

Machine Learning-based Channel Classification and Its Application to IEEE 802.11ad Communications

Ernest Kurniawan, Lin Zhiwei, and Sumei Sun

Institute for Infocomm Research, Agency for Science Technology and Research (A*STAR)

1 Fusionopolis Way, #21-01 Connexis (South Tower), Singapore 138632

Email: {ekurniawan, zwlin, sunsm}@i2r.a-star.edu.sg

Abstract—We study the application of machine learning to channel classification for identifying whether a channel belongs to the Line of Sight (LOS) or Non-Line of Sight (NLOS) classes. The machine learning approach is able to work on multiple features, resulting in a much more accurate pattern identification and classification performance. We show that even in the absence of channel estimation, it is possible to classify the channel using the received preamble sequence with machine learning. This allows quicker classification and it is robust to channel estimation error, which is favorable in the low Signal to Noise Ratio (SNR) regime. The scheme is evaluated for IEEE 802.11ad systems, but the concept is also applicable to other wireless systems in general.

I. INTRODUCTION

The need for higher bandwidth to support very high throughput wireless transmissions has motivated the use of the 60 GHz channels. With 7 GHz (59-66 GHz) of unlicensed band, it becomes a very attractive channel to deliver multi Gigabit data rate for applications such as high-definition (HD) video streaming, in-car and in-flight entertainment (IFE) systems, and femtocell backhaul links. The IEEE 802 standardization body has worked out the specifications for the 802.11ad protocol [1], which further accelerates the deployment of 60 GHz communications.

One of the biggest challenges in wireless communications at 60 GHz is the propagation channel. At such a high frequency, the propagation loss is more severe than 2 GHz or 5 GHz channels, and the microwave signal starts to show its quasi-optical nature [2] (having a weak diffraction component, and structured/clustered reflective components). The channel quality is therefore highly dependent on the presence of the direct path or Line of Sight (LOS) component. Fig. 1 shows a link-level simulation of 802.11ad systems according to [1] assuming isotropic antennas at the transmitter and receiver, which are separated 3 meters apart (medium distance scenario). It can be seen from the figure that the packet error rate (PER) performance gap can reach up to 3 dB between the LOS and Non-LOS (NLOS) channel.

To maintain the Quality of Service (QoS), it is necessary to adapt the transmission strategy including the Modulation and Coding Schemes (MCS) according to the channel condition. From the example shown in Fig. 1, to achieve a target PER of 10^{-2} at 13 dB Signal to Noise Ratio (SNR) level, MCS 9 with Low Density Parity Check (LDPC) code rate of 13/16 and $\pi/2$ -Quadrature Phase Shift Keying (QPSK) modulation can be used when the channel has the LOS component.

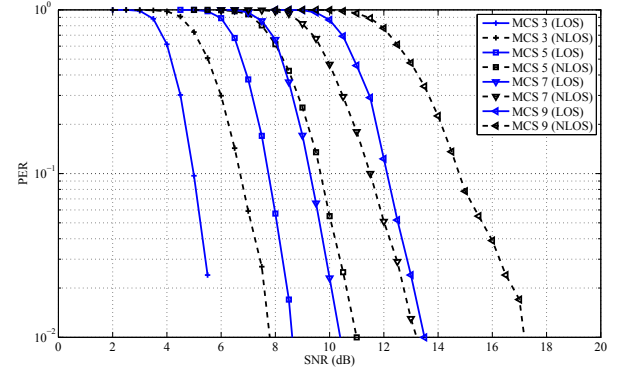


Fig. 1. Average Packet Error Rate

Otherwise, MCS 7 with LDPC code rate of 5/8 and $\pi/2$ -QPSK modulation should be used instead. This necessitates a method to identify the channel into LOS/NLOS classes, which needs to be performed in real time, especially in the IFE application whereby the dynamics of the in-cabin environment causes the LOS components to be frequently blocked by passing passengers or stewardess. Without an accurate channel identification, throughput loss will be incurred if a LOS channel is identified as a NLOS channel hence a low-rate MCS is used; or many re-transmissions will be caused when a NLOS channel is identified as a LOS channel hence a high-rate MCS is used to lead to a packet error. As a result, throughput loss is incurred.

Existing approach for LOS/NLOS classification is based on a computation of some metric (e.g. the Rician K-factor in [3], the Root Mean Square (RMS) delay spread, and mean excess delay in [4]), followed by a binary hypothesis testing. Machine learning has recently found its use in wireless communications, and it has been considered for applications such as jamming detection [5] and antenna selection [6]. This paper is motivated to use machine learning approach for channel classification. The ability of machine learning to work on multi-dimensional features and to identify patterns is shown to achieve better classification result than the single-metric classifier. Furthermore, the scheme enables us to perform the classification earlier by relying only on the received preamble sequence than waiting for the channel estimates using the received preambles.

The remaining part of the paper is organized as follows.

TABLE I
INTER-CLUSTER PARAMETER VALUES

	Index (i)	Reflector(s)	Block Prob.	ToA (ns)	Azimuth Angle (deg)		Elevation Angle (deg)		Reflect. Gain (dB)	
					Tx	Rx	Tx	Rx	Mean	Std. Dev
1st order	1	Wall	7%	U[10, 20]	U[0, 75]	U[-165, -90]	U[-4, -3.2]	U[3.2, 4]	-3	1.5
	2	Front Partition	9%	U[8, 23]	U[-60, 15] * U[-15, 0]	U[60, 180] * U[0, 15]	U[-3.6, -2.8]	U[2.8, 3.6]	-8.5	3.5
	3	Rear Partition	9%	U[8, 23]	U[60, 180] * U[0, 15]	U[-60, 15] * U[-15, 0]	U[-3.6, -2.8]	U[2.8, 3.6]	-8.5	3.5
	4	Ceiling	4%	U[2, 8]	0	0	U[15, 25]	U[30, 50]	-3	1.5
	5	Floor	4.5%	U[2, 8]	0	0	U[-55, -35]	U[-55, -35]	-8.5	3.5
2nd order	6	Wall – Wall	12.5%	U[24, 40]	U[-90, 90]	U[90, 270]	U[-5, -4.2]	U[4.2, 5]	-6	3
	7	Wall – Front Partition	12.5%	U[20, 36]	U[-50, 50]	U[130, 230]	U[-3.6, -2.8]	U[2.8, 3.6]	-11.5	5
	8	Wall – Rear Partition	12.5%	U[20, 36]	U[50, 140]	U[40, 130]	U[-3.6, -2.8]	U[2.8, 3.6]	-11.5	5
	9	Floor – Ceiling	5%	U[3, 9]	0	0	U[-55, -35]	U[-55, -35]	-11.5	5
	10	Floor – Wall	4.5%	U[3, 9]	0	0	U[-30, -15]	U[-22.5, -11.25]	-11.5	5

Section II describes the 60 GHz channel model used throughout the experiment. Section III explains the learning algorithm used, including the data preparation and training method. Performance evaluation is presented in Section IV. Finally, Section V concludes the paper.

II. CHANNEL MODEL

Considering the quasi-optical nature of 60 GHz propagation channel, we adopt the statistical model described in [2] whereby most of the transmitted signal power reaches the destination via a direct path (LOS component) and a few low-order reflected paths (NLOS component). The i^{th} reflected signal component departs from the transmitter at azimuth angle $\Phi_{tx}^{(i)}$ and elevation angle $\Theta_{tx}^{(i)}$, and arrives at the receiver with a propagation delay of $T^{(i)}$ at azimuth angle $\Phi_{rx}^{(i)}$ and elevation angle $\Theta_{rx}^{(i)}$. These parameters can be well approximated using image based ray-tracing method. Denoting by $A^{(i)}$ the amplitude gain of the i^{th} reflected signal component, the channel impulse response is expressed as [2]:

$$h(t, \varphi_{tx}, \theta_{tx}, \varphi_{rx}, \theta_{rx}) = \sum_{i=0}^{N_I} A^{(i)} C^{(i)}(t - T^{(i)}, \varphi_{tx} - \Phi_{tx}^{(i)}, \theta_{tx} - \Theta_{tx}^{(i)}, \varphi_{rx} - \Phi_{rx}^{(i)}, \theta_{rx} - \Theta_{rx}^{(i)}). \quad (1)$$

In the above, the function $C^{(i)}(\cdot)$ is used to describe the impulse response contributed from the i^{th} reflected component, which also consists of a cluster of rays closely spaced in time and angular domain. Denoting as $\tau^{(i,k)}$ the relative delay of the k^{th} ray in the i^{th} cluster, and $\varphi_{tx}^{(i,k)}, \theta_{tx}^{(i,k)}, \varphi_{rx}^{(i,k)}, \theta_{rx}^{(i,k)}$ as the relative angles at the transmitter and receiver for the azimuth and elevation direction of the k^{th} ray in the i^{th} cluster, respectively, the intra-cluster impulse response $C^{(i)}(\cdot)$ can be expressed as [2]:

$$C^{(i)}(t, \varphi_{tx}, \theta_{tx}, \varphi_{rx}, \theta_{rx}) = \sum_{k=1}^{N_K^{(i)}} \alpha^{(i,k)} \delta(t - \tau^{(i,k)}) \delta(\varphi_{tx} - \varphi_{tx}^{(i,k)}) \delta(\theta_{tx} - \theta_{tx}^{(i,k)}) \delta(\varphi_{rx} - \varphi_{rx}^{(i,k)}) \delta(\theta_{rx} - \theta_{rx}^{(i,k)}), \quad (2)$$

where $\delta(\cdot)$ is the Dirac delta function. The amplitude of the k^{th} ray in the i^{th} cluster $\alpha^{(i,k)}$ follows an exponential power

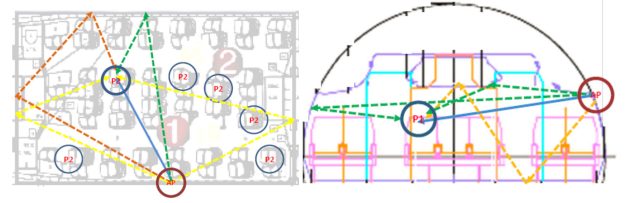


Fig. 2. Ray Tracing Simulation in a Mock-up Cabin of Airbus A350

decay with Rayleigh distributed amplitude, Normal distributed arrival/departure azimuth/elevation angles, and Poisson distributed Time of Arrival (ToA). We use the same parameters as the conference room environment reported in [2] for the intra cluster rays characteristics.

For the inter-cluster characteristics, we perform an image-based ray tracing simulation on Airbus A350 passenger aircraft layout and cross-section as shown in Fig. 2. This diagram is adapted from [7, Fig. 4] which gives the cabin mock-up cross section of Airbus A340. In total, we consider a maximum of 5 first order and 5 second order reflective components, and included the cabin wall, ceiling, floor, and front/rear area partition as the reflectors. The obtained parameters are summarized in Table I. We have used $U[a, b]$ to denote a uniform distribution from a to b , and $*$ to denote a convolution operation. The cells that are shaded red indicates the presence of correlation across parameters from different reflected components (column-wise correlation), while those that are shaded blue indicates the presence of correlation across parameters at the transmitter and the receiver (row-wise correlation).

As far as the channel gain is concerned, the following attenuated Friis transmission equation is used:

$$G^{(i)} \approx 20 \log_{10} A^{(i)} = \frac{g^{(i)} \lambda}{4\pi d^{(i)}}, \quad (3)$$

where λ is the wavelength (equal to 5×10^{-3} for the 60 GHz frequency), $d^{(i)}$ is the total distance traveled by i^{th} signal component, and $g^{(i)}$ is the attenuation factor, which is equal to $g^{(0)} = 1$ for the LOS component, and it is random according to a truncated log-normal distribution with mean and standard deviation as specified in Table I.

III. LEARNING ALGORITHM

We adopt a supervised learning approach, whereby a collection of labeled data set is used to train a model, which will then be used to classify a new set of input data. Since there are only two classes of interest (LOS and NLOS), this falls into a binary classification problem.

A. Data Generation and Collection

In order to obtain the data set, we randomly generate 1000 channel realizations in Matlab for each of the classes using the parameters obtained in Section II. Each of these channel realizations comprises of multiple taps with their corresponding delay and magnitude. The set of generated channel realizations forms a data set (*actual channel data set*) which will be used to perform classification under idealistic scenario, whereby the channel impulse response (CIR) is known perfectly.

In practical scenario, however, the receiver does not have access to the actual channel, and have to estimate them from the noisy and corrupted received signal. To generate a data set for the channel estimate, we use a link level simulation of 802.11ad system which we have developed in Matlab according to the specification in [1]. For simplicity, we choose a single carrier mode, and generated random data payload to create a physical (PHY) layer frame as illustrated in Fig. 3.



Fig. 3. 802.11ad Single Carrier PHY Frame Structure

This PHY frame packet is then transmitted over the channel which we have generated earlier. In addition to the multipath dispersion, the link level simulation also adds impairments such as In-phase/Quadrature (I/Q) imbalance, phase noise, Analog to Digital Conversion/Digital Analog to Conversion (ADC/DAC) clipping noise, Direct Current (DC) offset, carrier frequency offset, as well as additive Gaussian noise to the received signal. The system parameters used in the simulation are summarized in Table II.

TABLE II
802.11AD LINK LEVEL SIMULATION SYSTEM PARAMETERS

Parameters	Value
Centre Frequency	60.48GHz
Sampling frequency	3.52GHz
AD / DA bits	6 bits
Frequency offsets	40ppm
Tx / Rx IQ imbalance	Amplitude 1dB / Phase 10°
DC offset	I-Phase / Q-Phase 10%
Tx/Rx Phase noise	PSD(0) = -95dBc/Hz fp = 1MHz fz = 100MHz
Chip rate	1.76GHz
Packet duration	22.87 μs
MCS	7 (5/8 LDPC, π/2 QPSK)

At the receiver, the impairments are estimated and compensated using the Short Training Field (STF), and subsequently the CIR is estimated using the Channel Estimation Fields

(CEF). This channel estimate is then collected and labeled as LOS or NLOS accordingly, depending on the class of the multipath channel that has been used in the simulation. The obtained data set (*estimated channel data set*) will be used to perform classification under practical scenario, whereby the classification is performed on the estimated channel.

In addition to the data set for the actual and estimated channel, we also collected another data set from the received preamble sequence. In 802.11ad systems, the preamble comprises of a concatenation of different variations of complementary Golay sequences, therefore it has a certain pattern. The motivation of using the received preamble as the data set to perform classification is to study whether or not the different channel classes (LOS and NLOS) alter the preamble sequence pattern differently, and whether or not a machine learning algorithm is able to capture those differences to perform accurate classification.

The benefits of using the received preamble sequence includes quicker identification, as the classification does not need to wait for the channel estimation results which are obtained using the received channel estimation fields. The classifier is also independent of the channel estimation performance. The disadvantage of this method is the large feature size which incurs higher complexity in both the model training and the classification process. To collect the data set (*preamble data set*), we store both the real and imaginary part of the received preamble sequence, and label it as LOS or NLOS depending on the class of the channel used in the simulation.

B. Model Training

Using the labeled data set collected in the earlier step, we randomly select 60% for training and 40% for testing. To minimize the impact of this selection, we repeat this process 20 times, and take the average value of the classification performance.

For the learning algorithm, we use Random Forest [8] due to its superior accuracy compared to the other classifiers. In Random Forest, multiple random subsets of input features are chosen, and multiple decision trees (hence the name forest) are constructed, one for each set of the selected features. The final decision made by the Random Forest is a weighted average of the output of the decision trees in the forest, with its weights optimized according to the performance of each individual tree. We set the maximum number of trees generated to be 50, which is known to achieve a good trade-off between the complexity and the classification performance [5].

Using the actual channel data set and the estimated channel data set, the following features are considered.

1) *K Factor*: Based on the CIR equation given by (1) and (2), the K-Factor can be calculated using the following formula

$$K = \frac{|A^{(0)}\alpha^{(0,1)}|^2}{\sum_{i=1}^{N_I} \sum_{k=1}^{N_k^{(i)}} |A^{(i)}\alpha^{(i,k)}|^2}. \quad (4)$$

2) *RMS delay spread*: The RMS delay spread of the channel can be calculated using

$$\rho = \frac{\sum_{i=0}^{N_I} \sum_{k=1}^{N_k^{(i)}} ((T^{(i)} + \tau^{(i,k)}) - \mu)^2 |A^{(i)} \alpha^{(i,k)}|^2}{\sum_{i=0}^{N_I} \sum_{k=1}^{N_k^{(i)}} |A^{(i)} \alpha^{(i,k)}|^2}, \quad (5)$$

where μ is the mean excess delay defined in (6), and it is another feature considered for classification as described in the next point.

3) *Mean excess delay*: The formula to calculate the mean excess delay is given in the following

$$\mu = \frac{\sum_{i=0}^{N_I} \sum_{k=1}^{N_k^{(i)}} (T^{(i)} + \tau^{(i,k)}) |A^{(i)} \alpha^{(i,k)}|^2}{\sum_{i=0}^{N_I} \sum_{k=1}^{N_k^{(i)}} |A^{(i)} \alpha^{(i,k)}|^2}. \quad (6)$$

The above three features are scalar features, and they have been considered in [3] and [4] for binary hypothesis testing. Since there is only one feature, the Random Forest algorithm reduces to a classification tree in these cases. The other features considered from the actual channel data set and the estimated channel data set are as follows.

4) *Channel magnitude gain (All taps)*: Here, we use the magnitude of the CIR itself as the feature for classification. Considering that the shape of the magnitude of the CIR is different for LOS and NLOS scenario, it is expected that the machine learning algorithm is able to differentiate between the two patterns.

5) *Top five channel taps*: Considering only the five dominant taps in the CIR, we use both the magnitude and the relative delay of the five dominant taps as the classification feature. Similarly, the different pattern of the five dominant taps in the LOS and NLOS scenario suggests that it may be sufficient to look at only the five dominant taps to classify the channel.

6) *Dominant channel tap*: Following the same consideration as using the top five dominant taps as features, this considers an extreme case whereby only the dominant tap (both the magnitude and its relative delay) is considered as the classification feature.

Finally, the last feature which is only applicable to the preamble data set, is the received preamble.

7) *Received preamble sequence*: Here, we use both the real and imaginary components of the received preamble sequence. In terms of the feature size, this is the largest compared to the other features considered earlier. However, it is not clear if it will be suitable for classification as the channel impairments are still present and may disrupt the pattern necessary for correct identification.

As far as the model training is concerned, we export the data sets generated in Matlab into an Excel file, and use R software [9], a programming language and environment for statistical computing, to train the Random Forest and to use the trained model to classify the channels in the test data set. The classification results using different features under different scenarios are given in the following section.

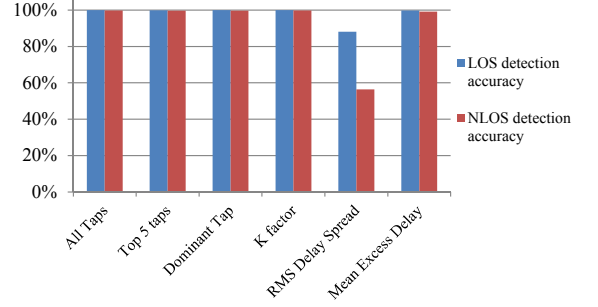


Fig. 4. Classification Performance on the Actual Channel Realizations

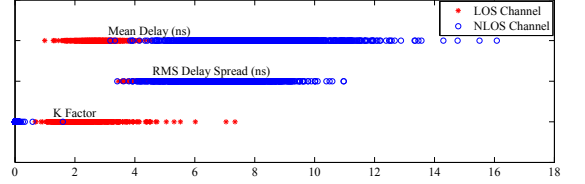


Fig. 5. Feature Distribution on the Actual Channel Realizations

IV. PERFORMANCE EVALUATION

Both the LOS as well as the NLOS classification performance are considered. We begin with the ideal scenario whereby the perfect CIR is assumed to be known, and the features are calculated using this actual CIR.

A. Ideal Scenario

Fig. 4 shows the classification performance for the first six features using the actual channel data set. It can be seen that the classification performance of all the features (except the RMS delay spread) achieves close to perfect 100% score.

To analyze the cause of the poor performance of the RMS delay spread feature, we plot the values of all the single-value features (namely the K-Factor, RMS delay spread, and mean excess delay) in Fig. 5. It is apparent that the RMS delay spread calculated for the LOS and NLOS channels overlap with one another, therefore it not a good feature to use for classifying the channel. In the meantime, both K-Factor and mean excess delay show clear separation between those values calculated from LOS channel and those from NLOS channels.

Despite the good classification performance, it should be noted that this result is obtained under the ideal scenario where perfect knowledge of the CIR is available. In practical scenario, this information is not available, and the receiver must perform channel estimation and construct the features based on that estimate.

B. Practical Scenario

Here, the classification is performed using the features calculated from the estimated channel data set. The additive Gaussian noise power is set to -10 dB in this case. Fig. 6 shows the classification performance for both the LOS and NLOS channels.

Compared to the classification performance under ideal scenario as shown in Fig. 4, it is apparent that the presence of

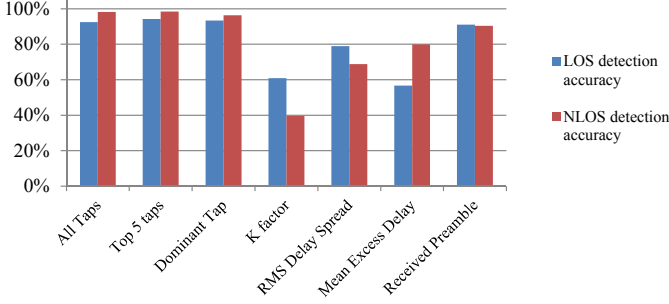


Fig. 6. Classification Performance on the Estimated Channel at -10dB SNR

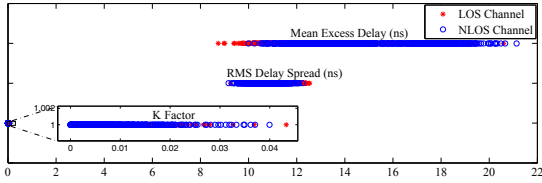


Fig. 7. Feature Distribution on the Estimated Channel at -10dB SNR

noise (and therefore the channel estimation error) causes significant degradation, especially to those using the single-value features. Meanwhile, the classification performance using the channel taps information (with either all taps, top 5 taps, or dominant tap), shows only a slight degradation, and still able to achieve more than 90% correct classification. This shows that even though the channel estimation error causes a deviation to the values of the CIR, its effect on the shape/pattern of the CIR is not very severe. As such, the machine learning approach which relies more on the shape/pattern of the CIR estimate, is still able to perform well.

On the contrary, all of the single-value features use a fixed formula to calculate based on the actual values of the CIR. Therefore, any deviation on the CIR estimate from the actual CIR will have a direct impact to those features. To illustrate this, we plot in Fig. 7 the values of the single-value features based on the channels in the estimated channel data set.

It is apparent that the separation between the feature values from the two classes is no longer possible, and there is a significant overlap between the LOS and NLOS group.

The RMS delay spread does not perform well in the ideal case, and it is the same in this scenario. For the K-Factor, all the values become concentrated into small numbers. Meanwhile, for the mean excess delay, the values from LOS group become larger and overlap with the NLOS group. To see the cause of this behavior, we plot one example CIR from the LOS class and the corresponding estimate in Fig. 8.

From the figure, it is apparent that the main reason why the K-Factor becomes very small is due to imperfect synchronization. This causes the LOS component in the CIR estimate to be slightly shifted away from the first tap. Consequently, the K-Factor calculated according to (4) will shrink, as it relies on the amplitude of the first tap in its numerator. As such, the K-Factor for both LOS and NLOS channels is no longer distinguishable. This problem occurs even in the high SNR

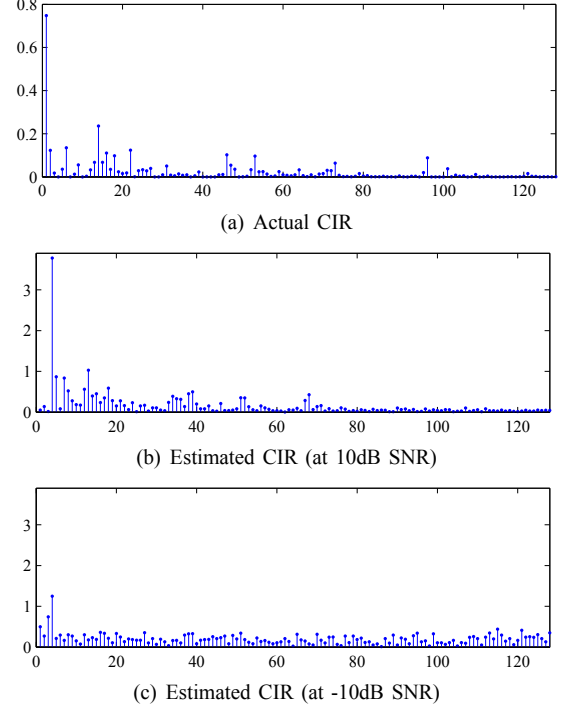


Fig. 8. Example CIR from LOS Class and the Estimates

case as shown in Fig. 8b.

As for the mean excess delay, the presence of estimation noise generally increases the magnitude of the channel taps which are small, and decreases those which are large. Therefore, its quantity for the LOS channel which is supposed to be small becomes larger and comparable to the NLOS channel.

An additional feature that we have considered is the received preamble sequence, which uses the preamble data set for classification. The performance of this feature is shown in the right most bar chart in Fig. 6, and it is comparable to the classification performance using the channel tap information (around 90% accuracy). This shows that even though the channel impairments and additive noise corrupt the received preamble sequence, its structure is preserved, allowing the machine learning algorithm to perform correct classification.

C. Impact of Additive Noise and Estimation Error

Having seen the classification performance for the different features under ideal and noisy scenario, we are now interested in evaluating how they perform at different noise levels. Fig. 9 shows the average classification performance of the different features at different SNR values.

The K-Factor and RMS delay spread are not able to achieve good performance in the SNR range considered. The only single-value feature that has consistent performance (improved classification accuracy with increasing SNR) is the mean excess delay. However, its performance is inferior compared to the features that use channel tap information and the received preamble sequence.

The estimated CIR is shown to be a good feature for

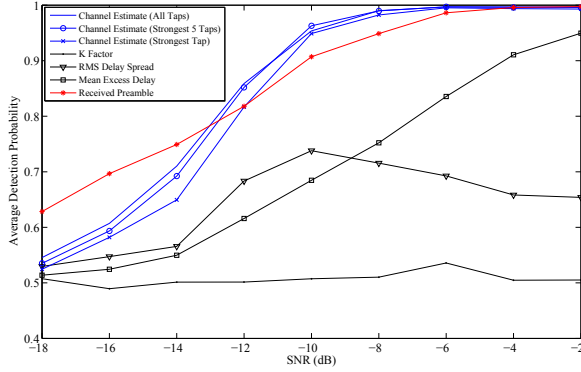


Fig. 9. Correct Classification Probability at Different SNR Values

classification, and all the three features (all taps, top 5 taps, and dominant tap) are able to achieve higher than 90% classification accuracy at SNR -10 dB and above. In general, the more taps considered for classification, the better the resulting performance.

Lastly, the received preamble sequence also exhibits good classification performance in the SNR range considered, with only slight degradation compared to those using channel tap information at SNR levels larger than -15 dB. The performance eventually converges to close to 100% accuracy at -4 dB and above.

At the low SNR regime below -15 dB, the received preamble sequence turns out to be a better feature for classification than the channel tap information. Although these two features rely on the shape/pattern for performing classification, at this low SNR regime, the distortion introduced by the channel estimation error severely impact the pattern in the channel tap information used for classification. On the other hand, with the receive preamble, the pattern is only affected by the channel impairments, and not by the estimation algorithm. Therefore the machine learning algorithm is still able to exploit it to perform better classification.

V. CONCLUSIONS

This paper studies the application of machine learning technique for channel classification. Using the IEEE 802.11ad system in the aircraft cabin environment, we simulated the propagation channel and considered different features for the classification. Compared to the single-valued features such as K-Factor, RMS delay spread, and mean excess delay, the channel tap information of the estimated channel impulse response is shown to be good for classification. Furthermore, the received preamble sequence also offers a good classification feature that is robust to the channel estimation error. The ability of the machine learning to adapt the model as the propagation condition changes using an on-line learning concept is a subject of future work.

ACKNOWLEDGMENT

The authors would like to thank Dr. Zhao Weijiang from the Institute of High Performance Computing (IHPC), Agency

for Science Technology and Research (A*STAR) Singapore for providing us with the ray tracing tools and results for the inter-cluster parameters of multipath delay profile in Airbus A350 cabin environment. The authors would also like to thank Dr. Tan Peng Hui, Dr. Chen Qian, and Dr. Zhang Peng from the Institute for Infocomm Research, Agency for Science Technology and Research (A*STAR) Singapore for the stimulating and useful discussion.

REFERENCES

- [1] IEEE Std. 802.11ad, "IEEE Standard for Local and Metropolitan Area Networks - part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications Amendment 4: Enhancements for Very High Throughput in the 60 GHz Band," 2012.
- [2] A. Maltsev, R. Maslennikov, A. Lomayev, A. Sevastyanov, and A. Khoryaev, "Statistical Channel Model for 60 GHz WLAN Systems in Conference Room Environment," *Radioengineering*, vol. 20, no. 2, pp. 409-411, June 2011.
- [3] F. Benedetto, G. Giunta, A. Toscano, and L. Vegni, "Dynamic LOS/NLOS Statistical Determination of Wireless Mobile Channels," *IEEE Vehic. Tech. Conf.*, pp. 3071-3075, Dublin, Ireland, April 2007.
- [4] I. Guvenc, C. C. Chong, and F. Watanabe, "NLOS Identification and Mitigation for UWB Localization Systems," *IEEE Wireless Commun. and Networking Conf. 2007*, pp. 1571-1576, Hong Kong, China, March 2007.
- [5] O. Punal, I. Aktas, C. J. Schnelke, and J. Gross, "Machine Learning-based Jamming Detection for IEEE 802.11: Design and Experimental Evaluation," *IEEE Int. Symp. on World of Wireless, Mobile, and Multimedia Networks*, In Proceedings, June 2014.
- [6] J. Joung, "Machine Learning-Based Antenna Selection in Wireless Communications," *IEEE Communications Letters*, vol. 20, no. 11, pp. 2241-2244, Nov. 2016.
- [7] A. P. Garcia, W. Kotterman, R. S. Thoma, U. Trautwein, D. Bruckner, and W. Wornitzer, "60 GHz In-cabin Real-time Channel Sounding," *Fourth Int. Conf. on Commun. and Networking in China 2009*, pp. 1-5, Xi'an, China, Aug. 2009.
- [8] L. Breiman, "Random Forest," Technical report, 2001.
- [9] R Core Team (2013), *R: A language and environment for statistical computing*. R Foundation for Statistical Computing, Vienna, Austria. URL <http://www.R-project.org/>.