

Efficient PHY Layer Abstraction for Fast Simulations in Complex System Environments

Sian Jin, *Student Member, IEEE*, Sumit Roy, *Fellow, IEEE*, Thomas R. Henderson

Abstract—Packet-level wireless network simulators face escalating system dimensionalities resulting from dense deployment scenarios supporting wideband, Multi-Input Multi-Output (MIMO), Multi-User (MU) transmission. Managing the resulting network simulation complexity and achieving practical runtimes require continuing enhancements to physical (PHY) layer abstractions. This work improves the state-of-the-art PHY layer abstractions via a new computational workflow that maps extensive offline link simulation results for OFDM/OFDMA MIMO/MU-MIMO system performance over frequency-selective i.i.d. block fading channels into Packet Error Ratio (PER) for network simulations. The proposed method is shown to require modest additional storage and the runtime is insensitive to the increase in system dimensionalities (e.g., MIMO dimensions, MU dimensions, etc.). We describe the principles of this new method and provide details about its implementation, performance, and validation.

Index Terms—Communication system modeling, fast simulations, software verification.

I. INTRODUCTION

For achieving higher network spectral efficiencies, the main wireless network families (3GPP mobile cellular networks and IEEE 802.11 WLANs) are implementing small cells (higher node density) and more advanced physical (PHY) layer technologies. For example, IEEE 802.11ax dense deployment scenarios include enterprise, multi-dwelling establishments, and crowded public hotspots (malls, airports, sports stadium, etc.); the number of antennas per device scales to 8 for Multi-Input Multi-Output (MIMO) operation and channel bandwidths scale up to 160 MHz [2]. In addition, new modes of Multi-User (MU) transmission are introduced, based on Orthogonal Frequency Division Multiple Access (OFDMA) and Multi-User MIMO (MU-MIMO). Performance evaluation of such (even modest scale) networks in various operational scenarios of interest via pure mathematical analyses is infeasible and field trials are costly [3]. This leaves system-level (network) simulations as the most feasible option, rendering network simulators like ns-3¹ indispensable for investigating wireless network performance [3] as a function of network dimensionality.

With increasing node density and complexity in the PHY layer, associated computational complexity is a major concern for network simulators [3]. System-level simulations quantify the network performance (e.g., throughput) from the packet-level performance metrics such as *instantaneous (per-packet)*

Packet Error Ratio (PER) [4]. In contrast, *full PHY (link) simulations*, typically run on link simulators, obtain packet performance at a single receiver, using runtime-costly symbol-level simulation of the PHY layer. Running a full PHY simulation that involves generating channel realizations and transceiver signal processing is impractical within a network simulator. As shown in Table I, the average runtimes of a full PHY simulation for 40000 PHY layer packets² on a single Orthogonal Frequency Division Multiplexing (OFDM) MIMO link require order of hour runtimes. Thus, system-level simulation must incorporate suitable *PHY layer abstractions* that represent the PHY layer performance with sufficient accuracy to be incorporated within network simulators at runtime. The PHY layer abstraction in a network simulator is the packet error model that produces a decision on whether the packet is successfully received or not based on the PHY layer setup including the received (RX) signal/interference/noise power, the modulation and coding type, the channel models, etc.

TABLE I: Average runtime comparison using MATLAB WLAN Toolbox on Intel Core i5 CPU at 2.0GHz (averaged over 10 trials): 40000-packet simulation for a single-link IEEE 802.11ax OFDM MIMO with full spatial streams under a RX SNR [7].

$n_t \times n_r$	Bandwidth	Full PHY	Traditional PHY Layer Abstraction [8]
1×1	20MHz	40 min	16 min
1×1	40MHz	46 min	20 min
4×2	20MHz	97 min	36 min
4×2	40MHz	132 min	44 min
8×2	20MHz	180 min	68 min
8×2	40MHz	295 min	76 min

The existing ns-3 Wi-Fi PHY layer abstraction - the YANS model [9] - was developed for the OFDM Single-Input Single-Output (SISO) transmission over Additive White Gaussian Noise (AWGN) channel. However, WiFi systems have progressed to MIMO transmission over wideband frequency-selective fading channels. To realize PHY layer abstraction under such complex scenarios, *link-to-system* (L2S) mapping has been widely adopted [4], [8], [10]–[12]. As shown in Fig. 1, realizations of the frequency-selective fading channel matrices, the MIMO precoding (i.e., multi-stream transmit beamforming [13]) matrices and the MIMO decoding matrices

S. Jin, S. Roy and T. R. Henderson are with the Department of Electrical and Computer Engineering, University of Washington, Seattle, WA, 98195 USA (e-mail: sianjin@uw.edu; sroy@uw.edu, tomhend@u.washington.edu). This work was partly presented in WNS3 2020 [1].

¹<https://www.nsnam.org>

²The sample mean $\frac{1}{n} \sum_{i=1}^n Y_{k,i}$ is used as estimator of the average PER $p_{avg,k}$ at receiver k , where $Y_{k,i} \in \{0, 1\}$ is the binary packet error state of the i -th packet and n is the number of packets in a simulation. By [5], $n \geq \frac{400(1-p_{avg,k})}{p_{avg,k}}$ can achieve PER estimation accuracy bound $P \left[\left| \frac{1}{n} \sum_{i=1}^n Y_{k,i} - p_{avg,k} \right| \leq 0.1 p_{avg,k} \right] \geq 0.95$. Thus, for estimating PER down to 10^{-2} [6] while achieving the accuracy described above, 40000-packet runs are used in each full PHY simulation.

from the desired and interfering sources are generated for each packet. The channel, precoding and decoding matrices determine the packet-level link performance, which is fully characterized by a matrix of post-MIMO processing Signal-to-Interference-plus-Noise-Ratios (SINRs) on different sub-carriers and different MIMO streams. Thereafter, a suitable L2S mapping function is needed to translate the post-MIMO processing SINR matrix into a *single* scalar metric - the *effective SINR* [14]. The effective SINR is a random variable that summarizes the impact of the frequency-selective MIMO channel, beamforming gain, and any interference. The effective SINR can be used to predict the instantaneous (per-packet) PER on the link by looking up from a pre-stored PER-SNR table for the AWGN-SISO channel [15]. As the instantaneous PER is a function of the effective SINR, it is also a random variable³. The instantaneous PER is used in a Bernoulli trial to decide whether the current packet is successfully received or not [8]. The accuracy of the instantaneous PER prediction can be guaranteed via optimizing the L2S mapping tuning parameters.

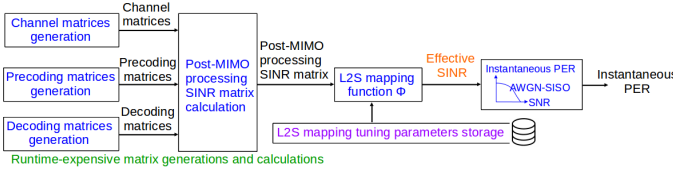


Fig. 1: Flow chart of traditional PHY layer abstraction in a network simulator.

The PHY layer abstraction in Fig. 1 is suggested by the IEEE TGax group [8] and has been developed for OFDM SISO [5], [15], OFDM MIMO [1], [17], and OFDM MU-MIMO systems [18]. However, our prior contribution in [1] found while these earlier PHY layer abstractions do reduce runtimes compared with full PHY simulation, they still suffer from scaling with MIMO dimensions (the number of transmit antennas and the number of receive antennas), MU dimensions (the number of users that are simultaneously served), bandwidth, and the number of interferers; see Table I that shows how average runtimes scale with the number of transmit (receive) antennas $n_t(n_r)$ and bandwidth. The reason is that these PHY layer abstractions require generating channel, precoding and the decoding matrices, and calculating post-MIMO processing SINR matrices online. These operations involve expensive matrix calculations that scale with MIMO dimensions, MU dimensions, bandwidth, and the number of interferers. This motivates us to create a new PHY layer abstraction whose runtime is insensitive to the above dimensionality variations.

Our approach is driven by a key underlying question: since the link performance (instantaneous PER) only needs the effective SINR, can we bypass channel generation, precoder calculation, decoder calculation as well as post-MIMO processing SINR matrix calculation steps, and directly model effective SINR? Under frequency-selective independent and

identically distributed (i.i.d.) block fading channels, we discover that the effective SINR distribution under Exponential Effective SINR Mapping (EESM) L2S mapping [5] can be well approximated by a 4-parameter distribution called *log-SGN distribution*. Therefore, under EESM L2S mapping, we only need to store 4 log-SGN parameters under a specific PHY layer setup (e.g., channel type, MIMO dimension, MCS, etc.) and draw effective SINR samples from this specific log-SGN distribution *directly*. We call such an approach the EESM-log-SGN PHY layer abstraction, whose flow chart is shown in Fig. 2. The payoff is significant: the runtime of the traditional PHY layer abstraction in Fig. 1 scales as system dimensionality increases, while the runtime of the EESM-log-SGN PHY layer abstraction in Fig. 2 is much reduced and is insensitive to the dimensionality change. A side benefit is that we do not have to build increasingly complex channel realizations (e.g., IEEE TGax channel [19], 3GPP TR 38.901 5G channel [20], 3GPP TR 37.885 V2X channel [21], etc.) within the network simulator.

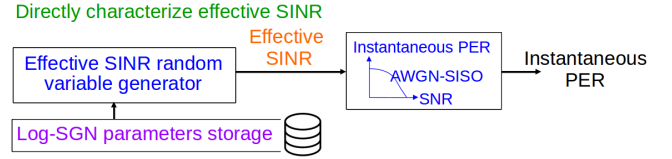


Fig. 2: Flow chart of the proposed EESM-log-SGN PHY layer abstraction in a network simulator.

Nonetheless, the proposed EESM-log-SGN PHY layer abstraction faces two storage-complexity challenges. First, the log-SGN parameters are sensitive to changes in the interference setup (e.g., the distances of interferers to the receiver). Storing log-SGN parameters for each interference scenario leads to high storage complexity. This challenge is mitigated by our discovery that the distributions of effective Signal-to-Noise-Ratio (SNR) and effective Interference-to-Noise-Ratio (INR) can be characterized and stored efficiently based on EESM-log-SGN PHY layer abstraction. Therefore, we propose a Low-Storage Complexity version of EESM-log-SGN (EESM-log-SGN-LSC) PHY layer abstraction that estimates effective SINR based on effective SNR and effective INR. The second challenge is that the log-SGN parameters for effective SNR/INR are sensitive to the change of RX SNR/INR (i.e., transmit SNR of the desired signal/interference subtracted by the path loss from the desired transmitter/interferer to the receiver in dB). In [1], we suggested storing log-SGN parameters of effective SNR for different RX SNRs with a granularity of 0.25dB, resulting in roughly 100 log-SGN parameter sets. This leads to high storage complexity and additional full PHY simulation burdens. In this work, we solve this issue by proposing an EESM-log-SGN *mixture* model that estimates effective SNR/INR distributions for the desired range of RX SNRs/INRs using stored effective SNR/INR distributions for only a few RX SNR/INR values. The proposed two low-storage-complexity solutions greatly facilitate the implementation of the proposed EESM-log-SGN PHY layer abstraction in network simulators.

³As in [4], we use ‘instantaneous’ to indicate the random property of per-packet PER. This contrasts with the average PER that is a constant equaling to the ensemble averaging [16] of the random instantaneous PER.

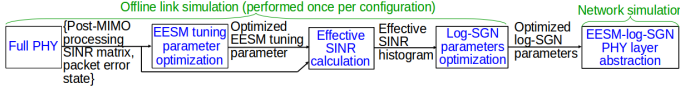


Fig. 3: Simulation steps for implementing the proposed EESM-log-SGN PHY layer abstraction.

The implementation flow of EESM-log-SGN is shown in Fig. 3, which consists of an offline link simulation part using a link simulator (MATLAB in our case) and the actual network simulation part using a network simulator. The offline link simulation part is conducted once per configuration and generates log-SGN parameters for use at runtime in network simulations. The offline simulation implements a full PHY using MATLAB WLAN Toolbox under the setup introduced in Section II-A, and outputs post-MIMO processing SINR matrix as well as its associated binary packet error state. Using multiple full PHY simulation outputs, the EESM tuning parameter optimization block introduced in Section II-B generates the optimized EESM tuning parameter. Using the optimized EESM tuning parameter and multiple post-MIMO processing SINR matrices, we obtain corresponding effective SINR defined in Section II-B and statistically fit the effective SINR histogram with the log-SGN distribution introduced in Section III. The log-SGN parameters of the fitting are stored in a network simulator for further use in network simulation. The network simulation part in Fig. 3 uses the EESM-log-SGN method to directly draw the instantaneous PER. The basic EESM-log-SGN method is introduced in Section III while the low-storage-complexity EESM-log-SGN-LSC method and EESM-log-SGN mixture model are introduced in Section IV. All MATLAB codes and detailed coding guides related to this work are available at [7].

II. PHY LAYER ABSTRACTIONS FOR OFDM/OFDMA MIMO/MU-MIMO SYSTEM

A. OFDM/OFDMA MIMO/MU-MIMO System Setup

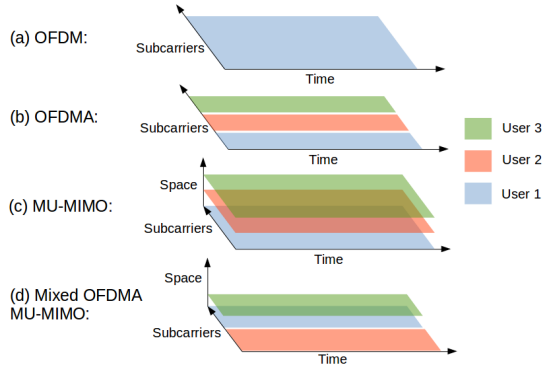


Fig. 4: Illustration of OFDM, OFDMA, MU-MIMO and mixed OFDMA MU-MIMO [22].

We review the basic principles of OFDMA, MU-MIMO, and mixed OFDMA MU-MIMO as relevant to this work. OFDM is a modulation scheme that modulates data symbols

in parallel on multiple subcarriers [23]. As shown in Fig. 4 (a), all subcarriers in a channel (e.g., 20MHz for a typical Wi-Fi channel) are allocated to a single user at a time [23]. OFDMA is a multiple access scheme based on OFDM, whereby - as shown in Fig. 4 (b) - different subsets of subcarriers are assigned to different users [23]⁴. MU-MIMO is a multi-user extension of MIMO - shown in Fig. 4 (c) - that serves multiple receivers on the same time-frequency (subcarrier) resources; the data symbols intended for different receivers are separated in space via precoding (i.e., multi-stream transmit beamforming [13]). Mixed OFDMA MU-MIMO, as shown in Fig. 4 (d), can serve multiple receivers on a subset of subcarriers via precoding at the same time [22].

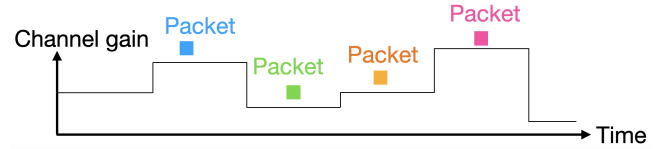


Fig. 5: Frequency-selective i.i.d. block fading channel.

In this work, the frequency-selective channel instance is assumed to be almost invariant during the transmission of each packet (e.g., IEEE TGax channel models [19]) and different channel instances for different packets are assumed to be i.i.d. [4], [25], [26]. An illustrative example of such frequency-selective i.i.d. block fading [14] channels used in our simulation is shown in Fig. 5. Under the considered block fading channels, the Doppler shift is assumed to be small (e.g., IEEE TGax channel models [19]) such that inter-carrier interference can be ignored. Noise at each receiver is assumed to be AWGN [4], with identical power over different subcarriers. We consider the downlink OFDM/OFDMA MIMO/MU-MIMO transmission with perfect synchronization, perfect phase tracking, and Channel State Information at the Receiver/Transmitter (CSIR/CSIT) determined from noise-free channel estimate [4]. We also consider a time-frequency synchronous (fully overlapped) interference model (see Fig. 6), where interference packets occupy the same time slots and subcarriers as the desired packet, such that the interference event is considered invariant over the packet [27].

Suppose $n_{ss,k}$ spatial streams (independent information flows) are transmitted from the desired transmitter to receiver k using the set of subcarriers $\mathcal{N}_{sc,k}$, and K receivers share the same set of subcarriers, i.e., $\mathcal{N}_{sc,k} = \mathcal{N}_{sc}, k = 1, 2, \dots, K$. On each subcarrier $i \in \mathcal{N}_{sc}$, the modulated $n_{ss,k}$ spatial streams are then mapped into n_t transmit antennas using a $n_t \times n_{ss,k}$ MIMO precoding matrix $\mathbf{F}_{k,i}$ for receiver k and subcarrier i [14]. Using CSIT of the desired channel, MIMO precoding can provide transmit beamforming gain (power gain and/or transmit diversity gain) in multi-stream MIMO systems [6], [13], and can suppress inter-user interference in downlink MU-MIMO broadcast channels (see Fig. 6) [6].

⁴In Fig. 4 (b), the OFDMA subcarriers are contiguous in frequency, which is true for the IEEE 802.11ax system. However, in general (e.g., OFDMA in 5G NR [24]), non-contiguous subcarriers can also be used in OFDMA.

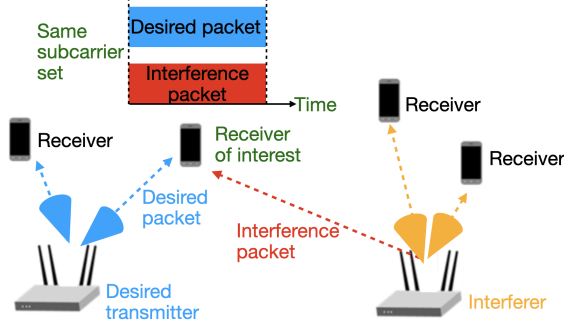


Fig. 6: Example of OFDM/OFDMA MIMO/MU-MIMO system setup with interference.

The desired transmitted packet is passed through the $n_{r,k} \times n_t$ frequency-domain channel matrices $\mathbf{H}_{k,i}, i \in \mathcal{N}_{sc,k}$ and arrives at the $n_{r,k}$ receive antennas of receiver k . The frequency-domain channel matrices $\mathbf{H}_{k,i}, i \in \mathcal{N}_{sc,k}$ considered in this work capture two key components of OFDM/OFDMA MIMO/MU-MIMO channels: frequency-selectivity (the property that $\mathbf{H}_{k,i}$ varies with subcarrier index i under a frequency-selective channel) and antenna correlation (the correlation between entries within $\mathbf{H}_{k,i}$, which might cause rank reduction in MIMO channel matrix) [6], [14], [23]. In later sections, we validate our methods using IEEE TGax channel model-D, which incorporates channel frequency selectivity and antenna correlation, and is widely used for IEEE 802.11 system performance evaluation [6].

As shown in Fig. 6, each receiver receives desired packets under potential time-frequency synchronous interference packets. The interference packets to the victim receiver are transmitted via the side-lobe of the interfering beam, as the precoder at the interferer is not designed for the channel from the interferer to the victim receiver. Each receiver uses linear MIMO decoding to recover the desired signal. At receiver k 's i -th subcarrier $i \in \mathcal{N}_{sc,k}$ and for stream $j \in \{1, 2, \dots, n_{ss,k}\}$, the post-MIMO processing SINR $\Gamma_{k,i,j}$ is [8]

$$\Gamma_{k,i,j} = \frac{S_{k,i,j}}{I_{k,i,j}^s + I_{k,i,j}^o + N_{k,i,j}}, \quad (1)$$

where $S_{k,i,j}$ is the received signal power, $I_{k,i,j}^s$ is the inter-stream interference, $I_{k,i,j}^o$ is the interference from other interferers, and $N_{k,i,j}$ is the post-MIMO processing noise power. At subcarrier i , let $\mathbf{W}_{k,i}$ denote the $n_{r,k} \times n_{ss,k}$ linear MIMO decoding matrix of receiver k ; let $\mathbf{F}_i^u$ denote the precoding matrix of interferer u ; let $\mathbf{H}_{k,i}$ denote the channel matrix from interferer u to receiver k . By [8], we have $S_{k,i,j} = P_{t,k} |[\mathbf{W}_{k,i}]_j^* \mathbf{H}_{k,i} [\mathbf{F}_{k,i}]_j|^2$, $I_{k,i,j}^s = P_{t,k} |[\mathbf{W}_{k,i}]_j^* \mathbf{H}_{k,i} \mathbf{F}_{k,i}|^2 - S_{k,i,j}$, $I_{k,i,j}^o = \sum_{u \neq k} P_{u,k} |[\mathbf{W}_{k,i}]_j^* \mathbf{H}_{k,i} \mathbf{F}_i^u|^2$, and $N_{k,i,j} = \sigma_k^2 |[\mathbf{W}_{k,i}]_j|^2$, where $P_{t,k}$ is the received signal power at receiver k from the desired transmitter, $P_{u,k}$ is the received signal power at receiver k from interferer u , σ_k^2 is additive noise power on each subcarrier of receiver k , $[\cdot]_j$ denotes the j -th column of a matrix, and $\|\cdot\|$ is the Euclidean norm of a vector. The post-MIMO processing SINR matrix at receiver k is defined by $\mathbf{\Gamma}_k \triangleq (\Gamma_{k,i,j})_{i \in \mathcal{N}_{sc,k}, 1 \leq j \leq n_{ss,k}}$, which reflects

channel frequency-selectivity, antenna correlation captured by $\mathbf{H}_{k,i}, i \in \mathcal{N}_{sc,k}$, and transmit beamforming gain captured by $\mathbf{F}_{k,i}, i \in \mathcal{N}_{sc,k}$.

B. Overview of Traditional PHY Layer Abstraction

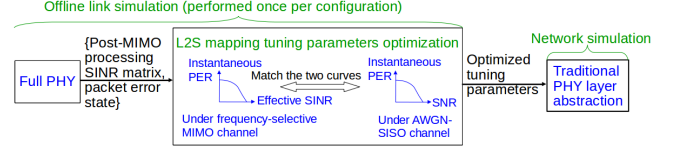


Fig. 7: Simulation steps for implementing traditional PHY layer abstraction.

A network simulator requires PHY layer abstraction to generate accurate link performance (instantaneous PER). To implement the traditional PHY layer abstraction suggested by the IEEE TGax group [8], the workflow in Fig. 7 is required. This workflow consists of the offline link simulation and the network simulation parts [1], [5]. The link simulation is conducted using a credible link simulator (e.g., MATLAB WLAN Toolbox in our case) to provide calibrated L2S mapping tuning parameters. Thereafter, a PHY layer abstraction with calibrated L2S mapping tuning parameters is used at runtime within a suitable network simulator (e.g., ns-3 in our case).

In the link simulation part of the implementation, a full PHY simulation is first conducted to obtain the post-MIMO processing SINR matrix and the associated binary packet error state (1 indicates packet error and 0 indicates packet success) for each packet. Under the OFDM/OFDMA MIMO/MU-MIMO setup specified in Section II-A, a closed-form post-MIMO processing SINR matrix $\mathbf{\Gamma}_k$ is given by (1), and the associated binary packet error state is determined by the full PHY simulation. The set of {post-MIMO processing SINR matrix $\mathbf{\Gamma}_k$, binary packet error state} pairs are then provided as inputs into the second block in Fig. 7 for optimizing the tuning parameters. In this block, an L2S mapping function Φ compresses the post-MIMO processing SINRs on different subcarriers and streams into a *single* metric called *effective SINR*, which would yield the same instantaneous PER if the simulation was run for an AWGN-SISO channel. The effective SINR at receiver k is [4], [8], [10]–[12]

$$\Gamma_{eff,k}^{sinr} = \alpha \Phi^{-1} \left(\frac{1}{n_{sc,k}} \frac{1}{n_{ss,k}} \sum_{i \in \mathcal{N}_{sc,k}} \sum_{j=1}^{n_{ss,k}} \Phi \left(\frac{\Gamma_{k,i,j}}{\beta} \right) \right), \quad (2)$$

where Φ^{-1} is the inverse L2S mapping function, $\mathcal{N}_{sc,k}$ is the set of subcarriers for receiver k , $n_{sc,k} \triangleq |\mathcal{N}_{sc,k}|$ is the number of subcarriers allocated to receiver k , $n_{ss,k}$ is the number of spatial streams sent to receiver k , α and β are L2S mapping tuning parameters that depend on PHY layer configurations (channel type, OFDM/OFDMA MIMO/MU-MIMO setup, MCS and channel coding). The effective SINR is a generalization of *effective SINR* $\Gamma_{eff,k}^{sinr}$, which is for interference-free cases (i.e., $I_{k,i,j}^o = 0$ in (1)) [1], [5]. Abstracting the post-MIMO processing SINRs over all subcarriers and

spatial streams using a single effective SINR greatly simplifies the link-to-system interface [28]. The single effective SINR is a convenient metric to describe the packet-level performance for a network simulator, at the cost of losing detail for each spatial stream (symbol-level performance). For example, in the case of the ill-conditioned MIMO channel (low-rank MIMO channel matrix) due to high antenna correlation [23], a low effective SINR does not show which spatial streams have low post-MIMO processing SINRs. This issue can be potentially addressed by modeling different effective SINRs for different spatial streams, but this leads to higher PHY layer abstraction complexity and is a matter left for future investigation.

The tuning parameters α and β in (2) are optimized so that the Mean Square Error (MSE) between the instantaneous PER-effective SINR curve for the simulated frequency-selective fading channel and the instantaneous PER-SNR curve under the AWGN-SISO channel is minimized (see Fig. 7). Once the tuning is complete, the optimized parameters are stored for use in network simulations at runtime.

The current state-of-the-art PHY layer abstraction flow chart is shown in Fig. 1 and is referred to as the *traditional* PHY layer abstraction. The traditional PHY layer abstraction requires the network simulator to implement a channel matrix generator. Precoding matrices and MIMO decoding matrices are calculated based on the channel matrices. Using the channel matrices, precoding matrices, and MIMO decoding matrices, (1) is used to calculate post-MIMO processing SINRs. Then, (2) with optimized tuning parameters is used to calculate the effective SINR $\Gamma_{eff,k}^{sinr}$. The effective SINR $\Gamma_{eff,k}^{sinr}$ is mapped into an instantaneous PER $P_{ins,k}$ using the instantaneous PER-SNR curve (lookup table) under the AWGN-SISO channel at the specified packet length, channel coding and MCS, and each packet fails in decoding with probability $P_{ins,k}$.

The accuracy of the instantaneous PER prediction depends on the L2S mapping function Φ , of which there are several. Two widely adopted L2S mapping functions are Exponential Effective SINR Mapping (EESM) and Received Bit Information Rate (RBIR) mapping. For EESM L2S mapping, $\alpha = \beta$ and L2S mapping function $\Phi(x) = \exp(-x)$ [5], [11]. Then, (2) reduces to

$$\Gamma_{eff,k}^{sinr} = -\beta \ln \left(\frac{1}{n_{sc,k}} \frac{1}{n_{ss,k}} \sum_{i \in \mathcal{N}_{sc,k}} \sum_{j=1}^{n_{ss,k}} \exp \left(-\frac{\Gamma_{k,i,j}}{\beta} \right) \right). \quad (3)$$

For RBIR L2S mapping, $\alpha \neq \beta$ in general, and the L2S mapping function for M-QAM modulation is given by [8] $\Phi(x; M) = \log_2 M - \frac{1}{M} \sum_{m=1}^M \mathbb{E}_Z [\log_2 (\sum_{k=1}^M \exp[|Z|^2 - |\sqrt{x}(s_k - s_m) + Z|^2])]$, where $Z \sim \mathcal{CN}(0, 1)$, s_k and s_m are constellation points with normalized (unit) energy [10]. By [28], $\Phi(x; M)$ for RBIR is upper bounded by the mutual information under M-QAM modulation, i.e., $\Phi(x; M) \leq \log_2 M$.

C. Validation of EESM and RBIR L2S Mappings

We next validate EESM and RBIR L2S mappings for the general OFDM/OFDMA MIMO/MU-MIMO systems. This

is achieved by demonstrating that the instantaneous PER-effective SINR curve under EESM/RBIR L2S mapping fits the instantaneous PER-SNR curve (lookup table) for AWGN-SISO channel (see Fig. 7). The traditional PHY layer abstractions are shown to be effective, with an example under a mixed OFDMA MU-MIMO setup shown in Fig. 8 and other simulation parameters shown in Table II. In this example, the AWGN-SISO curve shows the relationship between the input SNR and resulting instantaneous PER from full PHY AWGN-SISO simulations conducted at 0.25dB SNR increments. The EESM/RBIR points show the relationship between the EESM/RBIR based effective SNR/SINR on every 0.25dB bin [8] and the full PHY simulated instantaneous PER in each bin. The instantaneous PERs following those on the AWGN-SISO curve indicate that we can successfully predict instantaneous PERs using the AWGN-SISO lookup table and effective SNRs/SINRs.

TABLE II: PHY layer simulation setup.

Communication system	IEEE 802.11ax
Link simulator	MATLAB WLAN Toolbox R2020b
Number of packets/simulation	40000
Channel type	IEEE TGax channel model-D [19]
Channel for each packet	i.i.d.
Speed of the scatters/users	0.089km/h
Channel coding	LDPC
Payload length	1000
MCS	4
Bandwidth	20 MHz (242 subcarriers in total)
Channel estimation	Noise-free
OFDMA subcarriers/user	26 or 52 or 106 or 242
MIMO/MU-MIMO PHY	SVD/ZF precoding, MMSE decoding
Multi-user power allocation	Uniform
CPU	Intel Core i5 CPU at 2.0GHz

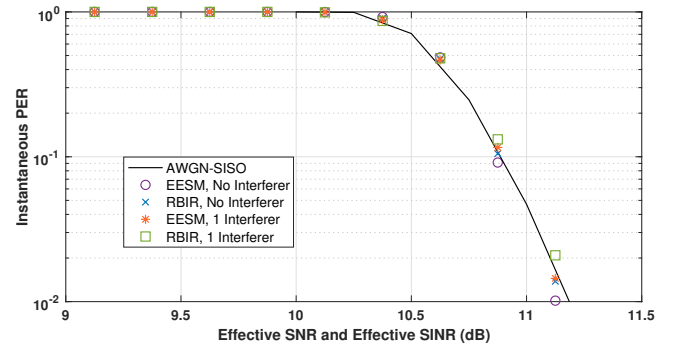


Fig. 8: Validation of EESM and RBIR L2S mappings under a setup where a receiver shares 106 subcarriers with another receiver using $8 \times \{2, 2\}$ MU-MIMO with 2 streams/user. For the interference case, the RX INR is 30dB lower than RX SNR.

D. Simulation Runtime Evaluation

We compare runtimes for full PHY simulation versus traditional EESM/RBIR PHY layer abstraction simulation under the setups in Table II. Consistent with the notation in [29], we use $\{n_{sc}, n_t \times \{n_{r,1}, \dots, n_{r,K}\} : \{n_{ss,1}, \dots, n_{ss,K}\}\}$ to denote that 1 transmitter with n_t antennas simultaneously transmits to K receivers using MU-MIMO on n_{sc} subcarriers,

where receiver k has $n_{r,k}$ receive antennas and receives $n_{ss,k}$ streams. If OFDMA is adopted, the notation $//$ is used to separate receivers on different subcarriers. For example, $\{106, 8 \times 2 : 2\} // \{106, 8 \times \{2, 2\} : \{2, 2\}\}$ means that 1) the transmitter with 8 antennas transmits to 3 receivers; 2) each receiver has 2 antennas and receives 2 streams; 3) the first user occupies 106 subcarriers and the last two users share another 106 subcarriers. As runtime fluctuation for any setup is very small on the same machine, we only average over 10 runs each comprising of 40000 packets (average PER down to 10^{-2} with high accuracy)⁵ and record the average runtimes in Table III.

TABLE III: Average runtime comparison between the full PHY simulation and traditional EESM and RBIR PHY layer abstractions for running a 40000-packet simulation at a specific RX SNR in MATLAB [7].

PHY Layer Setup	Full PHY	EESM/RBIR
$\{242, 1 \times 1 : 1\}$, 1-user OFDM SISO, no interferer	44 min	16 min
$\{242, 4 \times 2 : 2\}$, 1-user OFDM MIMO, no interferer	107 min	40 min
$\{242, 8 \times 2 : 2\}$, 1-user OFDM MIMO, no interferer	196 min	72 min
$\{242, 8 \times 2 : 2\}$, 1-user OFDM MIMO, 1 interferer	378 min	155 min
$\{242, 8 \times 2 : 2\}$, 1-user OFDM MIMO, 2 interferers	538 min	230 min
$\{106, 1 \times 1 : 1\} // \{52, 1 \times 1 : 1\} // \{52, 1 \times 1 : 1\}$, 3-user OFDMA SISO, no interferer	88 min	28 min
$\{106, 8 \times 2 : 2\} // \{52, 8 \times 2 : 2\} // \{52, 8 \times 2 : 2\}$, 3-user OFDMA MIMO, no interferer	472 min	152 min
$\{242, 8 \times \{2, 2\} : \{2, 2, 2\}\}$, 3-user OFDM MU-MIMO, no interferer	365 min	160 min
$\{106, 8 \times 2 : 2\} // \{106, 8 \times \{2, 2\} : \{2, 2\}\}$, 3-user OFDMA MU-MIMO, no interferer	398 min	156 min
$\{106, 8 \times 2 : 2\} // \{106, 8 \times \{2, 2\} : \{2, 2\}\}$, 3-user OFDMA MU-MIMO, 1 interferer	790 min	316 min

From Table III, we have the following observations:

- The traditional EESM/RBIR PHY layer abstraction simulations run faster than the full PHY simulations.
- The runtimes for EESM PHY layer abstraction simulations are almost equal to RBIR PHY layer abstraction simulations, which is consistent with [30].
- The runtime of the traditional EESM/RBIR PHY layer abstraction simulation scales with the MIMO dimensions (the number of transmit antennas and the number of receive antennas), the MU dimensions (the number of simultaneously served users), and the number of interferers.

The runtimes of the traditional EESM/RBIR PHY layer abstraction in Fig. 1 are large because they require generating channel matrices, precoding matrices as well as decoding matrices, and calculating post-MIMO processing SINR matrices online. These operations involve expensive matrix calculations (e.g., matrix operations for obtaining (1)) that scale with MIMO dimensions, MU dimensions, and the number of

interferers. This implies that we need a more efficient PHY layer abstraction to achieve low network simulation runtimes.

III. EFFICIENT PHY LAYER ABSTRACTION USING EFFECTIVE SINR DISTRIBUTION

Our method rests on the key observation that the necessary PHY layer abstraction for network simulation only needs the effective SINR and its mapped instantaneous PER. Hence, we seek to bypass individual channel generation, precoder as well as decoder calculation, and post-MIMO processing SINR matrix calculation steps in Fig. 1, and directly model the distribution of effective SINR.

A. Modeling Distribution of Effective SINR

We first focus on finding an approximate distribution of $\Gamma_{eff,k}^{sinnr}$ under EESM L2S mapping, based on evidence from our simulations in Fig. 9 and related works. First, under IEEE TGax channel models, even when subcarrier index difference $|i - i'|$ is moderately large, the correlation coefficient between $\Gamma_{k,i,j}$ and $\Gamma_{k,i',j}$ (i.e., $\rho(\Gamma_{k,i,j}, \Gamma_{k,i',j})$) is still significant. For example, in Fig. 9 (a), $\rho(\Gamma_{k,i,1}, \Gamma_{k,1,1}) \geq 0.6$ when $i < 25$. Since $\Gamma_{k,i,j}, i \in \mathcal{N}_{sc,k}, j = 1, \dots, n_{ss,k}$ do not satisfy the weakly dependent condition in [31] and $n_{sc,k}$ is not sufficiently large, the use of Central Limit Theorem (CLT) for approximating the Probability Density Function (PDF) of $\frac{1}{n_{sc,k}} \frac{1}{n_{ss,k}} \sum_{i \in \mathcal{N}_{sc,k}} \sum_{j=1}^{n_{ss,k}} \exp\left(-\frac{\Gamma_{k,i,j}}{\beta}\right)$ in (3) is not accurate. By simulation - see Fig. 9 (b) under IEEE TGax channel model-D - the resulting PDF is highly skewed (closer to Chi-square distribution under some special cases [32]). Thus, the normal or Beta approximation for $\frac{1}{n_{sc,k}} \frac{1}{n_{ss,k}} \sum_{i \in \mathcal{N}_{sc,k}} \sum_{j=1}^{n_{ss,k}} \exp\left(-\frac{\Gamma_{k,i,j}}{\beta}\right)$ based on CLT [33]–[35] no longer holds for IEEE TGax channels.

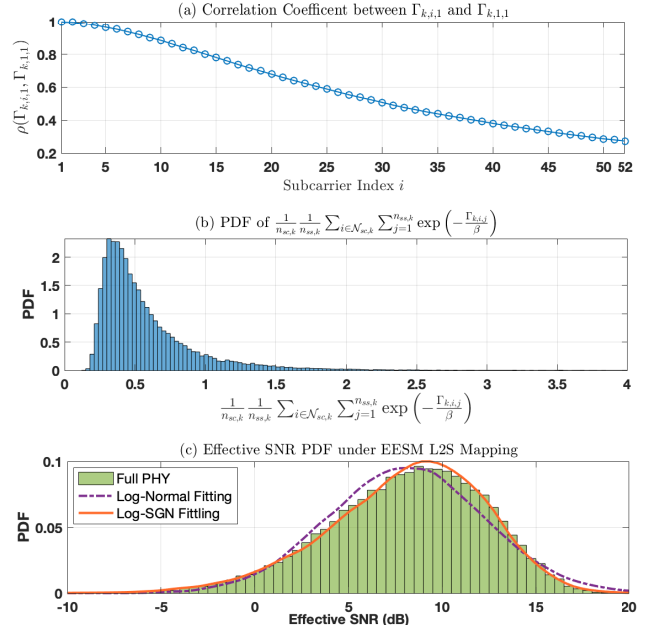


Fig. 9: Simulations under OFDMA with 52 subcarriers, SISO, IEEE TGax channel model-D, MCS4, RX SNR = 11dB, no interferer.

⁵For most data applications, the scope of average PERs is typically in $[10^{-2}, 1]$ [6]. For ultra-reliable applications requiring lower average PERs (e.g., average PERs down to 10^{-3}), more packets are required to be run in each full PHY simulation and that takes longer full PHY simulation runtime.

In contrast to [33], [35] and [34], [32] shows empirically that $\Gamma_{eff,k}^{sinr}$ under EESM L2S mapping can be approximated by a log-normal random variable (i.e., $\ln(\Gamma_{eff,k}^{sinr})$ is normally distributed), and this approximation is superior to modeling $\frac{1}{n_{sc,k}} \frac{1}{n_{ss,k}} \sum_{i \in \mathcal{N}_{sc,k}} \sum_{j=1}^{n_{ss,k}} \exp\left(-\frac{\Gamma_{k,i,j}}{\beta}\right)$ as a Chi-square/normal variable. In Fig. 9 (c), the histogram of $\Gamma_{eff,k}^{sinr}$ is approximated by a log-normal PDF using the method of moments; the PDF shown departs from the observed results because the log-normal variable has two parameters that can control mean and variance, but lacks flexibility in controlling the shape (e.g., skewness and kurtosis) simultaneously. To achieve better control of shape, we model $\ln(\Gamma_{eff,k}^{sinr})$ as a Skew-Generalized Normal (SGN) random variable [36], i.e.,

$$X \triangleq \ln(\Gamma_{eff,k}^{sinr}) \sim \text{SGN}(\hat{\mu}, \hat{\sigma}, \hat{\lambda}_1, \hat{\lambda}_2), \quad (4)$$

with PDF

$$f_X(x; \hat{\mu}, \hat{\sigma}, \hat{\lambda}_1, \hat{\lambda}_2) = \frac{2}{\hat{\sigma}} \psi\left(\frac{x - \hat{\mu}}{\hat{\sigma}}\right) \Psi\left(\frac{\hat{\lambda}_1(x - \hat{\mu})}{\sqrt{\hat{\sigma}^2 + \hat{\lambda}_2(x - \hat{\mu})^2}}\right), \quad x \in \mathbf{R}, \quad (5)$$

where $\hat{\mu} \in \mathbf{R}$ is the location parameter, $\hat{\sigma} > 0$ is the scale parameter, $\hat{\lambda}_1 \in \mathbf{R}$ and $\hat{\lambda}_2 \geq 0$ are shape parameters, $\psi(x)$ is the standard normal PDF, and $\Psi(x)$ is the standard normal cumulative distribution function. $\Gamma_{eff,k}^{sinr}$ satisfying (4) is called the log-Skew-Generalized Normal (log-SGN) random variable following log-SGN distribution. Modeling $\Gamma_{eff,k}^{sinr}$ using the log-SGN distribution has two advantages. First, the log-SGN distribution can simultaneously control mean, variance, and shape (e.g., skewness and kurtosis), and is thus suitable for $\Gamma_{eff,k}^{sinr}$ from complicated fading channels (e.g., IEEE TGax channels). It includes log-normal distribution as a special case and is able to achieve better approximation (see Fig. 9 (c)). Further, SGN random variables are easy to generate, as will be shown in Algorithm 2.

We next comment on the distribution of effective SINR under RBIR L2S mapping. As plotted in Fig. 10 (a), MCS 4 (16-QAM modulation), the effective SINR under RBIR L2S mapping saturates at 20dB. This can be understood by referring to Fig. 10 (b): where for MCS 4, RBIR saturates at 4 for $x \geq 20$ dB; this causes the effective SINR to saturate at 20dB after the inverse mapping Φ^{-1} in (2). The PDF of the effective SINR under RBIR L2S mapping hence deviates from log-SGN distribution (as shown in Fig. 10 (a)).

To summarize: under EESM L2S mapping, the log-SGN approximation for $\Gamma_{eff,k}^{sinr}$ is more accurate than the log-normal approximation [32]. Under RBIR L2S mapping, the distribution of $\Gamma_{eff,k}^{sinr}$ is bounded at high SINR, which makes the log-SGN approximation less accurate in this range. In the following, we focus on implementing the log-SGN approximation under EESM L2S mapping and validating its performance.

B. Log-SGN Parameter Estimation and Random Variable Generation

In this part, we first discuss how to generate the log-SGN parameters $\hat{\mu}, \hat{\sigma}, \hat{\lambda}_1, \hat{\lambda}_2$ for the considered PHY layer setup.

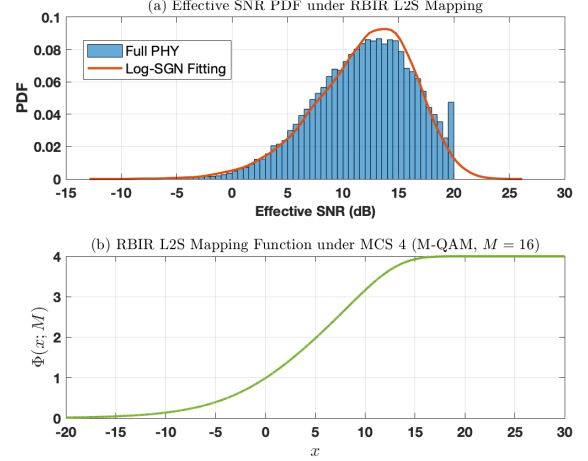


Fig. 10: (a) $\Gamma_{eff,k}^{sinr}$ under RBIR L2S mapping, OFDMA allocation with 52 subcarriers, SISO, IEEE TGax channel model-D, MCS4, RX SNR = 15dB; (b) RBIR L2S mapping function under MCS 4: $\Phi(x; 16)$.

The idea is to first obtain an effective SINR histogram using the optimized EESM parameter β and multiple post-MIMO processing SINR matrices generated by the full PHY simulation, and then statistically fit the effective SINR histogram with the log-SGN distribution defined in (4). The steps for obtaining the log-SGN parameters are shown in Algorithm 1, and the log-SGN parameters are stored in the network simulator for network simulations. Since the effective SINRs for log-SGN fitting are based on full PHY simulation results, the four log-SGN parameters summarize the key information in the PHY layer, including frequency selectivity of the channel, antenna correlation, OFDM/OFDMA MIMO/MU-MIMO setup, beamforming gain, MCS, channel coding type, number of interference sources, RX SNR and RX INRs. The log-SGN parameters are insensitive to the change of the packet length since the effective SINR distribution is insensitive to the change of the packet length (the impact of the packet length is mostly reflected in the generation of instantaneous PER, not effective SINR).

When running each packet in a network simulator, a realization of SGN distributed X can be generated using Algorithm 2 with stored log-SGN parameters $\hat{\mu}, \hat{\sigma}, \hat{\lambda}_1, \hat{\lambda}_2$ as inputs. The realization of X is then used to calculate $\Gamma_{eff,k}^{sinr} = \exp(X)$. Then, the effective SINR $\Gamma_{eff,k}^{sinr}$ is mapped into an instantaneous PER using the instantaneous PER-SNR lookup table under the AWGN-SISO channel at the specified packet length, channel coding and MCS. These steps are summarized in Fig. 2, and is called the *EESM-log-SGN PHY layer abstraction*. Compared to traditional PHY layer abstraction in Fig. 1, a benefit of the proposed method is that the implementation of increasingly complex channels (e.g., IEEE TGax channel [19], 3GPP TR 38.901 5G channel [20], 3GPP TR 37.885 V2X channel [21], etc.) can be avoided. In EESM-log-SGN PHY layer abstraction, reducing a complicated channel generation and PHY layer processing into drawing a log-SGN variable for calculating effective SINR is the fundamental innovation, that

Algorithm 1 Obtain Parameters of SGN Distributed X

Input: Channel type (e.g., IEEE TGax channel models A ~ F), MIMO/MU-MIMO dimension ($n_t, n_r, k, n_{ss,k}$), number of subcarriers to the user of interest (user k), index of the user sharing these subcarriers via MU-MIMO, MCS, channel coding type (e.g., LDPC or BCC), number of interferers, RX SNR and RX INRs.

Output: $\hat{\mu}, \hat{\sigma}, \hat{\lambda}_1, \hat{\lambda}_2$.

- 1: Obtain n realizations of post-MIMO processing SINR matrices Γ_k and the EESM tuning parameter β from the first two blocks in Fig. 3.
- 2: Generate n realizations of effective SINR $\Gamma_{eff,k}^{sinr}$ using (3), and the resulting n realizations of $X = \ln(\Gamma_{eff,k}^{sinr})$ are $x_i, i = 1, 2, \dots, n$.
- 3: Use MATLAB *fmincon* function to numerically maximize the log-likelihood function of the SGN random variable X [36]:

$$-\frac{n}{2} \ln \frac{\pi \hat{\sigma}^2}{2} - \frac{\sum_{i=1}^n (x_i - \hat{\mu})^2}{2\hat{\sigma}^2} + \sum_{i=1}^n \ln \Psi \left(\frac{\hat{\lambda}_1 (x_i - \hat{\mu})}{\sqrt{\hat{\sigma}^2 + \hat{\lambda}_2 (x_i - \hat{\mu})^2}} \right)$$

under the constraints $\hat{\sigma} > 0$ and $\hat{\lambda}_2 \geq 0$, and output the optimized solution.

makes the runtime insensitive to the dimension of the PHY layer parameters (e.g., $n_t, n_{r,k}$). Changing these PHY layer parameters only changes the values of log-SGN parameters $\hat{\mu}, \hat{\sigma}, \hat{\lambda}_1, \hat{\lambda}_2$, which hardly impacts the runtime in Algorithm 2.

Algorithm 2 Generate A Realization of SGN Distributed X [36], [37]

Input: $\hat{\mu}, \hat{\sigma}, \hat{\lambda}_1, \hat{\lambda}_2$ obtained from Algorithm 1.

Output: A realization of X .

- 1: Generate a Gaussian distributed random variable $\Xi = \sqrt{\hat{\lambda}_2} \Upsilon + \hat{\lambda}_1$, where $\Upsilon \sim \mathcal{N}(0, 1)$.
 - 2: Based on the realization of Ξ , generate two i.i.d. Gaussian random variables $U_1, U_2 \sim \mathcal{N}\left(\sqrt{\frac{1+\Xi^2}{2}} \hat{\mu}, \hat{\sigma}^2\right)$.
 - 3: Based on the realization of U_1 and U_2 , let $U = \max(U_1, U_2)$ and $V = \min(U_1, U_2)$. Then, $X = \frac{1+\Xi}{\sqrt{2(1+\Xi^2)}} U + \frac{1-\Xi}{\sqrt{2(1+\Xi^2)}} V$.
-

C. Validation and Runtime Performance of EESM-log-SGN PHY Layer Abstraction

In this section, we validate PHY layer abstraction methods and runtime performance under different OFDM/OFDMA MIMO/MU-MIMO scenarios. The implementations of the traditional EESM and RBIR PHY layer abstraction methods follow the flow in Fig. 7; the implementations of EESM-log-SGN methods follow the flow in Fig. 3; the implementation flows in Fig. 3 and Fig. 7 are suitable for all OFDM/OFDMA MIMO/MU-MIMO scenarios. The general simulation setup for all OFDM/OFDMA MIMO/MU-MIMO scenarios is shown in Table II. For estimating average PER in this section, the channel realizations for different packets are assumed to be i.i.d. [25], [26]; 40000 packets are sent in each simulation for achieving average PER down to 10^{-2} with high accuracy.

We first compare the effective SNR/SINR distribution obtained from both traditional EESM PHY layer abstraction in Fig. 1 and EESM-log-SGN PHY layer abstraction in Fig. 2. From Fig. 11, we can see that the effective SNR/SINR distribution of EESM-log-SGN PHY layer abstraction fits well

with the effective SNR/SINR distribution of traditional EESM PHY layer abstraction. In addition, the change in RX SNR (e.g., from 14dB to 10dB in Fig. 11) does not simply result in a shift of the effective SNR/SINR distribution, but changes also in skewness and kurtosis that necessitate the use of a new log-SGN distribution. The shapes of the effective SNR and effective SINR distributions are also different due to the frequency-selective interference.

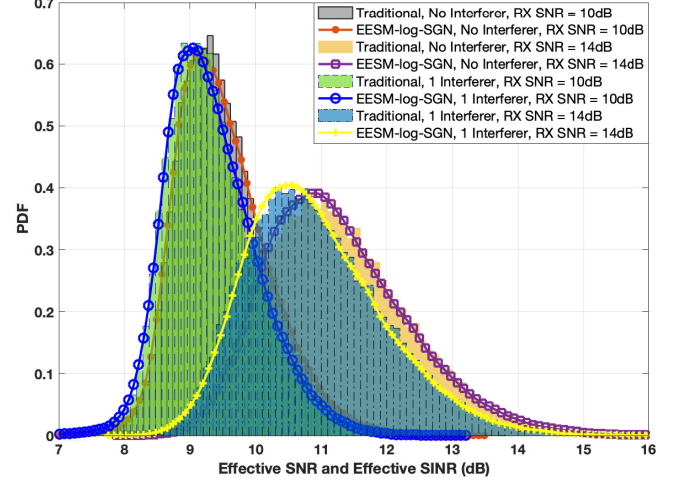


Fig. 11: Validation of EESM-log-SGN PHY layer abstraction under OFDMA allocation with 52 subcarriers, 8×2 MIMO with 2 streams, IEEE TGax channel model-D, MCS4. For the interference case, the RX INR is 20dB lower than the RX SNR.

We next compare the runtime of the EESM-log-SGN PHY layer abstraction with the traditional EESM PHY layer abstraction in Table IV. From Table IV, we can see that the runtime of the EESM-log-SGN PHY layer abstraction is much smaller than the runtime of the traditional EESM PHY layer abstraction in Fig. 1. The runtimes of the EESM-log-SGN based model do not scale with the MIMO dimensions, the MU dimensions, and the number of interferers, as all channel generation and PHY layer processing are mapped into the generation of X in Algorithm 2, whose runtime does not change with the change of the four log-SGN parameters.

IV. STORAGE-COMPLEXITY ASPECTS

The EESM-log-SGN PHY layer abstraction requires storing log-SGN parameters for each PHY layer configurations specified in Algorithm 1, including channel models, the number of subcarriers, MCS, MIMO/MU-MIMO setups and channel coding type specified by the standards. For example, the IEEE 802.11ax system configuration includes 6 different channels (IEEE TGax channel model-A to model-F), the number of subcarriers in set $\{26, 52, 106, 242, 484, 996\}$, up to 12 MCSs (MCS0 - MCS11), the number of antennas for each device in the range $1 \sim 8$ [2], and either BCC or LDPC channel coding. While the number of combinations of these setups can be large, this is not the primary contributor to storage complexity. Instead, two major storage complexity challenges arise from the fact that log-SGN parameters also depend on the interference scenario (the number of interferers

TABLE IV: Average runtime comparison between the traditional EESM PHY layer abstraction and the proposed EESM-log-SGN PHY layer abstraction for running a 40000-packet simulation at a specific RX SNR in MATLAB [7].

PHY Layer Setup	EESM	EESM-log-SGN
$\{242, 1 \times 1 : 1\}$, 1-user OFDM SISO, no interferer	16 min	0.6 sec
$\{242, 4 \times 2 : 2\}$, 1-user OFDM MIMO, no interferer	40 min	0.6 sec
$\{242, 8 \times 2 : 2\}$, 1-user OFDM MIMO, no interferer	72 min	0.6 sec
$\{242, 8 \times 2 : 2\}$, 1-user OFDM MIMO, 1 interferer	155 min	0.6 sec
$\{242, 8 \times 2 : 2\}$, 1-user OFDM MIMO, 2 interferers	230 min	0.6 sec
$\{106, 1 \times 1 : 1\} // \{52, 1 \times 1 : 1\} // \{52, 1 \times 1 : 1\}$, 3-user OFDMA SISO, no interferer	28 min	0.6 sec
$\{106, 8 \times 2 : 2\} // \{52, 8 \times 2 : 2\} // \{52, 8 \times 2 : 2\}$, 3-user OFDMA MIMO, no interferer	152 min	0.6 sec
$\{242, 8 \times 2 : 2\} // \{2, 2 : 2\} // \{2, 2 : 2\}$, 3-user OFDM MU-MIMO, no interferer	160 min	0.6 sec
$\{106, 8 \times 2 : 2\} // \{106, 8 \times 2 : 2\} // \{2, 2 : 2\}$, 3-user OFDMA MU-MIMO, no interferer	156 min	0.6 sec
$\{106, 8 \times 2 : 2\} // \{106, 8 \times 2 : 2\} // \{2, 2 : 2\}$, 3-user OFDMA MU-MIMO, 1 interferer	316 min	0.6 sec

and corresponding RX INRs) and RX SNR input, which involve numerous cases and can vary over wide ranges. Storing log-SGN parameters for each interference scenario and RX SNR input leads to high storage complexity. In this section, we address these two challenges by proposing low storage-complexity solutions.

A. Handling Interference Scenarios

In Section II-B, we defined the random variable effective SNR $\Gamma_{eff,k}^{snr}$ under an interference-free scenario. We now introduce the random variable effective INR $\Gamma_{eff,k}^{inr,v}$ as the effective SNR at receiver k with the signal from an interferer v only (no other transmitters). In Section IV-B, we will show that an effective SNR/INR distribution can be stored with low complexity under an arbitrary RX SNR/INR based on the EESM-log-SGN PHY layer abstraction. Therefore, for numerous interference scenarios, we consider the following motivating question: can we estimate an effective SINR distribution given an effective SNR and effective INR distribution?

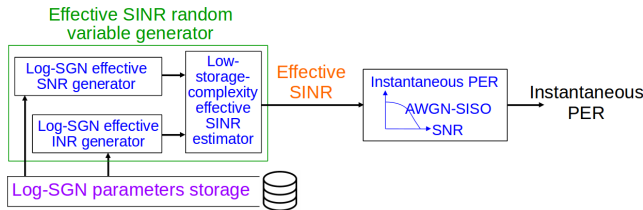


Fig. 12: Flow chart of EESM-log-SGN-LSC PHY layer abstraction.

We propose a Low-Storage-Complexity (LSC) version of EESM-log-SGN (EESM-log-SGN-LSC) PHY layer abstraction, specified in Fig. 12. The key component of the EESM-log-SGN-LSC PHY layer abstraction is an effective SINR estimator $\hat{\Gamma}_{eff,k}^{sinr}$. This is inspired by the definition: $SINR =$

$S/(N+I) = SNR/(1+INR)$. For any PHY layer setup (channel type, number of subcarriers, MCS, MIMO/MU-MIMO setup and channel coding type) with a given number of interferers, the effective SINR estimator is given by

$$\hat{\Gamma}_{eff,k}^{sinr} = \frac{\Gamma_{eff,k}^{snr}}{1 + \theta \sum_{v \in \mathcal{V}} \Gamma_{eff,k}^{inr,v}}, \quad (6)$$

where $\mathcal{V}$ is the set of interferers, $\Gamma_{eff,k}^{snr}$ is the EESM-log-SGN generated random variable under the same PHY layer setup without interference, $\Gamma_{eff,k}^{inr,v}$ is the EESM-log-SGN generated random variable under the same PHY layer setup with interferer v as the only transmitter, and the scalar θ is called the interference tuning parameter that is optimized to minimize the MSE between the distribution of $\hat{\Gamma}_{eff,k}^{sinr}$ and the distribution of $\Gamma_{eff,k}^{sinr}$ under the above PHY layer setup with a specified RX SNR and RX INRs. The interference tuning parameter θ comes from our innovation to make the effective SINR estimator accurate. For the effective SINR estimation to be an LSC solution, the optimal θ under a PHY layer setup with a specified RX SNR and RX INRs should provide accurate approximations of SINR distribution under the same PHY layer setup with other RX SNR and RX INRs. That is, the EESM-log-SGN-LSC method with a single θ should be insensitive to the change of RX SNR and RX INRs (the change of the geometry of the desired transmitter and interferers). Fortunately, our simulation results validate that a single θ can provide accurate effective SINR distribution estimation for a wide range of RX SNR/INRs, with three examples shown in Fig. 13 (a)-(c). For different numbers of interferers, we find different θ should be used to preserve SINR modeling accuracy, as shown in Fig. 13 (b)-(c). For any PHY setup, we recommend storing different values of θ for a limited number of interferers (e.g., $1 \sim 4$) using EESM-log-SGN-LSC and leave cases of a larger number of interferers for future work.

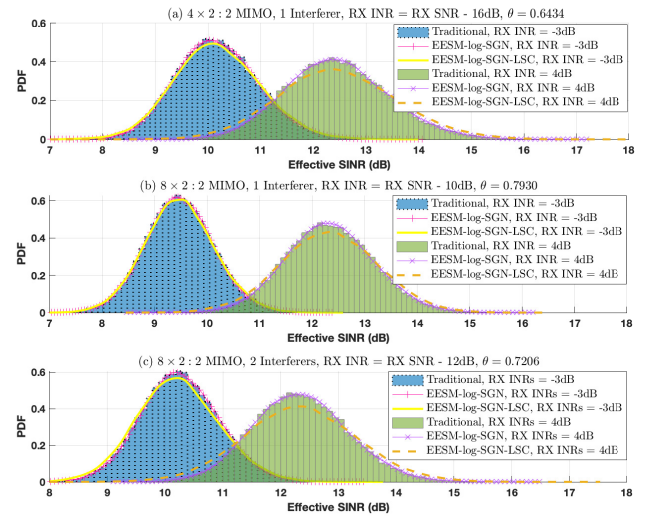


Fig. 13: Validation of EESM-log-SGN-LSC PHY layer abstraction under OFDM, 20MHz IEEE TGax channel model-D, MCS4, where θ in each subfigure is obtained at 4dB RX INR.

We compare the runtimes of the EESM-log-SGN and the EESM-log-SGN-LSC PHY layer abstractions in Table V.

While the runtime of EESM-log-SGN PHY layer abstraction is insensitive to the change of PHY layer setup, the positions of interferers, and the number of interferers, the runtime of EESM-log-SGN-LSC linearly scales with the number of interferers, because (6) requires computations that scale with the number of interferers. However, the runtimes of both proposed schemes are in the order of seconds and insensitive to the increase of MIMO and MU dimensions.

TABLE V: Average runtime comparison between the EESM-log-SGN PHY layer abstraction and the EESM-log-SGN-LSC PHY layer abstraction for running a 40000-packet simulation at a specific RX SNR in MATLAB [7].

PHY Layer Setup	EESM-log-SGN	EESM-log-SGN-LSC
{242, 1 × 1 : 1}, 1-user OFDM SISO, no interferer	0.6 sec	0.6 sec
{242, 4 × 2 : 2}, 1-user OFDM MIMO, no interferer	0.6 sec	0.6 sec
{242, 4 × 2 : 2}, 1-user OFDM MIMO, 1 interferer	0.6 sec	1.2 sec
{242, 8 × 2 : 2}, 1-user OFDM MIMO, no interferer	0.6 sec	0.6 sec
{242, 8 × 2 : 2}, 1-user OFDM MIMO, 1 interferer	0.6 sec	1.2 sec
{242, 8 × 2 : 2}, 1-user OFDM MIMO, 2 interferers	0.6 sec	1.8 sec
{106, 8 × 2 : 2} // {106, 8 × {2, 2} : {2, 2}}, 3-user OFDMA MU-MIMO, 1 interferer	0.6 sec	1.2 sec

B. Handling Wide Range of RX SNRs and INRs

Section IV-A suggests estimating an effective SINR using log-SGN parameters of an effective SNR and effective INRs. We now focus on how to generate an effective SNR distribution under an arbitrary RX SNR with low complexity, and the same principle can also be applied to generate an effective INR distribution.

For generating effective SNR under an arbitrary RX SNR, a simple approach is finding the closest stored RX SNR, and then lookup its log-SGN parameters. This requires storing a large number of log-SGN parameters under different RX SNRs for achieving high accuracy. From Section 5 in [6] (Figures 5.8, 5.13 - 5.15) and our MATLAB simulations in Section III-C, we can see that varying transmit/receive SNR over 25dB range suffices for obtaining all reasonable average PERs down to 10^{-2} for IEEE 802.11 system. The simulation in our conference paper [1] shows that 0.25dB changes to the RX SNR result in a discernible but not too significant change of effective SNR distribution. Thus, [1] suggests storing $25\text{dB}/0.25\text{dB} = 100$ sets of log-SGN parameters for every 0.25dB RX SNR increment for each PHY layer setup. However, storing 100 sets of log-SGN parameters for each PHY layer setup requiring moderately large storage complexity and increased burden of running the additional underlying full PHY simulations.

To avoid having to generate and store parameter sets at small RX SNR increments, we explore whether we can estimate effective SNR for any RX SNR using a small number of stored effective SNR distributions. We recommend an EESM-log-SGN *mixture model*, similar to the widely adopted Gaussian mixture model [16]. Specifically, consider an RX SNR

$\gamma \in [\gamma_1, \gamma_2]$, where the log-SGN parameters under RX SNRs γ_1 and γ_2 are stored, and γ_1 and γ_2 are the closest stored RX SNR values. Then, the PDF of effective SNR $\Gamma_{eff,k}^{snr}$ at user k with RX SNR γ is estimated by

$$\hat{f}(\Gamma_{eff,k}^{snr}; \gamma) = (1 - \xi)f(\Gamma_{eff,k}^{snr}; \gamma_1) + \xi f(\Gamma_{eff,k}^{snr}; \gamma_2), \quad (7)$$

where $\xi = \frac{\gamma - \gamma_1}{\gamma_2 - \gamma_1}$, and $f(\Gamma_{eff,k}^{snr}; \gamma_i)$ is the known log-SGN distribution under RX SNR $\gamma_i, i \in \{1, 2\}$. Typically, the RX SNR interval of interest⁶ is in $[\gamma_{min}, \gamma_{max}]$, where γ_{min} is the minimum stored RX SNR chosen to be small enough such that the average PER is close to 1, γ_{max} is the maximum stored RX SNR chosen to be large enough such that the average PER is below 10^{-2} . We implement the EESM-log-SGN mixture model in Algorithm 3. Using the EESM-log-SGN mixture model and a few values of stored effective SNR distributions, we are able to estimate instantaneous effective SNRs for any RX SNR in $[\gamma_{min}, \gamma_{max}]$. Compared with the method in [1] that stores 100 sets of log-SGN parameters and uses the closest RX SNR for log-SGN parameters lookup, the EESM-log-SGN mixture model significantly reduces the link simulation burden and the storage complexity in implementing the proposed EESM-log-SGN method.

Algorithm 3 Generate Effective SNR Random Variable for Any RX SNR $\gamma \in [\gamma_{min}, \gamma_{max}]$

Input: The RX SNR $\gamma \in [\gamma_{min}, \gamma_{max}]$ at receiver k .

Output: A realization of $\Gamma_{eff,k}^{snr}$.

- 1: Obtain RX SNR γ 's minimum interval $\gamma \in [\gamma_1, \gamma_2]$, where the log-SGN parameters of RX SNRs γ_1 and γ_2 are stored.
- 2: Generate a realization of uniform random variable $u \sim \text{unif}(0, 1)$.
- 3: Calculate $\xi = \frac{\gamma - \gamma_1}{\gamma_2 - \gamma_1}$.
- 4: **if** $u < 1 - \xi$ **then**
- 5: Generate a realization of $\Gamma_{eff,k}^{snr}$ from the log-SGN PDF $f(\Gamma_{eff,k}^{snr}; \gamma_1)$;
- 6: **else**
- 7: Generate a realization of $\Gamma_{eff,k}^{snr}$ from the log-SGN PDF $f(\Gamma_{eff,k}^{snr}; \gamma_2)$.
- 8: **end if**

We validate the proposed EESM-log-SGN mixture model for estimating arbitrary effective SNR distributions in the interference-free case. In Fig. 14, the green dash-dotted curve shows the distribution $\hat{f}(\Gamma_{eff,k}^{snr}; 21\text{dB})$ estimated from the stored log-SGN distributions $f(\Gamma_{eff,k}^{snr}; 20\text{dB})$ and $f(\Gamma_{eff,k}^{snr}; 22\text{dB})$. Fig. 14 compares the distribution $\hat{f}(\Gamma_{eff,k}^{snr}; 21\text{dB})$ with the log-SGN distribution $f(\Gamma_{eff,k}^{snr}; 21\text{dB})$ and shows that the proposed EESM-log-SGN mixture model can produce an acceptable estimate of effective SNR distribution.

Since the EESM-log-SGN mixture model has been shown to be effective, we can combine the EESM-log-SGN-LSC PHY layer abstraction and EESM-log-SGN mixture model into an EESM-log-SGN-LSC mixture model where the effective SINR distribution can be estimated using (6) as well as effective SNR/INR distributions produced by the mixture model under arbitrary RX SNR and RX INRs.

⁶If the RX SNR $\gamma < \gamma_{min}$ or $\gamma > \gamma_{max}$, the network simulator is required to use an alternative approach to estimate the effective SNR distribution under such an RX SNR and we leave this for future work.

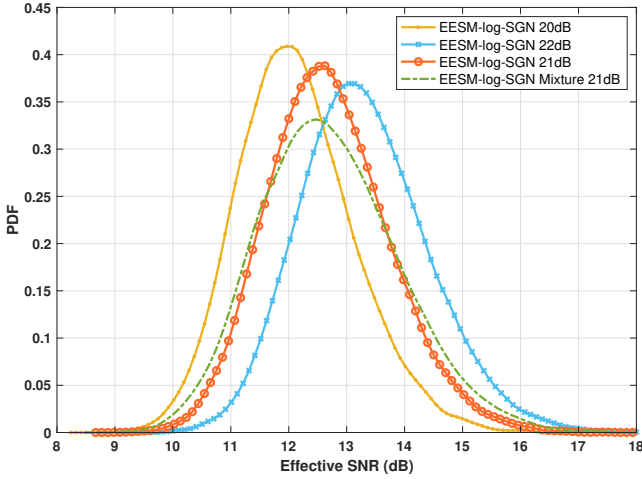


Fig. 14: Validation of EESM-log-SGN mixture model under OFDMA allocation with 106 subcarriers, $8 \times \{2, 2\}$ MU-MIMO with 2 streams/user, IEEE TGax channel model-D, MCS4.

We validated all proposed methods via results shown in Fig. 15 under the interference-free case and the 1 interferer case. Under the PHY layer setup in Fig. 15, log-SGN parameters for 14, 16, 18, 20, 22dB RX SNRs are stored for generating effective SNR distributions, and log-SGN parameters for $-16, -14, -12, -10, -8$ dB of RX INRs are stored for generating effective INR distributions. For the two curves under the mixture models, the effective SNR/SINR distributions for 15, 17, 19, 21dB of RX SNRs are estimated. The PER at each RX SNR is averaged over 40000 realizations. From Fig. 15, we can see that all PHY layer abstractions approximate the full PHY simulation results over the range. Furthermore, Fig. 15 shows that the mixture models can in general produce acceptable PER prediction when the log-SGN parameters are stored for every 2dB of RX SNR increment for the considered setup. Specifically, in the low RX SNR region (average PER is above 0.1), the PER predictions by the mixture models are accurate; in the high RX SNR region (average PER is below 0.1), the log-SGN parameters under different RX SNRs are recommended to be stored with higher granularity to improve PER prediction accuracy. For practical implementation, we recommend storing log-SGN parameters under 6 $\sim$ 10 different RX SNRs and 6 $\sim$ 10 different RX INRs in the network simulator for each PHY layer setup (channel type, MCS, OFDM/OFDMA MIMO/MU-MIMO setup, and channel coding type), and use the mixture models to estimate effective SNR/INR/SINR distributions under other RX SNRs/INRs.

V. CONCLUSION AND FUTURE WORK

In this work, traditional EESM and RBIR PHY layer abstraction methods are extended to general OFDM/OFDMA MIMO/MU-MIMO setups. These preserve accuracy at the cost of huge computation cost, due to the need for generating the channel instances and computing post-MIMO processing SINR matrices online. To manage this increase in simulation runtime, we next developed EESM-log-SGN PHY layer

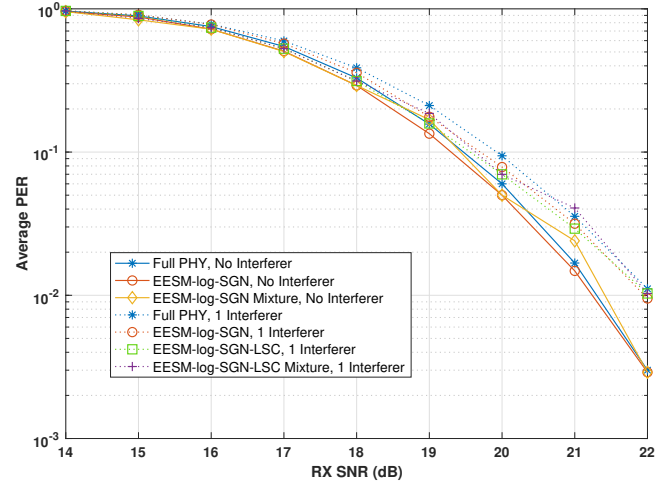


Fig. 15: Average PER versus RX SNR under OFDMA allocation with 106 subcarriers, $8 \times \{2, 2\}$ MU-MIMO with 2 streams/user, IEEE TGax channel model-D, MCS4. For the interference case, the RX INR is 30dB lower than RX SNR, and a single $\theta = 1.1535$ is used for all RX SNRs.

abstraction that directly characterizes the distribution of the output effective SINR at the receiver. The proposed EESM-log-SGN PHY layer abstraction achieves good accuracy in modeling the PHY layer performance, while the runtime is insensitive to system dimensionality change and is much reduced as compared to the traditional PHY layer abstraction methods. We finally discussed two storage challenges for the numerous interference scenarios and wide RX SNR/INR ranges, and proposed the corresponding low-complexity solutions.

Future priorities include considering the impact of temporal correlation in channels [38], extremely fast fading where channel varies during the transmission of each packet [39], inter-carrier interference [39], and time-frequency asynchronous interference [27]. In addition, inclusion of the impact of noisy channel estimation [40], imperfect time/frequency synchronization, imperfect phase tracking, and other PHY layer impairments (e.g., phase noise, power amplifier non-linearity, etc.) [8] are also needed. Extending the proposed EESM-log-SGN PHY layer abstraction to 5G system with Hybrid Automatic Repeat Request (HARQ) [15], heterogeneous user setup (different users are configured to different MCSs, channels, number of received streams, etc.), and millimeter-wave channel with hybrid precoding [13] are other considerations.

REFERENCES

- [1] S. Jin, S. Roy, W. Jiang, and T. R. Henderson, "Efficient abstractions for implementing tgn channel and ofdm-mimo links in ns-3," ser. WNS3 2020. New York, NY, USA: ACM, 2020, p. 33–40.
- [2] E. Khorov, A. Kiryanov, A. Lyakhov, and G. Bianchi, "A tutorial on ieee 802.11ax high efficiency wlns," *IEEE Communications Surveys Tutorials*, vol. 21, no. 1, pp. 197–216, 2019.
- [3] M. K. Müller, F. Ademaj, T. Dittrich, A. Fastenbauer, B. Ramos Elbal, A. Nabavi, L. Nagel, S. Schwarz, and M. Rupp, "Flexible multi-node simulation of cellular mobile communications: the vienna 5g system level simulator," *EURASIP journal on wireless communications and networking*, vol. 2018, no. 1, pp. 1–17, 2018.
- [4] Mathworks, "Physical layer abstraction for system-level simulation," 2021.

- [5] R. Patidar, S. Roy, T. R. Henderson, and A. Chandramohan, "Link-to-system mapping for ns-3 wi-fi ofdm error models," ser. WNS3 '17. New York, NY, USA: ACM, 2017, pp. 31–38.
- [6] E. Perahia and R. Stacey, *Next Generation Wireless LANs: 802.11n and 802.11ac*, 2nd ed. Cambridge University Press, 2013.
- [7] S. Jin, "EESM-log-SGN," *download information*: <https://github.com/sianjin/EESM-log-SGN>, 2021.
- [8] IEEE TGax Group, "Ieee p802.11 wireless lans: 11ax evaluation methodology," 2016.
- [9] M. Lacage and T. R. Henderson, "Yet another network simulator," ser. WNS2 '06. New York, NY, USA: ACM, 2006, pp. 12–22.
- [10] R. P. F. Hoefel and O. Bejarano, "On application of phy layer abstraction techniques for system level simulation and adaptive modulation in ieee 802.11 ac/ax systems," *Journal of Communication and Information Systems*, vol. 31, no. 1, 2016.
- [11] IEEE TGax Group, "Phy abstraction method comparison," in *IEEE 802.11-14/0647r2*, 2014.
- [12] T. L. Jensen, S. Kant, J. Wehinger, and B. H. Fleury, "Fast link adaptation for mimo ofdm," *IEEE Transactions on Vehicular Technology*, vol. 59, no. 8, pp. 3766–3778, Oct 2010.
- [13] T. Rappaport, R. Heath, R. Daniels, and J. Murdock, *Millimeter wave wireless communications*. Prentice Hall, 2015.
- [14] R. W. Heath Jr. and A. Lozano, *Foundations of MIMO Communication*. Cambridge University Press, 2018.
- [15] S. Lagen, K. Wanuga, H. Elkothy, S. Goyal, N. Patriciello, and L. Giupponi, "New Radio Physical Layer Abstraction for System-Level Simulations of 5G Networks," in *IEEE ICC*, Jun. 2020.
- [16] S. Kay, *Fundamentals of statistical signal processing: Practical algorithm development*. Pearson Prentice Hall, 2013, vol. III.
- [17] F. Kaltenberger, I. Latif, and R. Knopp, "On scalability, robustness and accuracy of physical layer abstraction for large-scale system-level evaluations of lte networks," in *Asilomar*, 2013, pp. 1644–1648.
- [18] I. Latif, F. Kaltenberger, and R. Knopp, "Link abstraction for multi-user mimo in lte using interference-aware receiver," in *IEEE WCNC*, 2012, pp. 842–846.
- [19] IEEE TGax Group, "Ieee 802.11ax channel model document," 2014.
- [20] T. Zugno, M. Polese, N. Patriciello, B. Bojović, S. Lagen, and M. Zorzi, "Implementation of a spatial channel model for ns-3," ser. WNS3 2020. New York, NY, USA: ACM, 2020, p. 49–56.
- [21] M. Drago, T. Zugno, M. Polese, M. Giordani, and M. Zorzi, "Millicar: An ns-3 module for mmwave nr v2x networks," ser. WNS3 2020. New York, NY, USA: ACM, 2020, p. 9–16.
- [22] Mathworks, "802.11ax downlink ofdma and multi-user mimo throughput simulation," 2021.
- [23] A. F. Molisch, *Wireless communications*, 2nd ed. Wiley : IEEE, 2011.
- [24] S. Ahmadi, "Chapter 3 - New Radio Access Physical Layer Aspects (Part 1)," in *5G NR*, S. Ahmadi, Ed. Academic Press, 2019, pp. 285–409.
- [25] Mathworks, "802.11ax packet error rate simulation for single-user format," 2021.
- [26] —, "802.11ax packet error rate simulation for uplink trigger-based format," 2021.
- [27] IEEE TGax Group, "Phy abstraction with time varying interference," in *IEEE 802.11-14/1174r0*, 2014.
- [28] Lei Wan, Shiauhe Tsai, and M. Almgren, "A fading-insensitive performance metric for a unified link quality model," in *IEEE WCNC*, vol. 4, 2006, pp. 2110–2114.
- [29] Q. H. Spencer, A. L. Swindlehurst, and M. Haardt, "Zero-forcing methods for downlink spatial multiplexing in multiuser mimo channels," *IEEE Transactions on Signal Processing*, vol. 52, no. 2, pp. 461–471, 2004.
- [30] IEEE TGax Group, "Phy abstraction for and ieee 11ax phy system simulation and integrated system level simulation," in *IEEE 802.11-14/0585r5*, 2014.
- [31] J.-M. Bardet, P. Doukhan, G. Lang, and N. Ragache, "Dependent lindeberg central limit theorem and some applications," *ESAIM: Probability and Statistics*, vol. 12, pp. 154–172, 2008.
- [32] S. N. Donthi and N. B. Mehta, "An accurate model for eesm and its application to analysis of cqi feedback schemes and scheduling in lte," *IEEE Transactions on Wireless Communications*, vol. 10, no. 10, pp. 3436–3448, October 2011.
- [33] A. Oborina, V. Koivunen, and T. Henttonen, "Effective sinr distribution in mimo ofdm systems," in *Asilomar 2010*, Pacific Grove, CA, USA, Nov 2010, pp. 511–515.
- [34] H. Song, R. Kwan, and J. Zhang, "Approximations of eesm effective snr distribution," *IEEE Transactions on Communications*, vol. 59, no. 2, pp. 603–612, February 2011.
- [35] J. Francis and N. B. Mehta, "Eesm-based link adaptation in point-to-point and multi-cell ofdm systems: Modeling and analysis," *IEEE Transactions on Wireless Communications*, vol. 13, no. 1, pp. 407–417, January 2014.
- [36] R. B. Arellano-Valle, H. W. Gómez, and F. A. Quintana, "A new class of skew-normal distributions," *Communications in statistics-Theory and Methods*, vol. 33, no. 7, pp. 1465–1480, 2004.
- [37] D. Ghorbanzadeh, P. Durand, and L. Jaupi, "Generating the skew normal random variable," in *Proceedings of WCE*, London, UK, 2017.
- [38] S. Jin, S. Roy, and T. R. Henderson, "EESM-log-AR: An efficient error model for ofdm mimo systems over time-varying channels," in *WNS3 2021*. Accepted, 2021.
- [39] W. Anwar, S. Dev, A. Kumar, N. Franchi, and G. Fettweis, "Phy abstraction techniques for v2x enabling technologies: Modeling and analysis," *IEEE Transactions on Vehicular Technology*, vol. 70, no. 2, pp. 1501–1517, 2021.
- [40] E. Eraslan, B. Daneshrad, and C. Lou, "Performance indicator for mimo mmse receivers in the presence of channel estimation error," *IEEE Wireless Communications Letters*, vol. 2, no. 2, pp. 211–214, 2013.



Sian Jin received the B.E. degree in electrical engineering from University of Electronic Science and Technology of China, China, in 2016. He is currently pursuing the Ph.D. degree with the Department of Electrical & Computer Engineering, University of Washington, Seattle. His research interests include radar signal processing, WiFi PHY layer signal processing, array processing, salable network simulator design, channel modeling, and wireless systems building using MATLAB Toolboxes/ns-3.



Sumit Roy received the B. Tech. degree from the Indian Institute of Technology (Kanpur) in 1983, and the M. S. and Ph. D. degrees from the University of California (Santa Barbara), all in Electrical Engineering in 1985 and 1988 respectively, as well as an M. A. in Statistics and Applied Probability in 1988. He is a professor in University of Washington Electrical & Comp. Engineering, appointed to a term Distinguished Professorship for Integrated Systems. He has served as IEEE ComSoc Distinguished Lecturer and as Associate Editor for the major ComSoc

journals. He currently serves on the Executive Committee of the National Spectrum Consortium dedicated to efficient spectrum sharing between Federal licensed and civilian sectors. He was elevated to IEEE Fellow by Communications Society in 2007 for "contributions to multi-user communications theory and cross-layer design of wireless networking standards".



Thomas R. Henderson is the lead maintainer of the ns-3 network simulator project, and Director of the University of Washington ns-3 Consortium. He participates in research projects related to ns-3 through an Affiliate Professor appointment in the University of Washington Department of Electrical & Computing Engineering, and also works as a freelance software developer. He was previously a Technical Fellow with Boeing Research & Technology, a Member of Technical Staff at COMSAT Laboratories, and was a founding engineer of GlobeArc, a Bay Area technology startup in the late 1990s. He received the B.S. and M.S. degrees from Stanford University, and a Ph.D. from the University of California, Berkeley, all in Electrical Engineering.