is accurate only at high SNRs, and further (and more crucially) b) the symbol-to-packet error rate mapping assumes *independent* errors. For all coded modulation as in 802.11a, the latter assumption results in significantly pessimistic PER rates as compared to the link simulation results. The YANS PER approximations only suffer from the first; which are generally accurate for lower MCSs and get looser for higher modulation and coding scheme as evident from the TGn simulation results in Figure 2 for 802.11n.

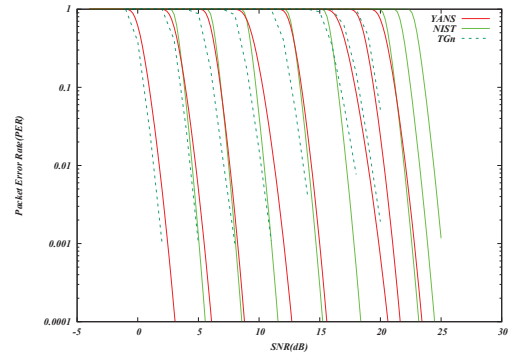


Figure 2: Comparison of ns-3 YANS, NIST and TGn for MCS 0 to 7 (left to right).

In this work, we address this lack of unification by following the methodology laid out within the current IEEE 802.11 TGax effort [15]. Our strategy is based on using a very reliable and publicly available link simulator (MATLAB WLAN System Toolbox) to generate lookup tables of PER vs SNR for each modulation and coding scheme. Since it is infeasible to generate such look-up tables for all desired packet sizes, we adopt the recommendation of TGax [15] that proposed estimating PER for any desired packet length by interpolating the results from two *reference* lengths: 32 (all lengths less than 400) bytes and 1458 (all lengths greater than 400) bytes, respectively. The interpolation formula for any length PL in terms of a reference length PL_o is given by

$$PER_{PL} = 1 - (1 - PER_{PL_o})^{PL/PL_o} \quad (1)$$

We define the (input-referenced) Signal to Noise Ratio (SNR)⁴ throughout the paper as:

$$SNR = \frac{P_{tx}}{BkT} \quad (2)$$

where P_{tx} is the transmitted power, B is *modulated (data + pilot) sub carrier bandwidth* (for 20 MHz channel $B = 20 * \frac{56}{64} MHz$), k is

⁴Note that this definition is independent of the channel characteristics.

Boltzman constant = $1.3807 \times 10^{-23} \text{ J/K}$ and T is ambient temperature in deg. Kelvin. The desire for an authoritative AWGN physical layer model for 802.11a (i.e. as a final upgrade to current default NIST) led us to exhaustively compare our (MATLAB WLAN toolbox derived) results with YANS and NIST. Thereafter, we generate *new* PER results for all 802.11n MCS and validate against the simulation curves in [10].

2.1 AWGN Simulation

For AWGN simulation, the simulation scenario is in accord with CC-59 [18], as mentioned below:

- Ideal channel with perfect estimation of channel is assumed.
- Perfect packet synchronization and packet detection is considered
- No phase tracking and phase correction taken into account
- Noise variance is known at the receiver side.
- Physical layer impairments (Phase noise, carrier frequency offset, non-linearity and others) are not included.

The diagram in Figure 3 shows all the blocks available in MATLAB WLAN System Toolbox. The number of packets needed to reliably estimate a PER value is computed using the consideration that ratio of the estimation error to the true value should be within 10% (i.e. relative error to one decimal place) with probability 0.95. Hence for each SNR value (a resolution of 0.2 dB was used in Figure 5), simulations were run until either (a minimum value of) 400 unsuccessfully decoded packets were observed or a total of 40,000 packets simulated. Note that MATLAB supports only Binary Convolution Coding (BCC) for now and LDPC is currently not supported (and hence these results are not included).

In order to validate simulation results, our results for MCS-0 to 7 (SISO) were compared with TGn results in Figure 5⁵ - they show excellent agreement for all MCSs and over the range of SNRs. Hence using the interpolation method in (1), the interpolated PER vs SNR curves for packet sizes 1000B and 200B are shown in Figure 4 and compared with MATLAB WLAN simulation results. In addition to TGn, our link simulations closely align with the YANS bound for 6 Mb/s rate of 802.11a, while those published in [4] show link simulation results more pessimistic by about 1 dB. One possible reason for the difference could be our use of soft decision decoding vs. the use of hard decision decoding.

2.2 ns-3 Implementation

The MATLAB link simulation results for 802.11n with an AWGN channel have been translated into a table-based error model (ns3 :: ErrorRateModel) for ns-3. The implementation preserves the current API between the ns-3 InterferenceHelper object and the error model, allowing the interference helper to pass in the modulation type, a SNR value, and the number of bits in the frame portion, and returning a probability of success. However, while existing error models use analytical formulas to return values for any combination of frame size and SNR value, the tabulated results are for selected fixed size packets and for SNR at discrete intervals⁶. Therefore, our

⁵The TGn data from 5 was manually transcribed from [10] for this purpose.

⁶For each MCS, we provide PER tables for frame sizes of 32 and 1458 bytes, and SNR stepped at 0.2 dB intervals.

implementation also provides for the following: 1) extrapolation of results to different frame sizes, according to Eq. 1, 2) interpolation of SNR values between available table values, and 3) ability for users to augment (with results for additional frame sizes) or replace altogether the default tables with their own. This ability to load different tables at configuration time implies the need for a table layout convention, which we have also specified; each MATLAB simulation at a particular file size produces a data file which has the mapping of SNR values to PER for MCS0 to MCS7 (extensible to higher MCSs), with SNR ranging from -4 dB to 25 dB, with separate tables for 802.11a and 802.11n/ac. Future extensions for larger bandwidth channels, multiple spatial streams, and receiver diversity can be accommodated in our framework, while currently using the base error models with correction factors applied, as is presently done. We modified the OFDM error model validation example programs in ns-3 to test also our new table-based error model implementation, and confirmed that the results align with the MATLAB WLAN System Toolbox curves of Figure 5.

3 OFDM ERROR MODEL FOR FREQUENCY SELECTIVE CHANNEL

ns-3 Wi-Fi error models do not currently support frequency selective channels. One of the main benefits of SpectrumWifiPhy is the ability to expand channel models to include frequency selectivity. For flat channels, the PER performance depends only on (input referenced) signal to noise ratio in Eq. 2, assuming that the channel power gains are normalized. Since OFDM is naturally a frequency domain modulation scheme, it is desirable to interpret PER performance as a function of the *sub-carrier* SNRs; clearly this is the same for all sub-channels for AWGN whereas they vary for frequency selective channels. As a result, the PER for frequency selective channels depends, in general, on the SNR for all sub bands, i.e. on the frequency selectivity of the channel. Hence the complexity of such a representation grows linearly with the number of sub-carriers; in the interest of a more efficient representation, the notion of link-to-system mapping via the notion of single *effective* SNR (γ_{eff}) is developed.

3.1 Link to System Mapping

Several mapping approaches have been proposed in [2], [1], [20] to estimate γ_{eff} as a single SNR representative of N sub-channel SNRs. The SNR for i^{th} subcarrier in SISO is defined as:

$$\gamma_i = \frac{P_{tx} |H_i|^2}{N \sigma_i^2} \quad (3)$$

where $\sigma_i^2 = B_{sc} kT$ with $B_{sc} = 312.5 \text{ kHz}$ represents the sub-channel bandwidth in 802.11a and N is sum total of data and pilot carriers (=52 for 802.11a).

All link to system mapping techniques can be written below, with different mapping function ϕ :

$$\gamma_{eff} = -\alpha_1 \phi^{-1} \left(\frac{1}{N_d} \sum_{i=1}^{N_d} \phi \left(-\frac{\gamma_i}{\alpha_2} \right) \right) \quad (4)$$

where γ_{eff} is effective SNR, ϕ^{-1} is inverse mapping function, N_d is number of data carriers, α_1 and α_2 are parameters to be

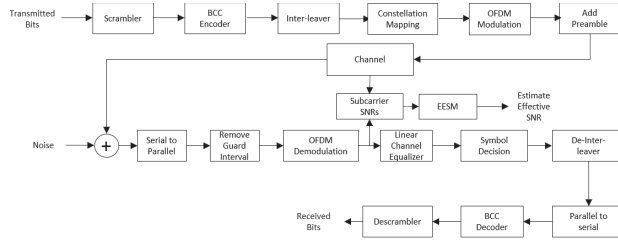


Figure 3: System block diagram for 802.11n

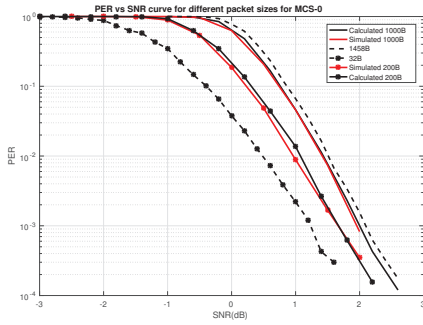


Figure 4: Validation of interpolation method for variable packet length

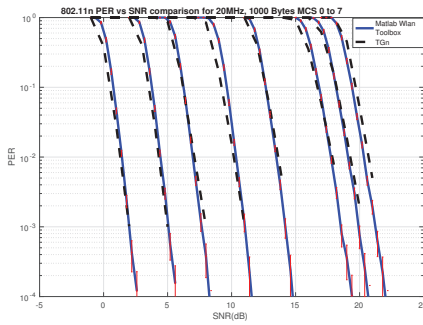


Figure 5: Validation of MATLAB WLAN System Toolbox simulations for MCS 0 to 7 against TGN [10] results.

selected. These techniques thus convert the vector of subcarrier SNRs to γ_{eff} as follows. For each combination of modulation and coding rate, α_1 and α_2 are chosen so that the resulting PER vs γ_{eff} curve approximates the PER vs SNR curve for AWGN channel, i.e., the PER under AWGN channel for SNR = γ_{eff} approximates PER for frequency selective transmission characterized by subcarrier SNRs $\gamma_1, \gamma_2, \dots, \gamma_{N_d}$. This allows reuse of PER vs SNR results of the AWGN channel to generate those for frequency selective channels. A schematic of mapping algorithms for this equivalence is shown in Figure 6.

3.1.1 *Choice of Mapping Function.* Multiple mapping functions have led to different link-to-system mappings; [2] and [21] provides comparison of the four primary methods - CESM(Capacity

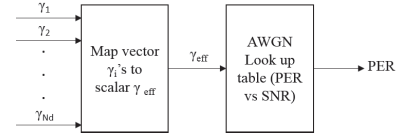


Figure 6: Functional block diagram for effective SNR mapping

effective SNR metric), EESM (Exponential effective SNR metric), MIESM (Mutual Information based effective SNR metric) and LESM (Logarithmic effective SNR metric). The mapping function may also be chosen dependent on the information rate - as captured by Received Bit Mutual Information Rate (RBIR) based on mutual information per coded bit (see Appendix B).

Unfortunately, there is no clear consensus as to which method is preferred for which scenario. While [22] shows EESM performs better than RBIR for MIMO 2x2 configuration, from a report submitted to IEEE TGax, EESM and RBIR are shown to have similar accuracy in [19]. Since EESM (described in Section 3.1.2) has one parameter and is easy to implement in comparison to RBIR with two parameters and a complex mapping function, EESM seems an appropriate choice for ns-3.

3.1.2 *EESM (Exponential Effective SNR Mapping).* EESM is derived based on Union-Chernoff bound on error probabilities [17]. When all the frequency carriers are modulated using same MCS, EESM can be used for SNR mapping. The mapping function ϕ is exponential and uses only one parameter $\beta = \alpha_1 = \alpha_2$. The EESM mapping is given by

$$\gamma_{eff} = -\beta \ln \left(\frac{1}{N_d} \sum_{i=1}^{N_d} \exp \left(-\frac{\gamma_i}{\beta} \right) \right) \quad (5)$$

3.2 MATLAB Simulation Results for EESM

A number of full link simulation runs are required to tune the EESM parameter β for each combination of modulation and coding rate. For 802.11n SISO, there are 8 such combinations MCS-0 to MCS-7. For each MCS, we performed full link simulation using 400,000 packets for each channel type (TGN Model-D,E etc) with one new channel realization for each packet ⁷. For each realization consisting of a single packet, sub-band SNRs γ_i 's calculated as per Eq. 3 are stored along with decoding result for the packet (0 for

⁷This effectively corresponds to a fast fading scenario, thus the resultant PER obtained is the average PER over all the channel realizations

correct decoding and 1 for decoding error). The following steps are then carried out to find optimal β

- (1) Initialize a value of β for EESM and calculate $\gamma_{eff}(\beta)$ for all simulated realizations, as per Eq. 5.
- (2) Combine the collection of $\gamma_{eff}(\beta)$ with corresponding decoding result for all realization. Sort values of $\gamma_{eff}(\beta)$ and quantize into 0.5 dB bins and calculate PER_j for j^{th} bin as per Eq. 6.

$$PER_j = \frac{\text{Total packets with decoding error in bin } j}{\text{Total packets in bin } j} \quad (6)$$

Let $\gamma_{eff,j}$ denote the mean of all γ_{eff} points in j^{th} bin.

- (3) Corresponding to each bin store PER_j against $\gamma_{eff,j}$ in vectors PER and Γ_{eff} respectively of length L .⁸
- (4) Interpolate AWGN table (tabulated version of Figure 5) for PER vector calculated in step-2 and store obtained SNR, in vector Γ_{AWGN} of length L .
- (5) Calculate Mean Squared Error (MSE) for the two SNR vectors:

$$\frac{1}{L} \sum_{i=1}^L (\Gamma_{AWGN}(i) - \Gamma_{eff}(\beta, i))^2 \quad (7)$$

- (6) Update β using an iterative optimization method to minimize MSE. We employ Nelder-Mead simplex direct search algorithm to update β , described in Appendix C. Move to step-3 with updated parameter, repeat for desired number of iterations (we performed 200 iterations).

In Step 3, the number of packets in a bin should be high enough to ensure small error bar for the PER_j . It is controlled by two parameters: the SNR (in Eq. 2) such that corresponding PER (in Figure 11) lies in range $[10^{-2}, 1.0]$. The other factor is the choice of the number of packets sent per selected SNR. Note that γ_{eff} simulated with a fixed SNR (Eq. 2) will vary with the channel realization (i.e new per packet). So to ensure sufficient number of packets lie in *each bin* so as to ensure desired estimation accuracy of (6), a total of 80,000 packets are sent for each SNR point (total of 5 SNR points at 2 dB spacing were simulated) as described in Appendix Sec. A. These parameter choices ensured maximum 10% error bar for PER_j down to 10^{-2} . The flowchart in Figure 7 and block diagram in Figure 8 summarize the above method to tune β for desired frequency selective channel and MCS combination.

The described steps for parameter tuning are utilized for two multipath fading channels recommended by IEEE TGn, channel models D and E, for both 20 and 40 MHz SISO configuration and a packet size of 1000 bytes. The indoor TGn channel model-D is configured for 10 m transmitter-receiver distance NLOS and no large scale fading. It uses an 18 tap channel with RMS delay spread of 50 ns which is typical for indoor scenarios. For wide-area typical outdoor scenarios, Channel Model E is configured to have 18 taps with 100 ns RMS delay, configured for 10 m transmitter receiver distance (LOS) [3]. The number of subcarriers N_d as in 5 are 56 and 114 for 20 and 40 MHz respectively, which includes both data and pilot carriers. Optimal β values are tabulated in Table 1 and 2 for each channel model with MSE given by Eq. 7.

⁸For PER_j down to 10^{-3} , vector size L is in range [7,9].

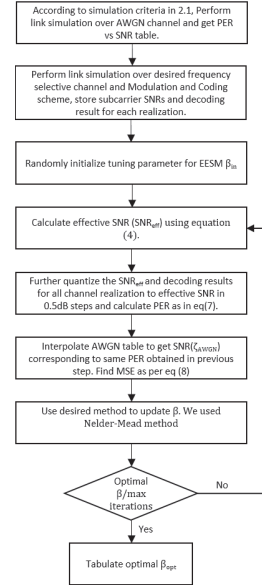


Figure 7: Flow chart for EESM parameter (β) tuning

Table 1: EESM optimal parameter for SISO channel model-D 20MHz and 40 MHz

MCS	Channel D 20 MHz		Channel D 40 MHz	
	Optimal β	MSE	Optimal β	MSE
0	1.01	0.0052	0.96	0.0120
1	2.10	0.0173	1.97	0.0105
2	1.99	0.0203	1.81	0.0029
3	7.72	0.0377	6.96	0.0225
4	9.09	0.0420	8.39	0.0380
5	32.56	0.0779	31.18	0.0788
6	33.17	0.1090	32.62	0.0667
7	37.41	0.1065	36.09	0.0720

Table 2: EESM optimal parameter for SISO channel model-E 20MHz and 40 MHz

MCS	Channel E 20 MHz		Channel E 40 MHz	
	Optimal β	MSE	Optimal β	MSE
0	0.97	0.0086	0.98	0.0038
1	2.06	0.0119	2.02	0.0147
2	1.82	0.0018	1.68	0.0082
3	7.31	0.0302	6.90	0.0147
4	8.75	0.0349	7.96	0.0025
5	32.12	0.0912	29.07	0.0640
6	31.93	0.0414	30.91	0.0284
7	35.82	0.0902	33.58	0.0222

Since the β is only dependent on MCS, β derived for one packet size (1000 bytes) can be used for other packet size values. Thus for packet size of 500 bytes, we derived β for two MCS, which is within 10% of the value for 1000 bytes. There is no significant performance difference for MSE when EESM parameters extracted for 1000 byte packet size were utilized for 500 bytes as in Table 3.

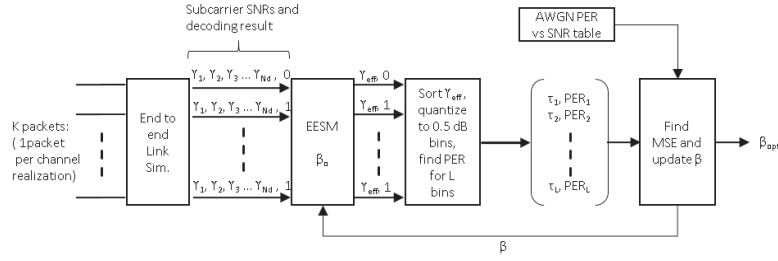


Figure 8: EESM parameter (β) tuning block diagram

Table 3: EESM optimal parameters for different packet size

MCS	β for packet size		MSE diff. if β_{1000} for 500 B
	1000 B	500 B	
0	1.01	0.99	0.001
5	32.56	30.43	0.005

The β parameter tables for simulated channels (D and E) are provided for lookup by ns-3. For frequency selective channels, MCS, channel model and bandwidth are the necessary input required to select relevant β . Should PER results be desired for a *specific channel realization*, using selected β and subcarrier γ_i 's for that channel realization, γ_{eff} is calculated, and then from the AWGN lookup table, PER against γ_{eff} obtained.

The EESM results were validated according to TGax evaluation methodology (see Step 3 for Box 0 in [15]). Figure 9 displays the results of the link simulator for 100 realizations and 1000 packets for 6 input SNRs per MCS for 20MHz model-D. The figure displays the close correspondence between actual PER observed in the link simulator (the discrete points) and the predicted PER that would have been calculated using EESM and effective SNR for the same realization (the solid curve labeled AWGN). Table 4 provides the MSE for simulated and predicted PER calculated for 600 channel realizations. Further description of the method and results can be found in [12].

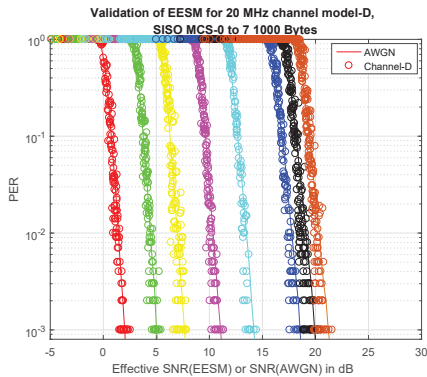


Figure 9: EESM validation for 20 MHz channel model-D, MCS 0-7

Table 4: EESM performance for channel model-D,E for 20 and 40 MHz bandwidth.

MCS Index	MSE			
	Model D		Model E	
	20 MHz	40 MHz	20 MHz	40 MHz
0	0.0013	0.0019	0.0009	0.0016
1	0.0182	0.0041	0.0326	0.0035
2	0.0097	0.0031	0.0176	0.0151
3	0.0059	0.0591	0.0207	0.0301
4	0.0030	0.0304	0.0168	0.0115
5	0.0344	0.0197	0.0095	0.0530
6	0.0061	0.0960	0.0448	0.0244
7	0.0326	0.0428	0.0169	0.0378

3.3 ns-3 Implementation

The ns-3 implementation corresponding to the above uses the SpectrumWifiPhy to subdivide the signal into OFDM subcarriers spaced at 312.5 kHz, and allocates power according to the standard use of these subcarriers (e.g. 52 subcarriers used for 802.11a). The signal is sent over a SpectrumChannel to which several propagation loss models can be chained, including distance-based path loss, shadowing, and frequency-selective fading. On the receive side, different error model tables are used for different fading channels, or the AWGN table is used if no corresponding table is found. The received packet event does not carry an indication of the channel type; this must be configured as a receiver-side error model. The received signal consists of subcarriers with different SNRs; these are combined using the EESM technique to form an effective SNR that is then used with the AWGN table. The use of different frequency-selective channels therefore corresponds to combination of beta tables and AWGN tables. The ns-3 implementation of a fading channel generator is presently incomplete, but we have support for testing against specific channel realizations produced by MATLAB.

4 CONCLUSION & FUTURE WORK

Our implementation and provision of MATLAB and ns-3 models allows for future users to generate new tables for new channels. All MATLAB and ns-3 code, and instructions for reproducing results, are publicly available⁹. Our implementation is designed to replace the existing YANS or NIST error models for those scenarios (5 GHz 802.11n OFDM) handled by the link simulations, while falling back

⁹URL: <https://bitbucket.org/rohanpatidar/wlan>

to analytical error models for other modulation types. We have not evaluated the applicability to other modulation types or future Wi-Fi versions such as 802.11ax, or legacy versions such as 802.11b, but in principle, the same approach should apply. We also have not evaluated the run-time complexity of the table based approach, but note that table-driven solutions are used elsewhere in ns-3 to improve run-time performance (32-bit CRC calculations).

Our next priority is the handling of interference cases, including frame capture effects and various frame overlap scenarios. At present, the existing ns-3 model for interference handling (InterferenceHelper) is reused, but we believe that link simulation results may result in a different frame-based interference model. The possible combination of interference (frame overlap) scenarios provides a challenge for generating all of the combinations and for efficient table representation, so a focus will be to identify important interference cases and to validate that other cases may be extrapolated from a smaller set of table results. The other significant challenge is the handling of MIMO, as the table complexity for different combinations of channels (between different antenna pairs) also may be daunting.

A ESTIMATION OF MINIMUM PACKET RUNS

For the link simulations presented, it was desired to have observed FER be accurate to within fraction x around true FER with 95% probability. Consider a binary 0 – 1 sequence of n identical independent random variables $\{X_1, X_2, \dots, X_n\}$ drawn from a distribution with mean $E[X_i] = PER$ and finite variance $Var[X_i] = \sigma^2$. Then by well-known properties of the Bernoulli distribution $\sigma^2 = PER(1 - PER)$. Hence, invoking the Law of Large Numbers: the sample average $M_n = \frac{1}{n} \sum_{i=1}^n X_i$ converges to true PER and by the Central Limit Theorem, the distribution of $\sqrt{n}|M_n - PER|$ converge to $N(0, \sigma^2)$, i.e.

$$\lim_{n \rightarrow \infty} P\left(\sqrt{n} \frac{|M_n - PER|}{\sigma} < y\right) = \phi\left(\frac{y}{\sigma}\right) \quad (8)$$

where $\phi(\cdot)$ is cumulative distribution function for standard normal distribution. We want:

$$\frac{\sigma y}{\sqrt{n}} = x * PER \quad (9)$$

The 95% probability corresponds to $y = 1.96$, and (9) yields:

$$n = \frac{Y^2 \sqrt{1 - PER}}{x^2 \sqrt{PER}} \quad (10)$$

Table 5: Choice of n per bin based on PER and error bar.

PER	n	%prob	x (error bar fraction)
0.5	384	95	0.1
0.1	3457	95	0.1
0.01	38032	95	0.1
0.005	40000	95	0.14
0.001	400000	95	0.1
0.001	40000	95	0.3

Table 5 shows the number of packets required for 95% confidence interval for PER estimate to satisfy relative error fraction x . The

optimized simulations were performed on 32-core machine, leading to run times on average of 12 hours for 1 MCS complete graph for 30 SNR points to obtain PER down to 10^{-2} with accuracy mentioned in Table 5. Note that in order to get accuracy of 10% relative error with 95% probability for $PER < 10^{-2}$, it requires roughly 120 hours to simulate a point. Due to this, the acceptable error bar size is relaxed for in this range.

B RBIR

The Mapping function ϕ for RBIR technique in SISO configuration is given as (see Appendix-1 of [15]):

$$\phi(SNR) = \log_2 M - \frac{1}{M} \sum_{m=1}^M E_U \left\{ \log_2 \left(\sum_{k=1}^M \exp[|U|^2 - |\sqrt{SNR}(s_k - s_m) + U|^2] \right) \right\} \quad (11)$$

where U is zero-mean unit-variance complex Gaussian random variable, M is the number of constellation points (dependence on MCS), s_k is constellation point with normalized energy. Further Y_{eff} is computed using Eq. 4.

C NELDER-MEAD METHOD

The Nelder-Mead method is used to find minimum of a function of several variables [7]. We apply the method to a single variable function $f(\beta)$ for which the algorithm steps are as follows:

- (1) Initialize variable x_1 , set another variable $x_2=0$.
- (2) Order variables according to function value that is $f(x_1) \leq f(x_2)$.
- (3) Perform reflection to find reflected point: $x_r = x_1 + \rho(x_1 - x_2)$ and evaluate $f(x_r)$. if $f(x_r) < f(x_1)$ proceed to Step-5 (expansion) else continue to contraction.
- (4) If $f(x_r) \leq f(x_2)$, do outside contraction: $x_c = x_1 + \gamma(x_r - x_1)$ else inside contraction as: $x_c = x_1 - \gamma(x_1 - x_2)$ and evaluate $f(x_c)$. If $f(x_c) \leq f(x_2)$, update $x_2 = x_c$, go to step-7 else go to step-6
- (5) Do expansion: $x_e = x_1 + \chi(x_r - x_1)$ and evaluate $f(x_e)$, if $f(x_e) < f(x_r)$: Update $x_2 = x_e$, else update $x_2 = x_r$. Next go to step-7.
- (6) Shrink: $x_2 = x_1 + \sigma(x_2 - x_1)$.
- (7) Stop if maximum iterations reached, else go to step-2

The parameters employed for the algorithm are standard $\rho = 1$, $\chi = 2$, $\gamma = 0.5$, $\sigma = 0.5$ provided in [7].

D CHANNEL DESCRIPTION

The TGn channel model D and E characteristics are tabulated in Table 6. The channel parameters such as tap average power, tap spacing, angular spectrum, tap delays etc are specified in [3]. Figure 10 shows average of $|H_i|^2$ over 1000 realizations for model D and E, 20 MHz (52 data carriers) and 40 MHz (108 data carriers) respectively. To validate, we compare our simulated PER vs SNR curves with the performance curves available in [14] (named as NGWL here) for SISO MCS 0 and 7, and model D in Figure 11. Our curves show better performance than NGWL as we have not considered physical layer impairments. Also it can be noted that curves for this frequency selective model are shifted rightward in comparison to Figure 5 due to the multipath fading effect.

Table 6: Profile for TGN channel model D and E

Parameters	RMS delay (ns)	Maximum delay (ns)	Rician K-Factor (dB)	Number of clusters	Number of taps
Model-D	50	390	3	3	18
Model-E	100	730	6	4	18

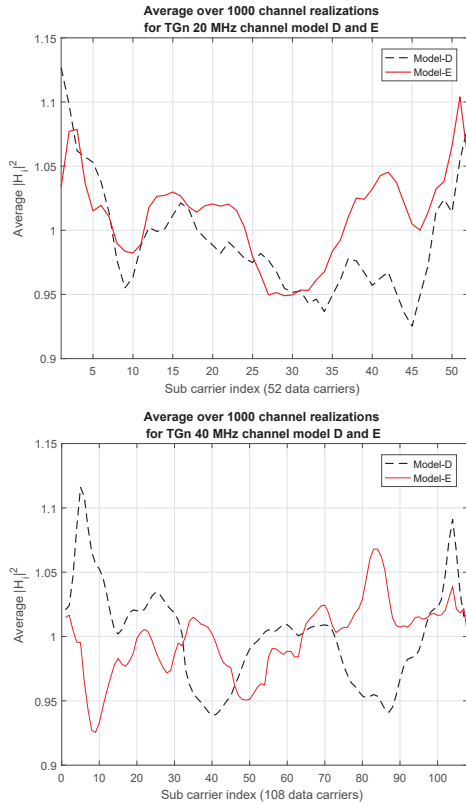


Figure 10: Average $|H_i^2|$ for bandwidth 20 and 40 MHz channel model D and E

ACKNOWLEDGEMENT

This work is supported in part by NSF CNS NeTS Award 1617153.

REFERENCES

- [1] M. Ahmed and A. Alexiou. 2003. *MTMR Baseband Transceivers Needs for Intra-system and Inter-system (UMTS/WLAN) Reconfigurability*. Technical Report. SPCOM - Signal Processing and Communications Group.
- [2] K. Brueninghaus, D. Astely, T. Salzer, S. Visuri, A. Alexiou, S. Karger, and G. A. Seraji. 2005. Link Performance Models for System Level Simulations of Broadband Radio Access Systems. In *2005 IEEE 16th International Symposium on Personal, Indoor and Mobile Radio Communications*. Berlin, Germany, 2306–2311 Vol. 4. DOI : <https://doi.org/10.1109/PIMRC.2005.1651855>.
- [3] V. Erceg. 2004. TGN Channel Models. *IEEE 802.11 document 03/940r4* (2004). URL: <https://mentor.ieee.org/802.11/dcn/03/11-03-0940-04-000n-tgn-channel-models.doc>.
- [4] C. Hepner, A. Witt, and R. Muenzner. 2015. In Depth Analysis of the ns-3 Physical Layer Abstraction for WLAN Systems and Evaluation of its Influences on Network Simulation Results. In *BW-CAR Symposium on Information and Communication Systems (SInCom)*. Konstanz, Germany, 46–51.
- [5] IEEE. 2012. IEEE 802.11 Working Group. Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications. (2012).

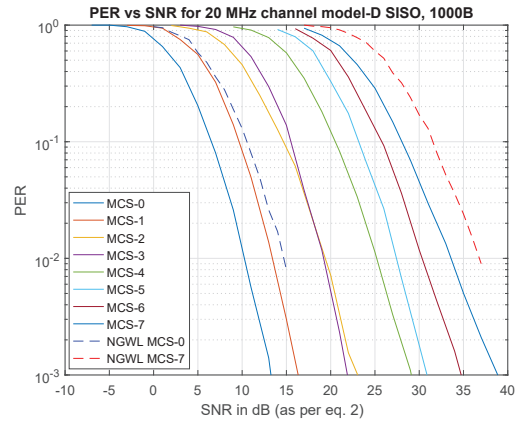


Figure 11: PER vs SNR for 20MHz channel model-D SISO, MCS 0-7

- [6] M. Lacage and T. R. Henderson. 2006. Yet Another Network Simulator. In *Proceeding from the 2006 Workshop on Ns-2: The IP Network Simulator (WNS2 '06)*. ACM, New York, NY, USA, Article 12. DOI : <https://doi.org/10.1145/1190455.1190467>.
- [7] J. C. Lagarias, J. A. Reeds, M. H. Wright, and P. E. Wright. 1998. Convergence Properties of the Nelder–Mead Simplex Method in Low Dimensions. *SIAM Journal on optimization* 9, 1 (1998), 112–147.
- [8] Matlab. 2016. MATLAB v 9.0 and WLAN System Toolbox v 1.1 . URL: <https://www.mathworks.com/help/wlan>.
- [9] LE Miller. 2003. Validation of 802.11 a/UWB Coexistence Simulation. *National Institute of Standards and Technology (NIST), WCTG white paper* (2003).
- [10] S. A. Mujtaba. 2005. TGN Sync Proposal PHY Results. *IEEE P802.11 Wireless LANs* (2005).
- [11] ns-3. 2017. ns-3 Design Documentation: Wi-Fi Module. (2017). URL: <https://www.nsnam.org/docs/models/html/wifi.html>.
- [12] R. Patidar, S. Roy, T. Henderson, and A. Chandramohan. 2017. Technical Report on Validation of Error Models for IEEE 802.11n. (2017). URL: <https://depts.washington.edu/funlab/research>.
- [13] G. Pei and T. R Henderson. 2010. Validation of OFDM Error Rate Model in ns-3. *Boeing Research Technology* (2010), 1–15.
- [14] E. Perahia and R. Stacey. 2013. *Next Generation Wireless LANs: 802.11n and 802.11 ac*. Cambridge University Press.
- [15] R. Porat. 2016. 11ax Evaluation Methodology. *IEEE P802.11 Wireless LANs* (2016). URL: <https://mentor.ieee.org/802.11/dcn/14/11-14-0571-05-00ax-evaluation-methodology.docx>.
- [16] H. Safavi-Naeini, F. Nadeem, and S. Roy. 2016. Investigation and Improvements to the OFDM Wi-Fi Physical Layer Abstraction in ns-3. In *Proceedings of the Workshop on ns-3 (WNS3 '16)*. Seattle, WA, USA, 65–70. DOI : <https://doi.org/10.1145/2915371.2915387>.
- [17] H. Song, R. Kwan, and J. Zhang. 2009. On Statistical Characterization of EESM Effective SNR over Frequency Selective Channels. *IEEE Transactions on Wireless Communications* 8, 8 (2009).
- [18] A. P. Stephens. 2004. IEEE 802.11 TGN Comparison Criteria. *IEEE P802.11 Wireless LANs* (2004). URL: <https://mentor.ieee.org/802.11/dcn/03/11-03-0814-31-000n-comparison-criteria.doc>.
- [19] F. Tong and K. Josiam. 2014. PHY Abstraction in System Level Simulation for HEW Study. (Jan 2014). URL: <https://mentor.ieee.org/802.11/dcn/14/11-14-0043-02-0hew-phy-abstraction-in-system-level-simulation-for-hew-study.pptx>.
- [20] S. Shawn Tsai and A. C. K. Soong. 2003. Effective SNR Mapping for Modelling Frame Error Rates in Multiple-state Channels. *3GPP2 - C30 - 20030429 - 010* (2003).
- [21] L. Wan, S. Tsai, and M. Almgren. 2006. A Fading-insensitive Performance Metric for a Unified Link Quality Model. In *IEEE Wireless Communications and Networking Conference, 2006.*, Vol. 4. Las Vegas, NV, USA, 2110–2114.
- [22] J. Wu, Z. Yin, J. Zhang, and W. Heng. 2010. Physical Layer Abstraction Algorithms Research for 802.11n and LTE Downlink. In *2010 International Symposium on Signals, Systems and Electronics*, Vol. 1. Nanjing, China, 1–4. DOI : <https://doi.org/10.1109/ISSSE.2010.5607125>.