ML Algorithm Performance to Classify MCS Schemes During UACN Link Adaptation

MST. NAJNIN SULTANA, (Student Member, IEEE), AND KYUNGHIC CHANG, (Senior Member, IEEE)
Department of Electronic Engineering, Inha University, Incheon 22212, Republic of Korea
Corresponding author: KyungHi Chang (khchang@inha.ac.kr)
This work was supported in part by the Ministry of Oceans and Fisheries, South Korea, through the Project titled Development of Distributed Underwater Monitoring and Control Networks.

ABSTRACT This research classifies the modulation and coding rate for link adaptation in Underwater Acoustic Communications Networks (UACNs). Recently, the UACN has become a promising technology for military, commercial, and civilian applications, as well as scientific research. However, we should minimize the dataset dimension for real-time implementation due to the sensor nodes’ energy limitations in the underwater environment. We used an Incheon sea trial’s measured dataset of 18 features, applying Principal Component Analysis (PCA) to select the dominant eigenvalue components in order to reduce the curse of dimensionality, and then selected 11 parameters. After that, we applied Machine Learning (ML) algorithms with different combinations of the parameters to separately classify the modulation and the coding rate and measured both individual and overall classification accuracy. The findings are compared with two Taean sea trial datasets with 11 features to finalize the selected parameters for link adaptation. For modulation classification, we observed 96.83% accuracy with the K-nearest Neighbors (KNN) algorithm in three-parameter and two-parameter cases. In coding rate classification, we found 100% accuracy with the KNN algorithm using the same three-parameter case. However, for the best fit among the three datasets, we finalized another three parameters at the expense of accuracy. To find the optimum threshold values for all modulation and coding rate labels, we used Rule-based (RB) 2D and 3D analysis. However, with a hard limit on non-overlapping data, at best, 35.51% classification accuracy was found for a 1/3 coding rate (Turbo code) with QPSK modulation, which showed much less reliability for RB analysis in a UACN, so it is not useful in this regard. Besides, our analysis shows data independence in the Doppler Spread (DS) and the Frequency Shift (FS), mitigating the time-variability channel’s challenge. We use the Gaussian distribution plot, a confusion matrix, multi-dimensional scatter plots, interpolated plots to analyze the data.

INDEX TERMS Boosted regression tree analysis, K-nearest neighbors, link adaptation, machine learning, multi-dimensional rule-based analysis, principal component analysis.

I. INTRODUCTION
Nowadays, the underwater sensor network (USN) is a promising technology among all sorts of next-generation wireless networks. A wireless link can be established with an acoustic signal or an optical signal, or by using OFDM MIMO systems. With minimum energy consumption, the USN is a promising technology for applications like communications between submarines; navy surveillance; disaster management after, for example, a tsunami, a cyclone, or an rip tide; research on marine life; finding undersea minerals; offshore oil exploration by autonomous underwater vehicles; and data exchange via undersea sensor networks for environmental monitoring, water level measurement in flooded areas or for rivers at risk; observation of water flow during bridge construction; nuclear power generation on rivers or the sea; deep seaport construction and maintenance; and data exchange among sensor nodes [1]. The conventional AWGN channel is critical for dealing with an underwater acoustic channel’s challenges, such as long delays, a larger Doppler spread...
due to the low propagation speed of sound (resulting in frequency and time-selective fading), temperature, viscosity, and streamflow direction. Still, channel modeling is an ongoing art due to such complexities. Parish et al. showed that the year-long time-varying nature of sound and speed, and the different weather conditions, cause a “bending” of acoustic pressure waves. In underwater acoustic communications (UAC), both attenuation and noise strongly depend on frequency, even though a transmitter–receiver (Tx-Rx) has a fixed distance [2].

Due to the inhomogeneous nature of UAC, modeling is usually either deterministic and physics-based or stochastic (or a combination of both). Socheleau et al. [3] studied the validity of stochastic modeling and inferred some statistical properties, such as the distribution of scattered random components, as well as time variation in shallow-water UACs, with a set of measured data in a coastal environment. Due to the extreme limitations on the available bandwidth in a UAC network, frequency reuse and cellular concepts are more appealing when trying to enhance a UAC network’s coverage and capacity [4]–[6]. A simulation-based link adaptation (LA) approach used a 12-path Rician fading channel for strategical LA [7]. In [8], the authors proposed a modern equipped with direct sequence spread spectrum signals of various coding rates and modulation orders. Adaptive selection of signals is achieved based on BER prediction via boosted trees. It speeds up communication by 10 to 20 times, compared to fixed-rate transmission. However, this proposed receiver showed vulnerability to high time-variable (high Doppler Spread) channels. A software-defined OFDM-based underwater acoustic (UWA) communications system demonstrates the superiority of adaptive transmission, where both modulation order/type and power on each subcarrier are selected based on channel conditions to maximize throughput. However, it is experimentally limited to a short-distance transmission setup [9]. Using new methods such as sparse adaptive convolution cores, time-domain turbo equalization and frequency-domain turbo equalization still cannot solve high computational complexity, and they have a low success rate for modulation classification. By applying Deep Learning, classification accuracy was found to be around 99% [10].

Link adaptation optimizes the reliability of communication networks through AMC. Previously, maximum likelihood techniques were used to classify MCS levels, but they have more computational and circuit complexities, mostly owing to the need to calculate natural logarithms in the likelihood function, and to the increased demand for additional signal samples. If the data size is not too large, a Machine Learning (ML) algorithm is a better option as a classifier [11].

In our previous work, we used all 11 parameters and 18 parameters of Taean and Incheon datasets, respectively [12]. In that case, with ML analysis the Boosted Regression Tree (BRT) algorithm showed the best performance at 99.9%. However, we cannot verify all 11 or 18 parameters for AMC classification in a real-time environment due to the lower energy consumption of buoy and underwater sensor nodes. This research’s main objective is to minimize the required number of implementation parameters while maximizing classification accuracy. We should choose, at best, two-dimensional or three-dimensional solutions with a maximum of four or five parameters for analysis. Due to lower accuracy by using the Taean dataset in previous work, here, we first take into consideration the Incheon dataset [13] with a 2 km Rx-Tx distance. Along with three types of modulation classifications (QPSK, 16QAM, 64QAM), we also consider two types of coding rate classification (1/2, 1/3) of Turbo code. At first, with Principal Component Analysis (PCA), we chose 11 to 15 parameters with dominant eigenvalues. These parameters are mostly like the available parameters at the Rx. Then, we analyzed those parameters multidimensionally (2D 3D, and more) with ML. Among the five algorithms like those used in [12] (also applied here), both Boosted Regression Tree and K-nearest Neighbors (KNN) algorithms performed outstanding classification in terms of accuracy in both modulation and coding rate classifications. Due to the fixed value for Doppler Spread (DS) and Frequency Shift (FS) during the Incheon sea trial, these parameters showed an irrelevant accuracy percentage, even with a single parameter. However, an analysis found that after training and validation sessions with any new input of these parameters, the accuracy measure will be zero. After further analysis, we found more relevant and variant parameters for AMC, which showed the system’s independence from these two parameters. The rule-based (RB) strategy has also been applied to classify MCS levels. With a $[-1.2\sigma, 1.2\sigma]$ interval Gaussian data distribution, although the Incheon data do not precisely follow this distribution, we used the similarities, and 2D and 3D RB showed acceptable overall accuracy. But the drawback is that lots of the data overlap among all modulations and coding rate levels. Then, we tried to find the thresholds for both cases of three types of modulations and two types of coding rate. However, with 2D and 3D rule-based analysis, we do not get an acceptable classification accuracy percentage with the defined thresholds. Figure 1 shows an example UAC network with such an arrangement of 2D, 3D, or more dimensional ML and RB classifiers. The analysis of 4D (and more) dimensional data achieved a better accuracy percentage in some cases, but we preferred to finalize 2D and 3D results. After that, the findings were compared with the 1 km and 3 km Taean sea trial datasets. Finally, the results are optimized as suitable for a real-time UACN with lower energy sensor nodes.

The paper was organized into eight sections. Section II describes the Underwater Acoustic Communications Network architecture and offers an Incheon dataset overview, while PCA analysis is described in Section III, and the significance of ML algorithms is described in Section IV. Modulation classification with Machine Learning analysis and 2D rule-based analysis is discussed in Section V, with Section VI presenting the analysis of ML and threshold findings from using 2D and 3D rule-based strategies for coding rate classification. Section VII shows the comparative results.
II. UAC NETWORK ARCHITECTURE AND DATASET OVERVIEW

Different kinds of network architecture concepts are available for underwater acoustic signal communications, such as wireless networks, sensor-based networks, optical wireless networks, MIMO-OFDM–based networks, green wireless networks, and 3D wireless networks [14]–[17]. The UAC network is still in the development stages, and that is why much attention and meaningful research is ongoing in this area. Because of bandwidth limitations, the cellular-based network structure for a UAC network is also a promising technology to implement the frequency reuse concept. Many literatures have been studied based on this concept, as used in the femtocell network, PS-LTE, LTE-R, 3GPP LTE-A networks, and so on [18]–[20]. We also consider this kind of frequency reuse scheme in a cellular UAC network for our research.

Figure 1 shows the UAC network architecture. Our primary concern is the underwater base station controller (UBSC), which is connected to three underwater base stations (UBSs) via interface 2, and these three UBSs are further connected to underwater equipment (UE) through interface 1. Interfaces 1 and 2 are wireless links where the 100 Hz acoustic wave is used as a carrier. We use the OFDM baseband signal from the transmission. The UBSC is connected to the core network (CN) via interface 3, which is LTE. Table 1 in [12] showed a list of system operating frequencies and bandwidths where A-DL to A-UL1 and A-UL2 connections are allocated to the UBSC and the UBS, respectively, with B-DL to B-UL3 to the UBS and UE, respectively. The UAC network layout is based on the basic cellular concept of spatial frequency reuse. Figure 1 from [7] showed the architecture considered for a one-tier UAC network layout and its details. The UAC network channel model has severe effects on system performance due to its challenging characteristics. So, it is vital to consider transmission loss, ambient noise, and multipath fading effects when analyzing the LA in order to increase bandwidth efficiency in a UAC network [21]. Indeed, the main concern with considering the cellular type of architecture under a UAC network channel model is to maximize the coverage and capacity within limited bandwidth resources. The OFDM frame used in the UAC network system is seven seconds in length with a 5 kHz bandwidth (BW), and it contains nine physical resource blocks (PRB) with two preambles. Each PRB contains six OFDM symbols, and each symbol is 0.1024 seconds in length and has a total of 512 subcarriers. In our experimental dataset, uplink data are measured (A-UL0); i.e., the transmitters are UBSs, the receiver is the UBSC, and FDD is used. The uplink OFDM symbol details are in Table 1.

The analysis has primarily been done based on two types of measured datasets; details are described in [12]. Here, we extend the analysis further with only the Incheon dataset because it showed better performance in the previous analysis. The Incheon dataset was measured during a 24-hour experiment conducted from July 5, 2017 [Starting Time:1438 (HM)], to July 6, 2017 [Ending Time: 1458 (HM)]. The experiment was carried out over a 2 km distance between a Tx and Rx in the Yellow Sea (Deokjeokdo, Incheon). The detailed parameters at the transmitting end are shown in Table 2.

The dataset was organized into 10,105 rows comprising the number of experiments, and 23 columns, where 18 columns held numeric values (considered features). One column is for modulation level, two columns are for channel coding level and coding rate description, one column is for uplink information, one is for time of the observation, and finally, one is for the BW of the OFDM baseband Tx signal’s date/time in the experiment. Each experiment was observed for four to six minutes, changing the combination of
TABLE 3. Parameters Available at the Receiver.

| Parameters                                      | Availability |
|-------------------------------------------------|--------------|
| CB (-10 dB) (corr 5), CB (-15 dB) (corr 5), CB  | x            |
| (-10 dB) (corr 9), CB (-15 dB) (corr 9), MED    | x            |
| (-15 dB), MED (-10 dB), SNR, RMS (-15 dB), RMS  | x            |
| (-10 dB)                                       | x            |
| Data Rate (bps), RP (time), RP (space), Pilot   | x            |
| (space), Pilot (time), Doppler Spread (-15 dB), | x            |
| Frequency Shift (-15 dB), Coded BER             | x            |
| Uncoded BER (Linear)                            | x            |

modulations (QPSK/16QAM/64QAM), the channel coding (Convolution/Turbo code), and the coding rate (1/2 or 1/3) for every Pilot Spacing (F, T) and every Repetition Pattern (F, T); plus setting the Data Rate (DR), Coded BER parameters at the transmitter, and measuring other parameters, such as Mean Estimated Delay (MED), RMS Delay Spread, Coherence Bandwidth (CB), SNR, RMS, RP, etc. It is notable that, here, Coded BER has a very low eigenvalue!

Covariance Method: The dataset’s covariance matrix is always a symmetric matrix with the variances on its diagonal and the covariances, off-diagonal. It defines both the spread (variance) and the orientation (covariance) of the data. The most significant eigenvector points the direction of the highest variance of the data where the eigenvalue represents its magnitude, equals the variance. The second-most significant eigenvector is always orthogonal to the largest eigenvector, and points in the direction of the second largest variance of the data [23].

Observations: In PCA with the covariance method, we linearly transform the Incheon dataset features into another principal component set. The component with the most significant eigenvalue is PC1, and the component with the second-highest eigenvalue is PC2, and so forth. We use the built-in pca function of MATLAB 2019a to do this. This function returns five parameters as outcomes. One of the important outcomes is the latent parameter, presenting the eigenvalues of the covariance matrix of the Incheon dataset that represent the variances of the principal components. Another is the explained parameter. It shows the rational variances of each principal component compared to the total variances. The term “total variances” means the summation of all the features’ variances, which is also the same as all the PCs’ total variances. A higher percentage for the rational variance of one component means it carries more information. Generally, the summation of rational variances for two or three PCs from the beginning of the list of all sequenced PCs covers 90% of the total variances, shown in the explained
parameters. Those are RMS eigenvalues, we finalize a few parameters as consistently key in the total variance of these components. According to the nine PCs in total because of the greater than 90% contribution values in PC8 and PC9 [Table 5, columns 2, 4]. We count Shift the coding rate. Table 4 summarizes the values of PCs to get more classification accuracy on the modulation and is not highly reliable. We need to take into consideration 8-11 by these two components, so dataset mapping by this method covered by PC1 and PC2, and too little information is carried PC. In this simulation, only about 35% of the variances are explained, but both types of covariance impact classification accuracy. In both methods, Doppler Spread and Frequency have positive covariances, and others have negative covariations (inverse relationships for PC1 and PC2). Table 5 summarizes the significant findings of all PCA methods. Some of them (inverse relationships for PC1 and PC2), and latent parameters. 

Using the biplot function, we can plot a two-dimensional orthonormal covariance graph aligned with the x-y axes of either PC1-PC2 or PC3-PC4, and so on, in order to present the original features from the first dataset. It shows the direction of the eigenvector, and the lengths of the vectors indicate the corresponding eigenvalues. Figure 3 presents such a plot for PC1 and PC2, where Data Rate (bps), Coded BER, and SNR have positive covariances (proportional relationships for PC1 and PC2), and CB, MED, and RMS have negative covariances (inverse relationships for PC1 and PC2). Table 5 summarizes the significant findings of all PCA methods. Some of them have positive covariances, and others have negative covariances, but both types of covariance impact classification accuracy. In both methods, Doppler Spread and Frequency Shift have very low eigenvalues until we find the highest values in PC8 and PC9 [Table 5, columns 2, 4]. We count nine PCs in total because of the greater than 90% contribution in the total variance of these components. According to the eigenvalues, we finalize a few parameters as consistently key parameters. Those are RMS, MED, Data Rate, Coded BER, CB, Pilot, SNR, RP, etc. Another notable point is that PCs are not really in chronological order in our analysis with respect to variances [Table 5, column 8], so reliability is not 100%

We followed the same procedure for the 1 km and 3 km Taean datasets. The parameters found in both datasets are summarized in Table 1 in the Appendix. For the Taean datasets, we only implemented the covariance method. The other two methods were not used for these second datasets. We sequenced the parameters from the highest value of the variances in PC1 up to PC9. The key parameters are almost similar, just as they are in the Incheon dataset; for example, Coded BER, RMS Delay Spread, MED, Pilot, EVM, CB, RP, SNR, Data Rate, etc.

**SVD Method:** The PCA process reflects computational complexity. The real-time data matrix contains thousands of pieces of data, so it is hard to compute the covariance matrix’s eigenvalues, which may sometimes cause errors, such as rounding errors [24]. According to the mathematical definition of the Singular Value Decomposition method, let the Incheon dataset is $X$, denoted as a $10,104 \times 18$ matrix of real-valued data and rank $r$, where (without loss of generality) $m \geq n$, and therefore, $r \leq n$; $x_{ij}$ is the element of each $i^{th}$ observation row and $j^{th}$ feature column. The elements of the $i^{th}$ row of $X$ form the $n$-dimensional vector $g_i$, and the elements of the $j^{th}$ column of $X$ form the $m$-dimensional vector $a_j$. The equation for singular value decomposition of $X$ is the following:

$$X = USV^T$$  \hspace{1cm} (1)

where $U$ is an $m \times n$ matrix, $S$ is an $n \times n$ diagonal matrix, and $V^T$ is also an $n \times n$ matrix. The columns of $U$ are called the left singular vectors, $\{u_i\}$, and form an orthonormal basis for the feature columns, so that $u_i \cdot u_j = 1$ for $i = j$, and $u_i \cdot u_j = 0$, otherwise. The rows of $V^T$ contain the elements of the right singular vectors, $\{v_k\}$, and form an orthonormal basis for all.

**TABLE 4. Values for Explained and Latent Parameters in All the PCs.**

| PC    | 1   | 2   | 3   | 4   | 5   | 6   | 7   | 8   | 9   |
|-------|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| explained | 36.5 | 14.9 | 10.44 | 9.13 | 6.58 | 5.8 | 5.5 | 5.2 | 5.0 |
| latent  | 5.99 | 2.68 | 1.88 | 1.64 | 1.78 | 1.64 | 0.99 | 0.94 | 0.9 |

**FIGURE 3. Orthonormal plot of the covariances for principal components (PCs) 1 and 2.**
TABLE 5. Parameters Found in PCA Analysis (Incheon Dataset).

| Parameter                  | Dominant Parameters found by the power method | Dominant parameters found by the covariance method with eigenvalues from different PCs | Parameters found by the SVD method |
|----------------------------|-----------------------------------------------|-----------------------------------------------------------------------------------|-----------------------------------|
| Data Rate (bps)            | 8327.7                                        | 0.37 (pc1)                                                                        | Pilot 5.493                       |
| CB                        | 4091.1                                        | 0.36 (pc1)                                                                        | RP 1.88                           |
| MED                       | 49.8352                                       | 0.39 (pc2)                                                                        | Data Rate (bps) 1.18               |
| SNR                       | 24.2487                                       | 0.4 (pc2)                                                                         | Coded BER (Linear) 0.94            |
| RMS                       | 17.1811                                       | 0.46 (pc2)                                                                        | Data Rate (bps) 0.94              |
| RP                        | 8.9971                                        | 0.637 (pc3)                                                                       | CB 0.43                           |
| Pilot                     | 5.9994                                        | 0.475 (pc4)                                                                       | SNR 0.58 (pc9)                     |
| Doppler Spread            | 5.8245                                        | 0.642 (pc8)                                                                       | Frequency Shift 0.642 (pc8)        |
|                           |                                               | 0.648 (pc9)                                                                       | Doppler Spread 0.648 (pc9)        |

received parameters in the row. The elements of $S$ are only nonzero on the diagonal and are called the singular values. Thus, $S = \text{diag}(s_1, \ldots, s_n)$. We can express the SVD of $X$ as

$$X = \sum_{k=1}^{l} u_k s_k v_k^T$$

which is the closest rank-1 matrix of $X$.

There is a direct relationship between PCA and SVD in cases where principal components are calculated from the covariance matrix.

If each row of $X$ is centered, $X^T X = \sum a_i a_i^T$ is proportional to the covariance matrix of the variables of $a_i$. In this case, left singular vectors $\{u_k\}$ are the same as the principal components of $\{a_i\}$. So, $s_k^2$ is again proportional to the variances of the principal components. The matrix $SV^T$ again contains the principal component scores, which are the coordinates of the features in the space of principal components [25].

**Observations:** In the SVD method, we have a separate covariance matrix and eigenvalue. We can select these parameters roughly from the covariance matrix: Pilot in both space and time, $RP$ in both space and time, Data Rate (bps), and MED, RMS, and CB, etc., based on their eigenvalues. In the simulation, we choose the number of top eigenvalues, topeig = 12, and count the first 10 PCs (k= 10), and then, it covers 95% of the variances.

In real-time applications, 2D or 3D data dimensions are feasible to implement due to sensor node (UE) energy limitations in the undersea environment. Due to the power constraint in a UAC on the sensor node (or UE), on the UBSs, and on the UBSC, we finally selected two or three parameters. For less energy consumption by sensor nodes in the UAC network with either a MIMO or an OFDM system, a lot of research has been done [26]–[29]. In this study, via PCA analysis, we sort out the major parameters from among all the received parameters that have the most significant roles in identifying the modulation and coding rate levels. Now, we are going to measure classification accuracy with combinations of the above-mentioned selected parameters.

**IV. MACHINE LEARNING ALGORITHMS FOR LINK ADAPTATION IN UAC NETWORKS**

Feature parameters have too many overlapping values and are random due to the fast-fading channel. Limited research has already been done on the stochastic model, statistical characterization of channel modeling, the eigenpath UAC channel model, the statistical time-varying model, etc. [12], [3],[30]–[34]. Again, considering the sea environments, such as heavy tides and windy conditions, some analysis has been done [35], [36]. Few works have been completed using optical or magneto-inductive wireless channel links instead of low-frequency acoustic links [37], [38]. However, it is still more challenging for accurate formulation of the UAC channel model. The challenge of transmission–CSI feedback timing mismatch was already discussed [12]. On the other hand, due to the fixed distances of Tx-Rx pairs, the Doppler Spread and Frequency Shift were fixed for every observation period of four, six, or 10 minutes, but were changed to different values for every trial. With multivariate data features, we need to make a classification decision based on a multistage decision tree where each stage utilizes a different feature. This increases the multi-dimensional complexity. Using a rule-based strategy to find appropriate thresholds at every stage for modulations, along with coding rates, is very difficult. Besides, to get LA from measured data may be possible, but the reverse is impossible, which is the case in LTE or on any other known wireless channel. We observed that the relationship between SNR vs. Data Rate shows inverse random behavior. As a result, the statistical ML approach is
a good option to test and validate a huge amount of measured training data and to classify the modulation and coding rate more precisely.

Here, we follow the same set of algorithms, without loss of generality [12]. Those algorithms showed comparatively better performance over there. Nevertheless, after a lot of trial and error, we finalized three of them: K-Nearest Neighbors, Boosted Regression Tree, and Support Vector Machine (SVM) algorithms. Other algorithms did not perform well in this research. Zhu et al. [11] described algorithm details of the KNN and SVM classifiers. In the KNN approach, both training signal and testing signal share the same signal source for the generation of reference feature values; thus, reference feature space is an accurate representation of the feature distribution of the testing signal. The classifier requires evaluating distances between the testing signals and reference signals, so the Euclidean distance is used for this purpose. Although KNN suffers from dimensionality, after applying PCA and with a reduced selection of 11 features, the performance was better than in previous research [12]. Although KNN is capable of doing multi-class classification, it suffers from the complexity in the distance calculation. Another disadvantage of the KNN classifier is that the classification decision–making features are not weighted. Therefore, one feature with a relatively sparse distribution between different modulations may dominate the distance evaluation.

BRT uses the boosting method in which the input data are weighted in subsequent trees. The weights are applied in a manner so that the previous trees’ poorly modeled data have a higher probability of being selected in the new tree. That means that after the first tree is fitted, the model will measure the prediction error of that tree to fit the next tree, and so on. By measuring the fit of the previous trees built, the model continuously tries to improve its accuracy. This sequential approach is unique to boosting [39], [40]. Boosting is a numerical optimization technique for minimizing the loss function by adding, at each step, a new tree that best reduces (steps down the gradient of) the loss function. For our data analysis, BRT uses a logit boost when using two parameters; with three parameters, AdaBoost2 is used as a booster. This algorithm is also robust to missing values and outliers.

SVM achieves classification by finding the hyperplane that separates data from different classes. The hyperplane is optimized by maximizing its distance to the signal samples on each side of the hyperplane. Depending on the nature of the signal being classified, the SVM classifier uses either linear kernel (linear programming) or non-linear RBF Kernel versions (quadratic programming). Nevertheless, AdaBoost only corresponds to linear programming. Gaussian kernels are universal kernels, i.e., their use with appropriate regularization guarantees a globally optimal predictor, minimizing both estimation and approximation errors of a classifier. Here, approximation error refers to the error incurred by limiting the space of the classification models over which the search is performed, and estimation error refers to an error in estimating the model parameters. The linear SVM classifiers have linear kernels. The reason for the effectiveness of SVM and AdaBoost is that they find linear classifiers for too-high dimensional spaces. When overfitting is addressed by maximizing the margin, the computational problem associated with operating in high-dimensional space is dealt with by the SVM through the method of the kernel. Kernels allow the algorithms to perform low-dimensional calculations that are mathematically equivalent to inner products in a high-dimensional “virtual” space. In this case, the boosting approach employs a greedy search; the weak learner is an oracle for finding coordinates with a non-negligible correlation to the label [41].

Compared with the KNN classifier, the SVM only needs to use the training signal when establishing the separating hyperplane. Once the hyperplane is optimized, there is no need to involve the training signal in any further calculations. The benefit is that the computation needed in the testing stage is relatively inexpensive, compared with KNN. The Gaussian kernel’s choice is better than the other two types of available kernels, like linear and polynomial kernels. With an increase in the order of the polynomial kernel, the size of the function class increases. The Gaussian kernel is non-parametric. A polynomial kernel is parametric. In a way, with a non-parametric kernel, we can put more and more data into the complexity cost. In contrast, a parametric model’s size is fixed, so the model is saturated after a certain point. So, asymptotically, for extensive dimensional data with no prior information about the function, invariances, data distributions, and so on, a non-parametric method is always better [42].

V. MODULATION CLASSIFICATION

Any classification can be done by applying either Machine Learning or rule-based algorithms. Here, we apply both for individual specific purposes. A new set of rules is generated by the ML process for learning the given training data with specific learning algorithms. The process predicts the classes in the given Incheon or Taean dataset. It makes a model for every new dataset. The classes are often referred to as targets, labels, or categories; in this case, the modulation techniques are QPSK, 16QAM, and 64QAM. The ultimate objective is for the machine to identify the target/label of the newly given data (the test data). On the other hand, a rule-based strategy can predict the classes by using threshold values sequentially. Here, making an analogy with the Gaussian distribution curve, applying the sigma interval concept to the parameters, and after several trials for shrinkage and expansion of each parameter, we set the parameters’ threshold values for every modulation label. The same procedure will also be repeated for coding rate classifications, as explained separately in Section VI.

A. MACHINE LEARNING APPROACH TO MODULATION CLASSIFICATION

Now, we apply ML algorithms to the different combinations of both PCA parameters and also PCA-excluded parameters.
We trained the algorithms with 70% of the Incheon dataset of 10,104 observations, with 30% of the data used as test data. This arrangement was used for both modulation and coding rate classifications. For three types of modulation label, the Boosted Regression Tree used AdaBoostM2, and for two types of coding rate classification, we used LogitBoost. All algorithms showed good performance, but from among them, the K-Nearest Neighbors algorithm performed best.

We observed that the combination of these parameters showed extremely good overall accuracy for modulation classification, for example, with \([DR\ (bps), \ CB(−15\ dB)(corr\ 5)],\ [DR\ (bps), \ Coded\ BER, \ CB\ (−10\ dB)(corr\ 5), \ CB(−15\ dB)\ (corr\ 5), \ Doppler\ Spread(−15\ dB), \ Pilot\ Space, \ Pilot\ Time, \ RP\ Space, \ RP\ Time],\ [DR(bps), \ CB(−10\ dB)(corr\ 5), \ Frequency\ Shift\ (−15\ dB)],\ [DR\ (bps), \ CB(−10\ dB)(corr\ 5), \ CB(−15\ dB)(corr\ 5), \ Frequency\ Shift\ (−15\ dB)]\). Different accuracy or performance terminologies are described in Section A of the Appendix. One noticeable point is that a combination containing \(Frequency\ Shift\ (FS)\) or \(Doppler\ Spread\ (DS)\) showed performance of 100% and 99.93%, respectively, although these parameters were not selected through PCA analysis. With PCA parameter combinations, the maximum overall accuracy was 87%. Then, we made a thorough investigation of these two parameters. We observed that these two values are constant for each experimental set of 36 observations (one period of observation time was four to six minutes) with different modulation and coding rate combinations, along with \(PS\) and \(RP\) combinations. For every experiment, the values were set to other constant values. So, there were 280 different combinations of \(FS\) and \(DS\) values. These values are independent of observation time variations. In communication theory, modulation and coding rate do not depend on \(DS\) and \(FS\). These are Tx-Rx relationship characteristics, such as in-between distance, the nature of mobility, or of Tx and Rx, etc. However, after measurement, when we generate a data file, the values just act as text numbers for Machine Learning. The constant nature for a certain specific time showed 100% accuracy in modulation classification. Again, when we tried the combination of \([DR\ (bps), \ Coded\ BER\ (Linear), \ Frequency\ Shift\ (−15\ dB)]\), the accuracy dropped to 97.16%. This happened because \(Coded\ BER\) has less variance and data overlap for all modulations. All trials of \(DS\) and \(FS\), along with \(Coded\ BER\), did not go extreme for the same reason.

One point should be made: We only discuss the overall accuracy percentage. The individual modulation and coding rate classifications are described later, for ML with the confusion matrix, and for rule-based analysis, numerically. However, we never measured the false alarm rate in the rule-based approach. In the ML approach, it is automatically measured by the confusion matrix.

1) 1D ANALYSIS
To check an individual parameter’s impact on classification accuracy, we conducted a simulation with one parameter. Here also, \(FS\) and \(DS\) showed a 100% contribution with the Boosted Regression Tree algorithm. The reason is that when we train with 70% of the data using 280 different values for \(FS\) and \(DS\), then in the 30% test data, the BRT algorithm correctly detects the modulation labels 100% of the time. However, for new future measured data, those parameters should have another new, constant value, and then, classification accuracy will be 0%. For that reason, we can avoid further consideration of \(DS\) and \(FS\), and show a minimum eigenvalue in PCA analysis. In both approaches, irrelevance was proven for classification. However, with only one parameter, KNN showed satisfactory performance with any delay parameter, such as \(RMS\ Delay, \ MED, \ CB, \) and \(SNR\). Among them, \(MED\) showed the highest contribution in classification accuracy at 96.14%, and the second highest was 95.94% with \(SNR\). SVM and Naïve Bayes algorithms cannot provide acceptable performance with a single parameter. To find more reasonable accuracy with the ML approach, we combined two parameters differently, as well as three parameters, and four or five parameters, etc. Because \(SNR\) shows better performance singly, we tried the maximum number of combinations with \(SNR\). We observed that several combinations showed an up-to-the-mark percentage in overall accuracy for modulation classification. However, \(Data\ Rate\) and \(Coded\ BER\) performed the worst in every combination, even with \(SNR\), except for \(DR\) with \(CB\), which resembled outstanding performance from all ML algorithms. It was about 96.83% with the KNN algorithm and 86.3% with BRT. This result ensures the credentialism of \(Coherence\ BW\ (CB)\).

Figure 4 is an example of a confusion matrix that shows individual modulation classification accuracy, false alarm rate, and overall accuracy for the combination of \(SNR\) and \(CB\) with KNN. There are more false detections as 16QAM;
for example, false detection of 64QAM as 16QAM is 0.7\% and false detection of QPSK as 16QAM was also 0.7\%. This result indicates that 16QAM data has too much overlapping with both QPSK and 64QAM. Here, QPSK achieved highest percentage of accuracy.

However, in many cases under all the experiments, 16QAM was falsely detected the most, and it had a much lower percentage in classification accuracy due to its narrow non-overlapped range. This will be clear when we examine the Gaussian curve under rule-based analysis in the next section.

2) MULTI-DIMENSIONAL ANALYSIS

The combination of SNR and CB performed best due to the significant contribution by both parameters. The performance from the combination of DR and Coded BER jumped to the highest in three cases when CB was added to these parameters. The combination of [Data Rate (bps), CB (−15 dB) (corr 5), Coded BER (Linear)] showed competence with all ML algorithms. Besides, when DR was added to [MED, SNR] or [CB, SNR] combinations, the reflected overall accuracy percentage was also better. Here, we ignore the performance from the arrangements with FS and DS because they were already discarded. With [RMS, SNR] or [CB, SNR], Coded BER also showed acceptable performance. The combination of four parameters [Data Rate (bps), CB (−15 dB)(corr 9), RP Space, SNR] generated 91.52\% accuracy in modulation classification, but increased dimensional complexity. For five parameters, that was also true. We tried to confine our findings to those from two or three parameters.

Now, we compare the primary findings with the parameters in Table 4, which are estimated at the receiver. We can omit parameters that are not on the list. In addition, because we use the Turbo code signal, the Uncoded BER (Linear) parameter is irrelevant. We summarize the acceptable parameters as CB (−10 dB) (corr 5), CB (−15 dB) (corr 5), CB (−10 dB) (corr 9), CB (−15 dB) (corr 9), MED (−15 dB), MED (−10 dB), SNR, RMS (−15 dB), and RMS (−10 dB) for the next stage of coding rate classification.

B. RULE-BASED STRATEGY FOR MODULATION CLASSIFICATION

1) 1D ANALYSIS

Besides the ML approach, we also tried to find the threshold values for contributor parameters; those combinations had good percentages in overall accuracy from ML analysis. We start this journey with a one-dimensional rule-based strategy. We conducted a Gaussian distribution analysis to find the mean, variance, and plot of the probability density function for a normal distribution, as illustrated in Figure 5. From the plot, according to the six-sigma graph [43], we want to calculate the percentage of data coverage. The definition for the percentage of data coverage is given in Section A of the Appendix to within ±1σ, ±1.5σ, and ±2σ intervals, and so on. Not all plots resemble an exact Gaussian distribution. First of all, each parameter is summarized with individual data size, minimum and maximum values, mean, variance, standard deviation, ±σ truncated data size, and the lower bound and upper bound values for every modulation scheme: QPSK, 16QAM, and 64QAM. The MATLAB pdf function with a normal distribution was used to plot the six-sigma graphs. Finally, the percentage of data coverage within the ±σ interval was counted. Figure 5 shows the overlapping points from the three plots for QPSK, 16QAM, and 64QAM, and the threshold of each modulation for every parameter was found. The threshold is an essential measure for LA. For Coherence BW(−15 dB), RMS Delay Spread (−15 dB), MED (−15 dB), we have only one threshold for each. We cannot differentiate between QPSK and 16QAM using Coherence BW(−15 dB), RMS Delay Spread, and MED. There are two threshold values due to three types of modulation classification labels. The definitions of thresholds are as follows.

Threshold1: determine the upper truncation value for QPSK level detection and the lower bound for 16QAM.

Threshold2: determine the upper truncation value for 16QAM level detection and the lower bound for 64QAM.

The percentage for individual modulation classification is defined as:

\[
\text{Data\_percentage\_QPSK} < \text{Threshold1}
\]

\[
\text{Threshold1} < \text{Data\_percentage\_16QAM} < \text{Threshold2}
\]

\[
\text{Data\_percentage\_64QAM} < \text{Threshold2}
\]
Threshold Value for *Data Rate*:  
We drew the pdf plot of the normal distributed original data for \([-\sigma, \sigma]\), \([-1.5\sigma, 1.5\sigma]\), and \([-2\sigma, 2\sigma]\) intervals, and found the threshold (crossing) points for every modulation. The percentage of data coverage within the \(\pm \sigma\) interval seems to be an outstanding result of above 90%, but because of data overlap, no real-time implementation of LA is possible with this result, and it shows the ambiguity of the real-time scenario, which we found in the LTE case. So, we further tried to find the optimum threshold values by observing all Gaussian plots (possible crossing points of QPSK-16QAM and 16QAM-64QAM) for the other two uniform sigma intervals. Then, we defined the optimum threshold value based on equation (3) by using the minimum value as the lower bound for QPSK and the maximum value as the upper bound for 64QAM. Then, we calculated the individual modulation classification accuracy and the overall classification accuracy. The thresholds that reflected more accuracy in all cases were selected as the final threshold pair. For the above three intervals in 16QAM and 64QAM, the percentage of data coverage went down to much lower values: 13.82% and 19.24%. Overall accuracy was similar in all three cases. That means most of the data for these two modulations overlap. However, for QPSK detection, performance remained the same as the \([-\sigma, \sigma]\) interval: 90.15%! We need to account further for the sigma interval increasing the data coverage percentage and the overall accuracy. We tried the non-uniform intervals of \([0, \sigma]\), and \([0, 0.7\sigma]\), getting similar results. For the latter, the detection of 16QAM and 64QAM slightly increased by 15.29% and 23.43% at the expense of decreased QPSK detection performance. But overall accuracy went down. So, we finalized the \(\pm \sigma\) interval result as the threshold-pair for *Data Rate*. In the 2D analysis, we used these thresholds for *DR*.

**Threshold for *Coherence BW* (-15 dB):**  
QPSK and 16QAM pdf plots overlap too much. We cannot distinguish these two modulation labels by using *CB*. Furthermore, at the overlap point of 100 with 64QAM, there is little accuracy. So, we tried to set the threshold point manually, and after several trials, set it to 30, where overall accuracy with *CB* was 56.12%. Although it is only one threshold point, this parameter showed competence in modulation accuracy through the ML and RB strategies.

**Threshold for *SNR*:**  
In ML analysis, *SNR* with the KNN algorithm showed extraordinary performance at 95.96%. By changing different sigma interval ranges, we tried to fix its threshold values. Threshold1 and Threshold2 at 7.6 and 5.7, respectively, showed the highest overall accuracy (49.44%) among all trials. However, it is notable that the threshold values are opposite, i.e., Threshold1 for QPSK is the upper truncated value of 16QAM, and Threshold2 for 64QAM is the lower truncated value 16QAM. Within this interval, again, 16QAM had lower accuracy.

### TABLE 6. Threshold Values of all Parameters for Modulation Classification.

| Modulation | Data Rate (bps) | Coherence BW | SNR | MED | RMS Delay | Coded BER (Linear) |
|------------|-----------------|--------------|-----|-----|-----------|-------------------|
| QPSK       | DR < 980       | CB > 30      | 7.6 | MED | RMS < 28.9 | BER < 0.2          |
| 16QAM      | <1694          | CB > 30      | 5.7 | MED | RMS < 28.9 | BER < 0.2          |
| 64QAM      | 1694 > DR      | CB < 30      | 5.7 | MED | RMS < 28.9 | BER < 0.2          |

**Threshold Value for *MED*:**  
Like *RMS Delay Spread*, *MED* also has one threshold value and a parameter very similar to *RMS Delay Spread*. So, *MED* has the same impact on modulation classification accuracy. It has an individual classification accuracy as same as the overall classification percentage. The threshold value is 28.9.

**Threshold Value for *RMS Delay Spread*:**  
There is only one threshold value for *RMS*, so we cannot differentiate QPSK and 16QAM with it; we could only differentiate 64QAM. We found the optimum threshold values from different sigma intervals, the classification accuracy for each modulation, and the overall accuracy. The \([-\sigma, \sigma]\) interval provided a better detection percentage for QPSK and 16QAM, but a very much lower percentage for 64QAM. The overall accuracies were also poor here. Finally, we chose a threshold value of 7.35 for the *RMS Delay Spread* parameter.

**Threshold Value for *Coded BER (Linear)*:**  
QPSK detection showed better performance in all types for the proposed sigma interval. It was maximized under the trial with a half interval. For \([-1.5\sigma, 1.5\sigma]\) and \([-2\sigma, 2\sigma]\), the detection percentage for 16QAM went down to 8.5% and went up to 18.62% during the half sigma interval. It showed a nature opposite to *DR* and *RMS Delay Spread* detection, i.e., a too narrowly non-overlapped interval for 16QAM. Detection of 64QAM increased well enough with a larger sigma interval. The total accuracy was still inadequate! The threshold values for *Coded BER* are 0.2 and 0.31. In Table 6, all threshold values are summarized.

One parameter is not sufficient to accurately classify the modulation, because the percentage accuracy for every parameter was very much less than the system efficiency level. We might try a 2D or 3D combined analysis for checking performance further. Our main objective is for real-time implementation of the underwater communications system; we need two or three of the most reliable parameters that decrease the data volume while increasing the accuracy. That means we will have a more reliable real-time link adaptation. From Figure 5, we can also see that the truncated interval cuts most of the 16QAM data.

2) 2D ANALYSIS  
Parameter combinations selected primarily from ML analysis are \([DR (bps) and CB (-15 dB) (corr 5)]\), and \([SNR and CB (-15 dB) (corr 5)]\). We conducted 2D rule-based analysis
M. N. Sultana, K. Chang: ML Algorithm Performance to Classify MCS Schemes During UACN Link Adaptation

FIGURE 6. The scatter plots from 2D rule-based analysis: (a) SNR vs. Data Rate (bps), (b) Data Rate (bps) vs. CB (−15 dB) (corr 5), and (c) SNR vs. CB (−15 dB) (corr 5) for real-time implementation. Another three-parameter combination considered later is [Data Rate (bps), CB (−15 dB) (corr 5), Coded BER (Linear)], selected by the ML algorithm analysis. For 2D analysis, we followed this algorithm to cover a higher percentage of the data and improve accuracy in the threshold range.

There is no direct relationship between Data Rate, SNR, and Coherence Bandwidth, in theory. Coherence Bandwidth has an inverse relationship with RMS Delay Spread. Besides, in our Incheon dataset, SNR showed a random inverse relationship with Data Rate. The 2D scatter plot in Figure 6(a) shows this abnormal nature. Now, from the 2D ML analysis in which combinations had a better overall accuracy, the percentage of classification accuracy and the percentage of overall accuracy were also verified with the rule-based strategy. Here, we primarily used the [−σ, σ] interval data, where every modulation has an overlap. By using the algorithm in Table 7, we have a better percentage for DR and CB (−15 dB) (corr 5) and SNR and CB (−15 dB) (corr 5).

The first pair had 90.63% in the overall classifications; 64QAM had a maximum classification accuracy of 99.23% with the same combination of parameters. One noticeable point is that the CB data distribution has an almost similar overlap for the three modulations, but from 2D analysis of the sigma interval, combined with Data Rate, we can classify the modulation with a high value of 90.63%! The second pair, [SNR and CB (−15 dB) (corr 5)] achieved an accuracy of 82.24%. Two other combinations had a lower performance than an acceptable level for system efficiency. Figures 6(b) and 6(c) show the 2D scatter plots of these two combinations. We see that we have some separation and some overlapping of data in the two-dimensional rule-based scatter plots. Data Rate (bps) and SNR have data separated well enough for modulation classification, but CB shows too much overlap. Another important point is that SNR has a low value for higher-order modulation, and vice versa, which is impractical! Still, SNR is valuable for modulation classification.

Later we will apply fine-tuning with an exact threshold value to find the real accuracy percentage. Before that, we need to sort out which parameters have more significance on the coding rate, and then, we will finalize the parameters and apply threshold values to those parameters for the final output. The algorithm of the 2D rule-based strategy is presented in Table 7. Table 8 accumulates all percentages of classification accuracy for every modulation after 2D RB analysis, and all overall accuracies found from ML and rule-based analyses are summarized in Table 9. Figure 6 shows the 2D scatter plots for all combinations. Figure 6(a) shows the ambiguity in SNR with the Data Rate. In communication theory, there is no relationship between Data Rate and CB. Because of SNR, every plot has an inverse adaptation.

### TABLE 7. Pseudocode of the Accuracy Algorithm for the RB Strategy.

**Input:** [-σ, σ] data interval for two parameters, PR1, PR2
Original data size of QPSK, 16QAM, 64QAM
for PR1: QPSK, [minimum, Threshold1]
for PR2: QPSK, [minimum, Threshold1]
step 1: count truncated Newer datasize
step 2: calculate % of classification accuracy
  = (Newer datasize/ Original datasize)*100
for PR1: 16QAM, [Threshold1, Threshold2]
for PR2: 16QAM, [Threshold1, Threshold2]
repeat step 1 and step 2
for PR1: 64QAM, [Threshold2, maximum]
for PR2: 64QAM, [Threshold2, maximum]
repeat step 1 and step 2
then step 3: calculate % of overall accuracy
  = [sum(Newer datasize of (QPSK, 16QAM,
  64QAM))/ sum(Original datasize of
  (QPSK, 16QAM))]*100
end
end

**Output:** % of Classification Accuracy,
% of Overall Accuracy
TABLE 8. Accuracy Measure from 2D Rule-based Analysis by Using \([-\sigma, \sigma]\) Values.

| Combination of Parameters | Modulation | Classification Accuracy | Overall Accuracy |
|---------------------------|------------|-------------------------|------------------|
| DR (bps), CB (-15 dB) \(\text{corr 5}\) | QPSK       | 85.97%                  |                  |
| 16QAM                     | 87.09%     | 90.63%                  |
| 64QAM                     | 99.23%     |                         |
| QPSK                      | 91.33%     |                         |
| SNR, CB(-15 dB)(corr 5)   | 16QAM       | 69.36%                  | 82.24%           |
| 64QAM                     | 85.23%     |                         |
| QPSK                      | 45.62%     |                         |
| SNR, MED(-15 dB)          | 16QAM       | 38.17%                  | 47.54%           |
| 64QAM                     | 58.99%     |                         |
| QPSK                      | 72.73%     |                         |
| SNR, RMS Delay Spread (-15 dB) | 16QAM   | 55.23%                  | 65.59%           |
| 64QAM                     | 68.17%     |                         |

TABLE 9. ML and 2D Rule-based Analysis for Modulation Classification.

| Parameters \([-\sigma, \sigma]\) | ML Analysis | Rule-based Strategy |
|----------------------------------|-------------|---------------------|
| SNR, CB (-15 dB) \(\text{corr 5}\) | 96.83% \(\text{KNN}\)   | 82.24%              |
| SNR, MED (-15 dB)                | 95.74% \(\text{KNN}\) | 47.54%              |
| SNR, RMS (-15 dB)                | 95.45% \(\text{KNN}\) | 65.59%              |
| Data Rate (bps), CB (-15 dB) \(\text{corr 5}\) | 86.3% \(\text{BRT}\) | 90.63%              |
| Data Rate (bps), SNR             | 65.98% \(\text{BRT}\) | 76.47%              |
| SNR, Coded BER                   | 64.56% \(\text{BRT}\) | 59.09%              |
| Data Rate (bps), Coded BER       | 57.77% \(\text{BRT}\) | 68.82%              |

For the lower value of SNR, we have 64QAM, and for the higher value of SNR, QPSK. These plots are drawn with the \([-\sigma, \sigma]\) interval. For a more separable plot, we further analyzed it after coding rate classification.

VI. CODING RATE CLASSIFICATION

In the Incheon dataset, there are two types of coding: Convolution code and Turbo code. Moreover, coding rates are 1/2 and 1/3. In this research, we only consider Turbo code. So, the total data size was reduced to 1728. By taking 30% as test data, the size of the test data for coding rate was 518. For coding rate classification, we also used the same two techniques: the ML approach and the RB approach.

Finally, after 1D RB analysis, we applied multidimensional analyses for both modulation and coding rate classification.

A. MACHINE LEARNING APPROACH TO CODING RATE CLASSIFICATION

At first, we tried the eleven parameters available at the receiver (see Table 4) for every single modulation case. We noticed that Data Rate (bps) is not available at the receiver, even though this parameter had the greatest impact on modulation classification with CB in the analysis from the previous section. We obtained 100% coding rate classification accuracy with the Boosted Regression Tree algorithm with QPSK modulation. KNN also classified accurately at a good percentage (97.1%) with same modulation case. SVM, Gaussian, and Naïve Bayes were also able to classify at a good percentage. Figure 7(a) shows the confusion matrix for the eleven parameters in case of QPSK modulation. However, we should minimize the required number of parameters to classify the coding rate with maximum accuracy, considering the realities of energy constraints. We already...
found a significant impact from two or three parameters in modulation classification. So, we decided to start with three parameters here. At first, we fixed QPSK. Then we tried different combinations, such as $\text{[MED}(-15\,\text{dB}), \text{RMS}(-15\,\text{dB}), \text{CB}(-15\,\text{dB})(\text{corr} 5)]$, $\text{[MED}(-15\,\text{dB}), \text{RMS}(-15\,\text{dB}), \text{SNR}]$, $\text{[MED}(-15\,\text{dB}), \text{CB}(-15\,\text{dB})(\text{corr} 5), \text{SNR}]$, and $\text{[RMS}(-15\,\text{dB}), \text{CB}(-15\,\text{dB})(\text{corr} 5), \text{SNR}]$ with all kinds of ML algorithms. Also, we tried four parameters: $\text{[MED}(-15\,\text{dB}), \text{RMS}(-15\,\text{dB}), \text{CB}(-15\,\text{dB})(\text{corr} 5), \text{SNR}]$. Then, we conducted trials with 16QAM and 64QAM. With three parameters, we can classify the coding rate at up to 98.3% accuracy with BRT from $\text{[RMS}(-15\,\text{dB}), \text{CB}(-15\,\text{dB})(\text{corr} 5), \text{SNR}]$ with QPSK modulation. Classification accuracy of 98.26% was achieved with the BRT algorithm in several cases, such as QPSK: $\text{[MED}(-15\,\text{dB}), \text{RMS}(-15\,\text{dB}), \text{SNR}]$, QPSK: $\text{[MED}(-15\,\text{dB}), \text{CB}(-15\,\text{dB})(\text{corr} 5), \text{SNR}]$, 16QAM: $\text{[MED}(-15\,\text{dB}), \text{RMS}(-15\,\text{dB}), \text{SNR}]$, $\text{[MED}(-15\,\text{dB}), \text{CB}(-15\,\text{dB})(\text{corr} 5), \text{SNR}]$, and QPSK: $\text{[MED}(-15\,\text{dB}), \text{RMS}(-15\,\text{dB}), \text{CB}(-15\,\text{dB})(\text{corr} 5), \text{SNR}]$. With QPSK modulation in the above-mentioned three combinations and four parameters, we achieved 97.10% overall accuracy with the KNN algorithm. Although we achieved 100% accuracy in case of 16QAM with both BRT and KNN using 11 parameters for coding rate classification, and 100% with BRT in case of QPSK and 64QAM, we ignore this result due to the curse of dimensionality. So, we finalize the parameter combinations as follows:

$\text{[MED}(-15\,\text{dB}), \text{RMS}(-15\,\text{dB}), \text{SNR}]$, $\text{[MED}(-15\,\text{dB}), \text{CB}(-15\,\text{dB})(\text{corr} 5), \text{SNR}]$, and $\text{[MED}(-15\,\text{dB}), \text{RMS}(-15\,\text{dB}), \text{CB}(-15\,\text{dB})(\text{corr} 5), \text{SNR}]$.

We display the confusion matrix for four parameters and three parameters in figures 7(b) and 7(c). With the same percentage, all combinations do the same: 1/3 coding rate was classified at 100%, but 1/2 code rate had some false alarms (1.7%).

**B. THE RULE-BASED STRATEGY FOR CODING RATE CLASSIFICATION**

Here, we conducted 3D and 4D rule-based graphical analysis with the selected parameters found in the previous section for coding rate. We generated 3D and 4D scatter plots with the same combinations of parameters for visual perception of the 1/2 and 1/3 coding rate data distributions (see Figure 8). Figure 8(a) shows QPSK data with a smaller SNR; all data for 1/2 and 1/3 code rates are spread over the whole RMS Delay Spread and MED parameter ranges. The MCS level at a lower value for RMS also laid down a lower MED value. The SNR value of approximately 2-10 dB covers all MCS levels. For the SNR range of 20-25, a 1/2 coding rate MCS level was found. In Figure 8(b), all MCS levels of QPSK data are in the lower SNR region of 2-10 dB, the 0-400 range for Coherence BW, and distributed over almost the whole range of MED values. A few 1/2 MCS levels were at the higher SNR value of 20 dB, and a few 1/3 MCS levels were at a higher value of 600-1400 with CB. Figure 8(c) shows the combined effect of all four features on coding rate classification. Under communication theory, Coherence BW is inversely proportional to RMS Delay Spread. This figure proves that.

For the highest CB value of 1521, RMS resembles the lowest value. Again, with the lower value for CB, RMS showed the highest value of 15.46. CB had the smallest value: 12.93. SNR has no direct relationship with CB and showed a random
TABLE 10. Accuracy Using Threshold Values of Individual Parameters for Coding Rate Classification.

| Modulation | Parameter          | 1/2 |       | 3/4 |       | Overall Accuracy |
|------------|------------------|-----|-------|-----|-------|-----------------|
|            |                  | 1/2 |       | 3/4 |       |                 |
| QPSK       | SNR              | SNR < 7.5 | 57% | SNR > 7.5 | 65.57% | 61.11%           |
|            | MED              | MED > 24 | 48.33% | MED > 24 | 60.86% | 54.34%           |
|            | Coherence BW     | CB < 180 | 92.33% | CB > 180 | 11.23% | 53.47%           |
|            | RMS Delay Spread | RMS < 6.26 | 56.33% | RMS > 6.26 | 35.86% | 46.53%           |
|            | SNR              | SNR < 6.43 | 47.27% | SNR > 6.43 | 50.82% | 49%              |
|            | MED              | MED > 25 | 49.45% | MED < 25 | 55.79% | 52.54%           |
|            | Coherence BW     | CB < 55 | 88.36% | CB > 180 | 86.75% | 87.57%           |
|            | RMS Delay Spread | RMS > 6.38 | 41.09% | RMS > 6.26 | 59.49% | 50.06%           |
|            | SNR              | SNR > 5 | 54.18% | SNR < 5 | 55.07% | 54.63%           |
|            | MED              | MED > 35.5 | 48.72% | MED > 35.5 | 56.52% | 52.63%           |
|            | Coherence BW     | CB > 41.5 | 64.72% | CB < 41.5 | 51.08% | 58.07%           |
|            | RMS Delay Spread | RMS < 6.26 | 56.33% | RMS > 6.26 | 35.86% | 46.53%           |

relationship with SNR. In the figure, SNR is the fourth coordinate. The highest SNR value of 10.743 showed a very much lower value for RMS Delay Spread: 1.78. Another value of 10.31 for SNR showed an RMS value of 10.74. We could say that sometimes a higher SNR value shows a comparatively high value for RMS, and sometimes shows a relationship with a low RMS value. Besides, the lowest SNR value shows a comparatively higher value for RMS. So, SNR also has a random relationship with RMS Delay Spread. In communication theory, SNR and MED have no direct relationship, but RMS Delay Spread and MED showed a similar trend. This fact is also reflected in this figure. The lowest value for SNR relates to the highest value for MED, but in the whole distribution, the relationship shows a lot of randomness, not linearity. In addition, 16QAM and 64QAM show a pattern similar in data distribution to a coding rate like QPSK. One remarkable point is that two types of coding rate overlap too much. Finding a natural threshold is a challenging job here! We will do every single-parameter analysis numerically with a rule-based strategy for coding rate classification, as was done for modulation classification. We will also do a multi-dimensional RB analysis.

1) 1D ANALYSIS

[QPSK]SNR: Because both code rates have mean values and distribution approximately similar to the 1/2 code rate, and the distributions of 3/4 code rates overlap each other, we can select any mean value, or the average of those, as threshold points to classify. Within the [−σ, σ] interval, the 1/2 code rate has a lower percentage of data coverage. So, the trial threshold was chosen at 7.4 (the mean of the 1/2 code rate). It increased the percentage of data in the 1/2 coding rate portion. If another trial threshold is 7.67, which is the mean of the 1/3 code rate, it decreased the percentage of classifications for this code rate. Then finally, SNR’s threshold value was chosen (at 7.5), which has an optimum accuracy percentage for both coding rates, and a maximum overall accuracy of 61.11%.

RMS Delay Spread (−15 dB), MED (−15 dB), CB (−15 dB)(cor 5): Based on the logic in the maximum percentage of overall accuracy obtained with a specific threshold value, we held several trials for each parameter surrounding their mean values, and primarily selected threshold values are summarized in Table 10.

2) MULTI-DIMENSIONAL ANALYSIS FOR MODULATION AND CODING RATE CLASSIFICATION

With ML algorithm analysis, we achieved 98.26% with BRT algorithm in case of QPSK with three-parameter [MED, CB, SNR] combination at best during coding rate classification. With two-parameter combination of [SNR, CB], 16QAM data performed mostly of 99.6% in the same classification.

As modulation classification, if we follow the [−σ, σ] interval for multi-dimensional coding rate classification in case of Rule-based analysis, we do not obtain better accuracy. With QPSK for the MED (−15 dB), CB (−15 dB)(cor 5), SNR combination, the accuracy was 56.67% (1/2), 70.65% (1/3), and 63.36% (overall). For the MED (−15 dB), RMS Delay Spread (−15 dB), SNR combination, the accuracy was 65% (1/2), 69.2% (1/3), and 67.01% (overall). However, two types of coding rate data overlap here. If we use the hard limit threshold values (no overlapping data), we have an overall accuracy of 23.26% and 24.82% corresponding to the above two sets of three-parameter combinations. After that, we expand the interval to [−1.2σ, 1.2σ], and we held a trial for different sets of three-parameter combinations. We achieved a reasonable overall accuracy: 75.69% for MED, CB, SNR, and 78.64% for MED, CB, RMS. By repeating a similar process with 16QAM and 64QAM, 75.53% and 73.50% overall accuracies, respectively, were achieved for MED, CB, SNR, and 78.64% and 76.81%, respectively, were achieved for MED, CB, RMS. Table 11 presents all these findings.

With the same interval, [−1.2σ, 1.2σ], we also verified several two-parameter combinations, which showed...
TABLE 11. Accuracy Measure of Modulation and Coding Rate Classifications with Two, Three, Four, and Eleven Parameters.

| Parameters | Modulation Classification | By ML | By Rule-based |
|------------|----------------------------|-------|---------------|
|            | Analytical Method | Percentage | Analytical Method | Percentage |
| SNR, CB    | KNN | 96.83% | 2D | 85.45% |
| SNR, MED   | KNN | 95.74% | 2D | 76.78% |
| SNR, RMS   | KNN | 95.45% | 2D | 72.62% |
| MED, CB, SNR | KNN | 96.83% | 3D | 74.67% |
| MED, CB, RMS | BRT | 96.83% | 4D |               |
| MED, RMS, CB, DS, FS, SNR | BRT | 99.90% | 11D |               |

| Parameters | Coding Rate Classification | By ML | By Rule-based |
|------------|----------------------------|-------|---------------|
|            | Analytical Method | Percentage | Analytical Method | Percentage |
| SNR, CB    | QPSK | 95.4% | 2D | 87.15% |
|            | 16QAM | 99.6% | 86.06% |
|            | 64QAM | 94.5% | 83.03% |
| SNR, MED   | QPSK | 94.2% | 2D | 76.22% |
|            | 16QAM | 97.2% | 73.97% |
|            | 64QAM | 94.5% | 77.17% |
| SNR, RMS   | QPSK | 97.1% | 2D | 77.6% |
|            | 16QAM | 99% | 78.25% |
|            | 64QAM | 94.5% | 74.34% |
| MED, CB, SNR | QPSK | 98.26% | 3D | 75.69% |
|            | 16QAM | 96.28% | 75.55% |
|            | 64QAM | 95.75% | 73.5% |
| MED, CB, RMS | QPSK | 96.13% | 3D | 78.64% |
|            | 16QAM | 96.13% | 78.64% |
|            | 64QAM | 94.54% | 76.81% |
| MED, RMS, CB, DS, FS, SNR | QPSK | 100% |               |               |
|            | 16QAM | 100% |               |               |
|            | 64QAM | 100% |               |               |

Performance of up to 87.15%. Among three two-parameter combinations of [SNR, CB], [SNR, MED], and [SNR, RMS]; [SNR, CB] produced maximum performance for all three types of modulation: QPSK, 16QAM, and 64QAM.

At the final stage, when finalizing the parameters, we verified the performance of all three combinations of two parameters, both combinations of three parameters obtained from coding rate classification, along with the available four parameters and 11 parameters at the receiving end, for modulation classification with ML algorithms and rule-based strategies. The results were as sound as the previous ML analysis of modulation classifications. The best accuracy of 96.83% was achieved for both two-parameter [SNR, CB] and three-parameter [MED, CB, SNR] combinations with the KNN algorithm, and with four parameters for the BRT algorithm. Although 11 parameters showed 99.9% competence with the BRT algorithm, we should remember the real-time system implementation constraint. All 2D, 3D, 4D, and 11-dimensional results for both modulation and coding rate classification are summarized in Table 11.

From the confusion matrix in Figure 10 for $MED(-15\, dB)$, $CB(-15\, dB)(corr\, 5)$. SNR parameters, observe that 16QAM has comparatively more false alarms at 4.6%, and QPSK and 64QAM had more accurate classification at 96.4%.

Finally, we need to establish the thresholds of every selected parameter for adjusting the performance percentage. For the Incheon data, the combination of $CB(-15\, dB)$ (corr 5), $SNR, MED(-15dB)$ was primarily chosen as the optimum three-parameter combination from among all of them. We maximized or minimized the threshold values in a way (for both classifications) to get the best result. The main objective is to minimize the overlap among all types of modulation and coding rate labels. The results with finalized threshold values are shown in Table 12 (parts I and II).
The scatter plots of these parameters for QPSK modulation data are in Figure 11, where green dots show the 1/2 coding rate data value after setting the interval or the thresholding, and red dots show the 1/3 coding rate data distribution after setting the calculation or the thresholding.

VII. TAEAN DATA ANALYSIS

For comparison purposes, we checked the results from the Incheon dataset analysis against the Taean dataset. In the Taean dataset, there are two types of experimental data. One is with 1 km between the Rx and Tx, and the other is with 3 km. The detailed dataset overview was described in [12], and the dataset is available from [13]. In the previous analysis of the Incheon data, the rule-based strategy showed comparatively poor performance, so in this case, we only analyzed the dataset with ML algorithms. There are 11 parameters for the different combinations. Here, we only verify the combinations of two and three parameters that are available at the receiver. There are four types of modulation in this dataset: BPSK, QPSK, 16QAM, and 64QAM. For channel coding, only Turbo code is used, and as before, there are the same two types of coding rate: 1/2 and 1/3.

A. THE MACHINE LEARNING APPROACH TO MODULATION CLASSIFICATION

In all cases, KNN performed impressively from among all the ML algorithms for modulation classification. In two-parameter combinations for the 1 km distance, [SNR, CB] and [SNR, RMS] showed equal competence for link adaptation, and classification accuracy was 90.4%. In three-parameter combinations, [MED, CB, RMS] and [MED, CB, SNR] were compatible, with the first combination achieving accuracy of 95.2%. In the 3 km trial dataset, [SNR, CB] and [SNR, MED] performed better, with the latter showing 97.2% accuracy. And both [MED, CB, RMS] and [MED, RMS, SNR] achieved 94.44% accuracy.

B. THE MACHINE LEARNING APPROACH TO CODING RATE CLASSIFICATION

In coding rate classification, a similar set of parameters for both two- and three-parameter scenarios showed eligibility. For the two-parameter combinations of [SNR, CB] and [SNR, RMS] for BPSK and 16QAM modulations, classification accuracy was 100%. For [MED, RMS, CB] and [SNR, MED, CB] scenarios, at least three of the four modulations attained 100% classification accuracy. For the 3 km–trial dataset, [SNR, CB], [SNR, RMS], and [SNR, MED] showed similar performance. In all combinations, for BPSK and 64QAM modulation, performance was 100%. However, for QPSK and 16QAM, classification accuracy was not so high, but still acceptable, at about 89%. On the other hand, among the three-parameter combinations of [MED, CB, RMS], [SNR, CB, RMS], and [SNR, MED, RMS], the first showed the best performance. With this combination, for BPSK, 16QAM, and 64QAM modulations, the classification accuracy was...
### TABLE 12. Thresholds for Three Parameters from the Incheon Dataset.

| Modulations | Parameters | Rule-based Strategy | Percentage of Accuracy | Overall Accuracy |
|-------------|------------|---------------------|------------------------|-----------------|
| QPSK        | CB         | CB > 70             |                        | 15.90%          |
|             | SNR        | SNR > 7.6           |                        |                 |
|             | MED        | MED < 15            |                        |                 |
| 16QAM       | SNR        | 5 < SNR < 7.6       | 6.97%                  | 16.95%          |
|             | MED        | 15 < MED < 29       |                        |                 |
| 64QAM       | CB         | CB < 40             |                        | 6.97%           |
|             | SNR        | SNR < 5             |                        |                 |
|             | MED        | MED > 29            |                        |                 |

| Modulations | Parameters | Rule-based Strategy | Percentage of Accuracy | Overall Accuracy |
|-------------|------------|---------------------|------------------------|-----------------|
|            | Code Rate  |                     |                        |                 |
| 1/2 QPSK   | CB         | < 50                | 31.67%                 | 33.51%          |
|            | SNR        | < 7.5               |                        |                 |
|            | MED        | > 19                |                        |                 |
| 1/3 QPSK   | SNR        | > 7.5               | 35.51%                 |                 |
|            | MED        | < 19                |                        |                 |
| 1/2 16QAM  | SNR        | < 6.43              | 24.73%                 | 22.67%          |
|            | MED        | > 19                |                        |                 |
|            | CB         | > 47                |                        |                 |
| 1/3 16QAM  | SNR        | > 6.43              | 20.51%                 |                 |
|            | MED        | < 19                |                        |                 |
|            | CB         | > 32                |                        |                 |
| 1/2 64QAM  | SNR        | > 4                 | 30.54%                 | 29.58%          |
|            | MED        | < 24                |                        |                 |
|            | CB         | > 24                |                        |                 |
| 1/3 64QAM  | SNR        | < 4                 | 28.62%                 |                 |
|            | MED        | < 32                |                        |                 |

All results are summarized in Table 1 of the Appendix. Figure 12 shows the confusion matrix for modulation and coding rate classifications of the Taean dataset.

If we compare these findings with the Incheon dataset findings, then we see that [MED, CB, SNR] is a common combination for both the Incheon dataset and the Taean 1 km dataset, showing the highest accuracy of 98.26% in coding rate classification of the Incheon dataset. However, this combination did not show such competence with the Taean 3 km dataset. Moreover, [MED, CB, RMS] was a better performer for both Taean datasets, at 100% accuracy in coding rate classification with KNN for three types of modulation.

As well, using the Incheon dataset obtained 96.13% accuracy in coding rate classification for two types of modulation with the BRT algorithm. For two-parameter combinations, [SNR, CB] showed unparalleled performance with all three datasets. In some cases, the performance reached 100% with [SNR, CB]. Although [SNR, MED] showed superior results for modulation classification of the Taean 3 km dataset, [SNR, RMS] showed mostly equal results with both types of Taean datasets.

Finally, we could choose [SNR, CB] for a two-parameter combination, and [MED, CB, RMS] for a three-parameter combination to obtain the optimal fit with all three datasets in a real-time, energy-saving link adaptation implementation.
of a UACN. However, [MED, CB, RMS] performs in a lesser extent of 95.5% with the Incheon dataset, very near to highest performance of 96.83%.

C. PERFORMANCE COMPARISON AMONG DIFFERENT MACHINE LEARNING APPROACHES

If we compare the performance summary in Table 4 from our previous work [12] with Table 11 for the Incheon dataset and tables 13 and 14 in the Appendix for the Taean datasets, we see that Boosted Regression Tree best performed with 18 features of the Incheon dataset [12] at 99.97%; then, SVM with the default kernel was the second highest performing algorithm. The Pseudolinear Discriminant algorithm came in just behind the SVM algorithm. None of the algorithms is good for the Taean datasets, even with a full set of 11 features.

On the other hand, in the present scenario, after PCA analysis, we get a better choice of feature selections. For the Taean datasets, we also achieved 100% accuracy in coding rate classification with several modulation types. For 4D and higher parameter combinations in the Incheon dataset, the Boosted Regression Tree algorithm showed better performance at 99.9%. With higher dimensions, the SVM with a Gaussian kernel also showed competence. Sometimes, Naïve Bayes and pseudolinear algorithms showed some competence, but this is not usually the case, and they had not so high a performance classifying the modulation and coding rate, achieving only marginal percentages. We discarded those results, either due to high dimensionality or due to the marginal results. However, this research reflects the highest potentiality of the K-nearest Neighbor algorithm in most of the 2D and 3D cases. With the Taean datasets, KNN performed best for all 2D and 3D combinations of parameters. This is also true for the Incheon dataset with a lower dimensional analysis. We can conclude that, with just a few parameters, KNN is the best option for Underwater Acoustic Communications Networks.

VIII. CONCLUSION

In this study, primarily the Incheon sea dataset in a 2 km–distance trial was thoroughly analyzed for link adaptation of underwater communication networks where uplink data are measured by using an acoustic signal at the UBSC. After that, the Taean sea trial datasets were compared to verify the findings. First, PCA was used to minimize the data volume for both datasets. Both Machine Learning algorithms and the rule-based strategy were used to classify modulation along with coding rate. Here, the accuracy of individual classifications of every transmitted modulation and coding rate type have been measured. Also, the minimum number of two- and three-parameter combinations were chosen for real-time implementation due to energy constraints at the receiver. Those chosen parameters were compared with Taean dataset parameters at both 1 km and 3 km distances. Finally, the set of MED(−15 dB), CB(−15 dB)(corr 5), RMS Delay Spread, was selected for the three-parameter set, and SNR, CB(−15 dB)(corr 5) was selected for the two-parameter set. For two parameters, a 96.83% overall accuracy was found for modulation classification, and 100% accuracy was found for coding rate classification. For the Incheon dataset analysis, BRT and KNN performed better than all other ML algorithms. For the Taean dataset analysis, KNN performed distinctively. With three parameters, coding rate classification achieved 100% accuracy with the KNN algorithm for both Taean datasets. Using two parameters in RB achieved a maximum 87.15% in coding rate classification accuracy. With [−1.2σ, 1.2σ], approximately 78.64% accuracy was found in coding rate classification with the above-mentioned three parameters. But with a hard limit on threshold values, poor performance reached 35.51% as a maximum for both classifications of the 1/3 code rate (Turbo code) with QPSK modulation, which showed a lot less reliability in the RB analysis. This research obtained similar performance with a four-parameter set, and 100% with 11 parameters, but those results were discarded due to the reality of energy constraints. Finally, both PCA and ML analyses showed the data independence of Doppler Spread and Frequency Shift on AMC, which mitigates a major challenge for high Doppler Spread due to fast fading of channels in UAC networks.

APPENDIX A

TERMINOLOGIES

In this article, the following terms are used to measure the modulation and coding rate classification accuracy in both rule-based analysis and Machine Learning–based analysis.

a. Percentage of Data Coverage: This term is used in rule-based analysis. It describes the truncated data percentage for any sigma interval. It contains overlapped data for different modulation labels.

- Formula for percentage of data coverage of any feature within the ±ks interval, where k = ±1, ±1.5, and ±2:

For modulation, (4), as shown at the bottom of the next page.
Two other definitions are also the same.
For coding rate, (5), as shown at the bottom of the next page.
The other definition is the same.

b. Percentage for Classification Accuracy: This term is used in both rule-based and ML-based approaches. It describes the data percentage considering threshold values within different modulations or coding rates. So, there are no overlapping data. It resembles the true detection of modulation labels or coding rates.

For the rule-based strategy.

- Formula for percentage of modulation classification accuracy using any feature within the ±ks interval: ±1σ, ±1.2σ, ±1.5σ, or ±2σ (6), as shown at the bottom of page 21.

- Formula for percentage of coding rate classification accuracy using any feature within the ±ks
TABLE 13. Parameters Found in PCA Analysis (Taean Dataset).

| Parameters with Positive Covariance | Eigenvalue | Parameters with Negative Covariance | Eigenvalue | PCs   | Individual Variance |
|-------------------------------------|------------|-------------------------------------|------------|-------|---------------------|
| Uncoded BER (Linear)               | 0.35311    | CB(-10 dB)(0.5 corr)                | -0.34457   | PC1   | 9190.035            |
| Uncoded BER (DFT)                  | 0.35303    | Pilot Time (T)                      | -0.27931   | PC2   | 5172.14             |
| Coded BER (DFT)                    | 0.35246    | Pilot Space (F)                     | -0.27931   | PC3   | 5134.966            |
| Coded BER (Linear)                 | 0.35131    | CB(-15 dB)(0.5 corr)                | -0.27799   | PC4   | 7459.23             |
| RMS Delay Spread (-10 dB)          | 0.37948    | Data Rate (bps)                     | -0.2284    | PC5   | 165                 |
| MED (-10 dB)                       | 0.36263    | RMS Delay Spread (-10 dB)           | -0.34671   | PC6   | 2631.8              |
| RMS Delay Spread (-15 dB)          | 0.31381    | SNR                                 | -0.63872   |       |                     |
| MED (-15 dB)                       | 0.55559    | RP(Space)(F)                        | -0.35129   |       |                     |
| Pilot (Time)(T)                    | 0.50312    | EVM                                 | -0.57329   |       |                     |
| Pilot (Space) (F)                  | 0.50312    |                                     |            |       |                     |
| EVM                                 | 0.44421    |                                     |            |       |                     |
| CB (-10 dB)(0.5 corr)              | 0.33909    |                                     |            |       |                     |
| RP (Time)(T)                       | 0.65301    |                                     |            |       |                     |
| RP (Space)(F)                      | 0.72854    |                                     |            |       |                     |
| SNR                                 | 0.45248    |                                     |            |       |                     |
| CB (-15 dB) (0.5 Corr)             | 0.83922    |                                     |            |       |                     |
| Data Rate (bps)                    | 0.60853    |                                     |            |       |                     |

interval: ±1σ, ±1.2σ, ±1.5σ, or ±2σ (7), as shown at the bottom of page 21.

For the ML approach, it measures how many test data could be correctly classified from the given test data for each modulation by using information about the training data. The formula is in equation (6), but the terminologies are as follows:

\[
n'_{QPSK} = \text{no. of correctly classified test QPSK data} \]
\[
n_{QPSK} = \text{no. of given test QPSK data} \] (8)

The other two definitions are the same. For coding rate classification, we can also use equation (7) as follows:

\[
n'_{\text{half_codingrate}} = \frac{\text{no. of correctly classified test half_codingrate data}}{n_{\text{half_codingrate}}} \]
\[
n_{\text{half_codingrate}} = \text{no. of given test half_coding rate data} \]

The other definition is the same.

\[
QPSK_{\text{data coverage}} = \frac{\text{no. of feature values of QPSK within the interval } [-k\sigma, k\sigma]}{\text{Total no. of feature values of QPSK}} \times 100
\]
\[
n'_{\text{data, QPSK}} = \text{no. of feature values of QPSK within the interval } [-k\sigma, k\sigma]
\]
\[
n_{\text{QPSK}} = \text{Total no. of feature values of QPSK} \] (4)

\[
\text{half_codingrate}_{\text{data coverage}} = \frac{\text{no. of feature values of half_codingrate within the interval } [-k\sigma, k\sigma]}{\text{Total no. of feature values of half_codingrate}} \times 100
\]
\[
n'_{\text{data, half_codingrate}} = \text{no. of feature values of half_codingrate within the interval } [-k\sigma, k\sigma]
\]
\[
n_{\text{half_codingrate}} = \text{Total no. of feature values of half_codingrate} \] (5)
### TABLE 14. Accuracy Measure of Modulation and Coding Rate Classifications with Two or Three Parameters.

#### I. TAEAN DATASET, 1 KM

| Parameters   | Modulation Classification | Percentage | Coding Rate Classification | Percentage |
|--------------|----------------------------|------------|----------------------------|------------|
| **SNR, CB**  | KNN                        | 90.40%     | BPSK                       | KNN        | 100%       |
| **SNR, RMS** | KNN                        | 90.40%     | QPSK                       | KNN        | 97%        |
| **MED, CB, RMS** | KNN               | 95.20%     | 16QAM                      | KNN        | 100%       |
| **MED, CB, SNR** | KNN              | 90.40%     | 64QAM                      | KNN        | 91.43%     |
| **SNR, RMS**  | QPSK                       |            | 16QAM                      | KNN        | 96.67%     |
| **MED, CB, RMS** | KNN                  |            | 64QAM                      | KNN        | 91.43%     |
| **MED, CB, SNR** | BPSK                 |            | QPSK                       | KNN        | 100%       |
|               |                           |            | 16QAM                      | KNN        | 100%       |
|               |                           |            | 64QAM                      | KNN        | 91.43%     |
|               |                           |            | BPSK                       | KNN        | 100%       |
|               |                           |            | QPSK                       | KNN        | 96.67%     |
|               |                           |            | 16QAM                      | KNN        | 100%       |
|               |                           |            | 64QAM                      | KNN        | 100%       |

#### II. TAEAN DATASET, 3 KM

| Parameters   | Modulation Classification | Percentage | Coding Rate Classification | Percentage |
|--------------|----------------------------|------------|----------------------------|------------|
| **SNR,CB**   | KNN                        | 94.40%     | BPSK                       | KNN        | 100%       |
| **SNR, MED** | KNN                        | 97.20%     | QPSK                       | KNN        | 89%        |
| **MED, CB, RMS** | KNN                   | 94.44%     | 16QAM                      | KNN        | 89%        |
| **MED, RMS, SNR** | KNN                 | 94.44%     | 64QAM                      | KNN        | 100%       |
| **SNR, RMS** | QPSK                       |            | BPSK                       | KNN        | 100%       |
| **SNR, MED** | QPSK                       |            | 16QAM                      | KNN        | 89%        |
|               |                           |            | 64QAM                      | KNN        | 89%        |
|               |                           |            | BPSK                       | KNN        | 100%       |
|               |                           |            | QPSK                       | KNN        | 89%        |
|               |                           |            | 16QAM                      | KNN        | 89%        |
|               |                           |            | 64QAM                      | KNN        | 100%       |

226480
TABLE 14. (Continued.) Accuracy Measure of Modulation and Coding Rate Classifications with Two or Three Parameters.

| Method                  | Modulation | Coding Rate | Accuracy |
|-------------------------|------------|-------------|----------|
| MED, CB, RMS            | BPSK       | KNN         | 100%     |
|                         | QPSK       | KNN         | 89%      |
|                         | 16QAM      | KNN         | 100%     |
|                         | 64QAM      | KNN         | 100%     |
| SNR, CB, RMS            | BPSK       | KNN         | 100%     |
|                         | QPSK       | KNN         | 89%      |
|                         | 16QAM      | KNN         | 89%      |
|                         | 64QAM      | KNN         | 100%     |
| SNR, MED, RMS           | QPSK       | KNN         | 89%      |
|                         | 16QAM      | KNN         | 89%      |
|                         | 64QAM      | KNN         | 100%     |

**c. Overall Accuracy (for modulation):** This represents the percentage of total rational data coverage or the total rational level of detection.

- **Formula for Overall Accuracy:** For rule-based analysis,

\[
\text{Overall Accuracy} = \frac{n'_{\text{QPSK}} + n'_{\text{16QAM}} + n'_{\text{64QAM}}}{n_{\text{QPSK}} + n_{16\text{QAM}} + n_{64\text{QAM}}} \times 100
\]  

For Machine Learning approaches, it measures the ratio of total correctly classified data for each modulation from the given test data and the total data size, including all training and test data. The formula is as seen in (9), but the terminologies are as follows:

\[
n'_{\text{QPSK}} = \text{no. of correctly classified test QPSK data} \\
n_{\text{QPSK}} = \text{Total no. of QPSK data}
\]

**QPSK Classification Accuracy**:

\[
\text{QPSK classification accuracy} = \frac{\text{no. of feature values of QPSK < } \text{Threshold}_1}{\text{Total no. of feature values of QPSK}} \times 100
\]

\[
= \frac{n'_{\text{QPSK}}}{n_{\text{QPSK}}} \times 100
\]

**16QAM Classification Accuracy**:

\[
\text{16QAM classification accuracy} = \frac{\text{no. of feature values of 16QAM < } \text{Threshold}_2}{\text{Total no. of feature values of 16QAM}} \times 100
\]

\[
= \frac{n'_{\text{16QAM}}}{n_{16\text{QAM}}} \times 100
\]

**64QAM Classification Accuracy**:

\[
\text{64QAM classification accuracy} = \frac{\text{no. of feature values of 64QAM < } \text{Threshold}_2}{\text{Total no. of feature values of 64QAM}} \times 100
\]

\[
= \frac{n'_{\text{64QAM}}}{n_{64\text{QAM}}} \times 100
\]

**Half Coding Rate Classification Accuracy**:

\[
\text{half\_codingrate classification accuracy} = \frac{\text{no. of feature values of half\_codingrate < } \text{Threshold}_1}{\text{Total no. of feature values of half\_codingrate}} \times 100
\]

\[
= \frac{n'_{\text{half\_codingrate}}}{n_{\text{half\_codingrate}}} \times 100
\]

**One-Third Coding Rate Classification Accuracy**:

\[
\text{onethird\_codingrate classification accuracy} = \frac{\text{no. of feature values of onethird\_codingrate < } \text{Threshold}_1}{\text{Total no. of feature values of onethird\_codingrate}} \times 100
\]

\[
= \frac{n'_{\text{onethird\_codingrate}}}{n_{\text{onethird\_codingrate}}} \times 100
\]

VOLUME 8, 2020
Two other definitions are the same.

Overall Accuracy (for coding rate): This represents the percentage of total rational data coverage or the total rational level of detection.

- Formula for Overall Accuracy:

$$\text{Overall Accuracy} = \frac{n'_\text{half\_codingrate} + n'_\text{nonthird\_codingrate}}{n\text{\_codingrate} + n\text{\_nonthird\_codingrate}} \times 100$$  \hspace{1cm} (11)

For an ML approach, it measures the ratio of total correctly classified data for each coding rate from the given test data and the total data size, including all training and test data.

The formula is as seen in (11), but the terminologies are as follows:

- \(n'_\text{half\_codingrate}\): no. of correctly classified test half\_codingratedata
- \(n'_\text{half\_codingrate}\): no. of half\_codingratedata

$$\text{Overall Accuracy} = \frac{n'_\text{half\_codingrate}}{\text{Total no. of half\_codingratedata}}$$  \hspace{1cm} (12)

REFERENCES

[1] F. Qu, Z. Wang, L. Yang, and Z. Wu, “A journey toward modeling and resolving Doppler in underwater acoustic communications,” IEEE Commun. Mag., vol. 54, no. 2, pp. 49–55, Feb. 2016, doi: 10.1109/MCOM.2016.7470260.

[2] N. Parrish, L. Tracy, S. Roy, P. Arabshahi, and W. Fox, “System design considerations for undersea networks: Link and multiple access protocols,” IEEE J. Sel. Areas Commun., vol. 26, no. 9, pp. 1720–1730, Dec. 2008, doi: 10.1109/JSAC.2008.081211.

[3] F. X. Sohleufer, J. M. Passerieux, and C. Laut, “Characterization of time-varying underwater acoustic communication channel with application to channel capacity,” in Underwater Acoustic Measurements, Version 1. Nafplion, Greece, 2009.

[4] S.-U. Kim, H.-S. Cheon, S.-B. Seo, S.-M. Song, and S.-Y. Park, “A hexagon tessellation approach for the transmission energy efficiency in underwater wireless sensor networks,” J. Inf. Process. Syst., vol. 6, no. 1, pp. 53–66, Mar. 2010, doi: 10.3745/JIPS.2010.6.1.053.

[5] M. Stojanovic, “On the design of underwater acoustic cellular systems,” in Proc. Oceans Eur., Jun. 2007, pp. 1–6, doi: 10.1109/OCEANSE.2007.4302226.

[6] B. Srinivasan and V. Rodoplu, “Capacity of underwater acoustic OFDM cellular networks,” in Proc. Oceans IEEE Sydney, May 2010, pp. 1–10, doi: 10.1109/OCEANSSYD.2010.5603911.

[7] I. Ahmad and K. Chang, “Effective SNR mapping and link adaptation strategy for next-generation underwater acoustic communications networks: A cross-layer approach,” IEEE Access, vol. 7, pp. 44150–44164, 2019, doi: 10.1109/ACCESS.2019.2908018.

[8] K. Pelekanakis and L. Cazzanti, “On adaptive modulation for low SNR underwater acoustic communications,” in Proc. OCEANS MTS/IEEE Charleston, Oct. 2018, pp. 1–6, doi: 10.1109/OCEANS.2018.8604521.

[9] M. Sadeghi, M. Elamassie, and M. Uysal, “Adaptive OFDM-based acoustic underwater transmission: System design and experimental verification,” in Proc. IEEE Int. Conf. Underwater Mater. Learn. Signal Process., Aug. 2010, pp. 391–396.

[10] Y. Wang, H. Zhang, Z. Sang, L. Xu, C. Cao, and T. A. Gulliver, “Modulation classification of underwater communication with deep learning network,” Comput. Intell. Neurosci., vol. 2019, pp. 1–12, 2019, Art. no. 8039632, doi: 10.1155/2019/8039632.

[11] Z. Zhu, M. Waqar Aslam, and A. K. Nandi, “Augmented genetic program¬ming for automatic digital modulation classification,” in Proc. IEEE Int. Workshop Mach. Learn. Signal Process., Aug. 2010, pp. 391–396.

[12] M. S. M. Alamgir, M. N. Sultana, and K. Chang, “Link adaptation on an underwater communications network using machine learning algorithms: Boosted regression tree approach,” IEEE Access, vol. 8, pp. 73957–73971, 2020, doi: 10.1109/ACCESS.2020.2981973.
[36] M. Doniec, M. Angermann, and D. Rus, “An end-to-end signal strength model for underwater optical communications,” *IEEE J. Ocean. Eng.*, vol. 38, no. 4, pp. 743–757, Oct. 2013, doi: 10.1109/JOE.2013.2278952.

[37] S. Tang, Y. Dong, and X. Zhang, “Impulse response modeling for underwater wireless optical communication links,” *IEEE Trans. Commun.*, vol. 62, no. 1, pp. 226–234, Jan. 2014, doi: 10.1109/TCOMM.2013.120713.130199.

[38] B. Gulbahar and O. B. Akan, “A communication theoretical modeling and analysis of underwater magneto-inductive wireless channels,” *IEEE Trans. Wireless Commun.*, vol. 11, no. 9, pp. 3326–3334, Sep. 2012, doi: 10.1109/TWC.2012.070912.111943.

[39] Y. Freund, and R. E. Schapire, “Experiments with a new boosting algorithm machine learning,” in *Proc. 13th Int. Conf.*, 1996, pp. 146–148.

[40] T. Kanamori and T. Takenouchi, “Improving logitboost with prior knowledge,” *Inf. Fusion*, vol. 14, no. 2, pp. 208–219, Apr. 2013.

[41] Y. Freund, R. E. Schapire, and N. Abe, “A short introduction to boosting,” *J. Jpn. Soc. Artif. Intell.*, vol. 14, no. 5, pp. 771–780, 1999.

[42] B. Caputo, K. Sim, F. Furesjo, and A. Smola, “Appearance-based object recognition using SVMS: Which Kernel should i use,” in *Proc. NIPS Workshop Stat. Methods Comput. Exp. Visual Process. Comput. Vis.*, 2002. [Online]. Available: https://scholar.google.com/citations?user=mHbdIAwAAAAJ&hl=en

From 1989 to 1990, he was a Member of the Research Staff with the Samsung Advanced Institute of Technology (SAIT), where he was involved in digital signal processing system design. From 1992 to 2003, he was a Principal Member of the Technical Staff with the Electronics and Telecommunications Research Institute (ETRI), where he led the design teams involved in the WCDMA UE modem and 4G Radio Transmission Technology (RTT). He is currently with the Electrical and Computer Engineering Department, Inha University. His research interests include radio transmission technology in 3GPP LTE and 5G systems, public safety, mobile ad hoc networks (especially for UAV), cellular-V2X technology, maritime and underwater communications, applications of AI Technologies, non-terrestrial networks (NTNs), and network intelligence for 6G. He was a recipient of the LG Academic Award in 2006, the Haedong Best Paper Award in 2007, the IEEE ComSoc Best Paper Award in 2008, the Haedong Academic Award in 2010, the SKT SafeNet Best Idea Award in 2015, the KICS Outstanding Academic Society Award in 2018 and 2019, and the MSIT Minister’s Commendation and KICS Fellow in 2020. He is also the Chairman of the Expert Committee in SafeNet Forum and the Mobile and Automotive Convergence Committee in 5G Forum. He has also served as an Editor for ITU-R TSG1 IMT.MOD.

**KYUNGHI CHANG** (Senior Member, IEEE) received the B.S. and M.S. degrees in electronics engineering from Yonsei University, Seoul, South Korea, in 1985 and 1987, respectively, and the Ph.D. degree in electrical engineering from Texas A&M University, College Station, TX, USA, in 1992.

From 1989 to 1990, he was a Member of the Research Staff with the Samsung Advanced Institute of Technology (SAIT), where he was involved in digital signal processing system design. From 1992 to 2003, he was a Principal Member of the Technical Staff with the Electronics and Telecommunications Research Institute (ETRI), where he led the design teams involved in the WCDMA UE modem and 4G Radio Transmission Technology (RTT). He is currently with the Electrical and Computer Engineering Department, Inha University. His research interests include radio transmission technology in 3GPP LTE and 5G systems, public safety, mobile ad hoc networks (especially for UAV), cellular-V2X technology, maritime and underwater communications, applications of AI Technologies, non-terrestrial networks (NTNs), and network intelligence for 6G. He was a recipient of the LG Academic Award in 2006, the Haedong Best Paper Award in 2007, the IEEE ComSoc Best Paper Award in 2008, the Haedong Academic Award in 2010, the SKT SafeNet Best Idea Award in 2015, the KICS Outstanding Academic Society Award in 2018 and 2019, and the MSIT Minister’s Commendation and KICS Fellow in 2020. He is also the Chairman of the Expert Committee in SafeNet Forum and the Mobile and Automotive Convergence Committee in 5G Forum. He has also served as an Editor for ITU-R TSG1 IMT.MOD.

**MST. NAJINN SULTANA** (Student Member, IEEE) was born in Noakhali, Bangladesh, in 1978. She received the B.Sc. and M.Sc. degrees in applied physics and electronics from the University of Dhaka, Bangladesh, in 2003 and 2005, respectively. She is currently pursuing the Ph.D. degree in electronic engineering with the Mobile Telecommunication Research Laboratory (MTRL), Inha University, South Korea. From 2008 to 2012, she was a Research Assistant with the UWB-ITRC Laboratory, Inha University. She is currently an Assistant Professor (now on study leave) with the Computer Science and Engineering Department, Uuttara University, Bangladesh, starting in 2016. From 2014 to 2016, she was an Assistant Professor with the Electronics and Telecommunication Department, Daffodil International University, Bangladesh. She is the author of one book and eight articles and a coauthor of book chapter. Her research interests include deep learning for LTE, 5G link adaptation, MIMO beamforming, and massive MIMO.

Ms. Sultana is an Associate Member of SEEK. Her honors include the First Grade Comilla Board Scholarship, Bangladesh, for pursuing the B.Sc. degree, the Scholarship from the University of Dhaka, for pursuing the M.Sc. degree, and the Jungseok Memorial International Scholarship, South Korea, to pursue the Ph.D. degree due to her excellent academic career.

**KYUNGHI CHANG** (Senior Member, IEEE) received the B.S. and M.S. degrees in electronics engineering from Yonsei University, Seoul, South Korea, in 1985 and 1987, respectively, and the Ph.D. degree in electrical engineering from Texas A&M University, College Station, TX, USA, in 1992.

From 1989 to 1990, he was a Member of the Research Staff with the Samsung Advanced Institute of Technology (SAIT), where he was involved in digital signal processing system design. From 1992 to 2003, he was a Principal Member of the Technical Staff with the Electronics and Telecommunications Research Institute (ETRI), where he led the design teams involved in the WCDMA UE modem and 4G Radio Transmission Technology (RTT). He is currently with the Electrical and Computer Engineering Department, Inha University. His research interests include radio transmission technology in 3GPP LTE and 5G systems, public safety, mobile ad hoc networks (especially for UAV), cellular-V2X technology, maritime and underwater communications, applications of AI Technologies, non-terrestrial networks (NTNs), and network intelligence for 6G. He was a recipient of the LG Academic Award in 2006, the Haedong Best Paper Award in 2007, the IEEE ComSoc Best Paper Award in 2008, the Haedong Academic Award in 2010, the SKT SafeNet Best Idea Award in 2015, the KICS Outstanding Academic Society Award in 2018 and 2019, and the MSIT Minister’s Commendation and KICS Fellow in 2020. He is also the Chairman of the Expert Committee in SafeNet Forum and the Mobile and Automotive Convergence Committee in 5G Forum. He has also served as an Editor for ITU-R TSG1 IMT.MOD.

**MST. NAJINN SULTANA** (Student Member, IEEE) was born in Noakhali, Bangladesh, in 1978. She received the B.Sc. and M.Sc. degrees in applied physics and electronics from the University of Dhaka, Bangladesh, in 2003 and 2005, respectively. She is currently pursuing the Ph.D. degree in electronic engineering with the Mobile Telecommunication Research Laboratory (MTRL), Inha University, South Korea. From 2008 to 2012, she was a Research Assistant with the UWB-ITRC Laboratory, Inha University. She is currently an Assistant Professor (now on study leave) with the Computer Science and Engineering Department, Uuttara University, Bangladesh, starting in 2016. From 2014 to 2016, she was an Assistant Professor with the Electronics and Telecommunication Department, Daffodil International University, Bangladesh. She is the author of one book and eight articles and a coauthor of book chapter. Her research interests include deep learning for LTE, 5G link adaptation, MIMO beamforming, and massive MIMO.

Ms. Sultana is an Associate Member of SEEK. Her honors include the First Grade Comilla Board Scholarship, Bangladesh, for pursuing the B.Sc. degree, the Scholarship from the University of Dhaka, for pursuing the M.Sc. degree, and the Jungseok Memorial International Scholarship, South Korea, to pursue the Ph.D. degree due to her excellent academic career.

**KYUNGHI CHANG** (Senior Member, IEEE) received the B.S. and M.S. degrees in electronics engineering from Yonsei University, Seoul, South Korea, in 1985 and 1987, respectively, and the Ph.D. degree in electrical engineering from Texas A&M University, College Station, TX, USA, in 1992.

From 1989 to 1990, he was a Member of the Research Staff with the Samsung Advanced Institute of Technology (SAIT), where he was involved in digital signal processing system design. From 1992 to 2003, he was a Principal Member of the Technical Staff with the Electronics and Telecommunications Research Institute (ETRI), where he led the design teams involved in the WCDMA UE modem and 4G Radio Transmission Technology (RTT). He is currently with the Electrical and Computer Engineering Department, Inha University. His research interests include radio transmission technology in 3GPP LTE and 5G systems, public safety, mobile ad hoc networks (especially for UAV), cellular-V2X technology, maritime and underwater communications, applications of AI Technologies, non-terrestrial networks (NTNs), and network intelligence for 6G. He was a recipient of the LG Academic Award in 2006, the Haedong Best Paper Award in 2007, the IEEE ComSoc Best Paper Award in 2008, the Haedong Academic Award in 2010, the SKT SafeNet Best Idea Award in 2015, the KICS Outstanding Academic Society Award in 2018 and 2019, and the MSIT Minister’s Commendation and KICS Fellow in 2020. He is also the Chairman of the Expert Committee in SafeNet Forum and the Mobile and Automotive Convergence Committee in 5G Forum. He has also served as an Editor for ITU-R TSG1 IMT.MOD.