This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/OJCOMS.2021.3079643, IEEE Open Journal of the Communications Society

Data-driven Precoded MIMO Detection Robust to Channel Estimation Errors

Abderrahmane Mayouche, *Student Member, IEEE*, Wallace A. Martins, *Senior Member, IEEE*, Symeon Chatzinotas, *Senior Member, IEEE* and Björn Ottersten, *Fellow, IEEE*

***Abstract*—We study the problem of symbol detection in downlink coded multiple-input multiple-output (MIMO) systems with precoding and without the explicit knowledge of the channel-state information (CSI) at the receiver. In this context, we investigate the impact of imperfect CSI at the transmitter (CSIT) on the detection performance. We first model the CSIT degradation based on channel estimation errors to investigate its impact on the detection performance at the receiver. To mitigate the effect of CSIT deterioration at the latter, we propose learning-based techniques for hard and soft detection that use downlink precoded pilot symbols as training data. We note that these pilots are originally intended for signal-to-interference-plus-noise ratio (SINR) estimation. We validate the approach by proposing a lightweight implementation that is suitable for online training using several state-of-the-art classifiers. We compare the bit-error rate (BER) and the runtime complexity of the proposed approaches where we achieve superior detection performance in harsh channel conditions while maintaining low computational requirements. Specifically, numerical results show that severe CSIT degradation impedes the correct detection when a conventional detector is used. However, the proposed learning-based detectors can achieve good detection performance even under severe CSIT deterioration, and can yield 4-8 dB power gain for BER values lower than 10^{-4} when compared to the classic linear minimum mean square error (MMSE) detector.**

***Index Terms*—MIMO detection, precoding, machine learning, channel coding, and imperfect CSIT.**

I. Introduction

Multiple-input multiple-output (MIMO) technology can drastically improve the spectral efficiency (measured in bits/s/Hz) by using multiple antennas at the transmitter (Tx) and the receiver (Rx). MIMO technology has been widely deployed in modern communication systems, however, most receivers rely heavily on accurate channel state information (CSI) in order to detect the data symbols sent by each transmit antenna [1]. Particularly, in the MIMO spatial multiplexing (SM) configuration where several information streams are sent in the same time frequency resource, the Rx has to detect multiple interfering symbol streams transmitted concurrently through a channel subject to random noise and interference [2]. This configuration creates the inter-channel-interference (ICI) *phenomenon*, which constitutes a fundamental limiting component of MIMO communication [3].

Symbol detection in MIMO systems is a well studied problem with numerous classical detection methods [4]. More recently, several learning-based approaches have been reported for symbol detection in MIMO systems. Notably, machine learning (ML) has attracted significant interest in the area of wireless communication [5], [6], among which we find the work where deep learning is used to learn the end-to-end link from the Tx to the Rx [7], [8]; the idea is to design a communication system to jointly optimize transmitter and receiver components by interpreting it as an autoencoder.

In the case where the Rx has CSI knowledge (CSIR), several learning-based MIMO detectors have been proposed [9]–[13]. Specifically, the authors in [9] achieved excellent detection performance with a deep neural network architecture called DetNet, e.g., with a performance matching a semi-definite relaxation (SDR) baseline for independent and identically distributed (i.i.d.) Gaussian channels while running $30\times$ faster. The work in [10] introduced OAMPNet, a deep learning based scheme that mimics the orthogonal approximate message-passing (OAMP) algorithm [14], which outperforms the OAMP algorithm in both i.i.d and small-sized Kronecker model-based correlated channels. We note that DetNet and OAMNet are both trained offline: using a single detector trained over several channel matrices. On the other hand, MMNEt in [11] is an adaptive neural-network based detection scheme tailored to realistic channels with spatial correlation and is suitable for online training, where the training is performed for each coherence time instead of offline training where the module could be trained in advance for a large number of coherence times by generating random channel coefficients. In the same context of online training and motivated by practical implementation, EPNet [12] was proposed to perform signal detection by unfolding the expectation propagation (EP) algorithm and training the damping factors. To support coded systems, a neural-network MIMO detector with impairments was proposed in [13], where the detection algorithm design is based upon projected gradient descent iterations for MIMO-OFDM systems. Empirical results show the robustness of the proposed detection scheme against several common communication impairments, since the NN does not assume any specific model. We note that the bit-error rate (BER) is the main metric used for performance evaluation of the aforementioned work [9]–[13].

However, in systems where CSIR is not explicitly available, few learning-based approaches have been proposed [15]–[18]

The authors are with the Interdisciplinary Centre for Security Reliability and Trust (SnT), University of Luxembourg, L-1855, Luxembourg (E-mails: {abderrahmane.mayouche, wallace.alvesmartins, symeon.chatzinotas, and bjorn.ottersten}@uni.lu).

This work is supported by the Engineering and Physical Sciences Research Council (EPSRC) and the Luxembourg National Research Fund (FNR) project titled Exploiting Interference for Physical Layer Security in 5G Networks (CI-PHY), and the AGNOSTIC project with the European Research Council under Horizon 2020 grant agreement 742648.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/OJCOMS.2021.3079643, IEEE Open Journal of the Communications Society

for MIMO detection. Blind detection for MIMO systems based on clustering was studied in [15], [16], where some ambiguity issues were shown in such unsupervised learning detectors. However, the approaches in [17], [18] employ supervised learning instead, where downlink pilot sequences are sent to train the proposed learning frameworks by considering systems with low-resolution analog-to-digital converters (ADCs). The key idea in [17] is to interpret the MIMO detection problem as a supervised classification problem. In [18], more efficient learning methods were proposed, which leverage the knowledge of cyclic redundancy check (CRC) and the to-be-detected data to further assist the training process.

On the other hand, the problem of detection in a fading MIMO channel with CSIT, i.e., CSI at the Tx, and no explicit CSIR has received relatively little attention in the literature [19]. This setting is common in a *reciprocal* channel, i.e., uplink and downlink channels are the same, where CSIT can be acquired through uplink pilots (known training sequences). For instance, this reciprocity applies to time-division duplex (TDD) communications systems [20]–[23]. In such systems, CSIT can be available without explicitly estimating it at the receiver CSIR. Thus, there are cases where it is interesting to consider systems that can take advantage of CSIT to mitigate the channel effect at the Rx when CSIR is not explicitly available. Motivated by the previously reported benefits from using ML for MIMO detection and inspired by our earlier work where we proposed a similar detection framework for eavesdropping [24], [25], we now propose an ML based technique for MIMO detection in the context of CSIT and no explicit use of CSIR. To our knowledge, the problem under consideration here has not been previously investigated.

The CSIT knowledge is used to precode the downlink transmitted signal to attain several goals: a) maximize the SNR, b) minimize the ICI effect at the Rx and/or c) equalize/mitigate the effect of fading, where the performance of the precoder is directly affected by the quality of the CSIT. Nonetheless, in practical systems, it is extremely difficult to obtain perfect CSIT. Channel estimation errors can occur due to several sources, for instance, pilot length, Doppler effect, and hardware impairments [26]. To support systems with imperfect CSIT, we investigate MIMO detection in the presence of imperfect CSIT. To model these imperfections, we use the Markov-Gauss formulation [27] where we assess the detection performance under varying imperfection conditions. In this setting, we propose a novel learning-based approach for MIMO detection that uses the downlink transmitted *precoded* pilots as training data. Contrary to the aforementioned supervised learning work [17], [18] where the Tx sends *un-precoded* pilots, in our work, the considered pilots are precoded; in fact, these precoded pilots are already deployed in most communication standards [28] for the purpose of signal-to-interference-plus-noise ratio (SINR) estimation. As we shall demonstrate in the numerical results, even in the case when the Rx estimates the effective channel from these precoded pilots and use it to equalize the received signal, the detection performance is still limited. To further improve the detection performance, we propose to use ML to better leverage the knowledge of these precoded pilots.

The contributions of the paper are:

1) We study the problem of symbol detection in a TDD-based coded downlink MIMO system with precoding, considering both block-level precoding (BLP) as well as symbol-level precoding (SLP). We investigate the impact of CSIT degradation on the detection performance at the Rx considering a multiple severity levels of CSIT imperfections.
2) In this setting, we propose supervised learning-based frameworks for soft and hard decoding at the receiver that leverage the knowledge of the transmitted precoded pilots. The proposed learning-based frameworks are generic and function with any multi-label classification algorithm or forward-error correction (FEC) scheme.
3) We propose a lightweight implementation of the learning-based detection frameworks that is suitable for online training, where the detector is optimized for each coherence time. We consider efficient and fast implementations to solve the ML problems for each proposed approach, and assess the prediction accuracy of each, selecting the ones with the highest accuracy to be further employed in the numerical results' section.
4) Numerical results show that severe CSIT degradation impede the decoding process when using a classical receiver. We show that the detection performance improves when using a linear minimum mean square error (MMSE) detector, which uses the estimated effective channel from the downlink precoded pilots. In the presence of severe CSIT deterioration, the proposed data-driven detectors can achieve 4-8 dB power gains when compared to the MMSE detector.

The rest of the paper is organized as follows: Section II introduces the system model, where we discuss CSIR estimation and downlink transmission. In Section III, we propose our novel learning-based detection frameworks whereas in Section IV we present a lightweight implementation of the proposed frameworks that is suitable for online training. Simulation results are discussed in Section V, followed by the conclusion in Section VI.

Notations: Upper and lower case boldface symbols are used to denote matrices and column vectors, respectively. $\|\cdot\|$ represents the Euclidean norm and the Frobenius norm is denoted by $\|\cdot\|_{\mathrm{F}}$. $\mathcal{CN}(\mathbf{m}, \mathbf{Q})$ denotes the circular symmetric complex Gaussian distribution with mean $\mathbf{m}$ and covariance matrix $\mathbf{Q}$. $\mathbb{R}^{m\times n}$ and $\mathbb{C}^{m\times n}$ represent the set of $m \times n$ real matrices, and the set of $m \times n$ complex matrices, respectively. The expectation operator is denoted by $\mathbb{E}[\cdot]$ and the absolute value by $|\cdot|$.

II. System model

We consider a downlink MIMO system with N_{t} antennas at the Tx and N_{r} at the Rx.[1] In a typical MIMO system, the Tx represents the base station with N_{t} antennas while the Rx represents a single user with N_{r} antennas. The Tx sends simultaneously N_{r} different data streams to the Rx.

[1]We assume that the number of antennas at the Tx is greater than or equal to the number of antennas at the Rx, $N_{\mathrm{t}} \geq N_{\mathrm{r}}$.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/OJCOMS.2021.3079643, IEEE Open Journal of the Communications Society

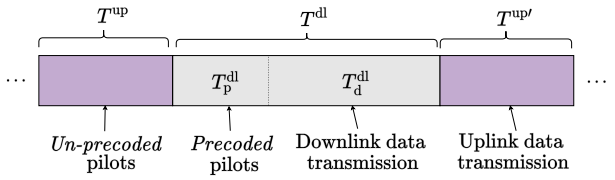


Fig. 1: Coherence time structure.

The transmission from the Tx to the Rx (downlink) and the transmission from the Rx to the Tx (uplink) share the same frequency resource and operate in TDD. Each coherence interval is divided into three phases: uplink training, downlink payload data transmission, and uplink data transmission, however, in this work, we focus on the downlink data transmission. Hence, for our case, we consider a coherence time with two phases: uplink training and downlink data transmission. In the uplink training phase, the users send orthogonal pilot sequences, one for every antenna so that the multiple channels can be estimated simultaneously at the Tx. The obtained channel estimates are used to precode the transmit signals in the downlink.

The notation and assumptions adopted in this work are as follows:

- We let h_{ij} denote the channel coefficient between the ith Tx's antenna and the jth Rx's antenna. We assume that $h_{ij}, i \in \{1,\dots,N_{\mathrm{t}}\}, j \in \{1,\dots,N_{\mathrm{r}}\}$, are i.i.d. random variables (RVs). This assumption models the case wherein the Tx and the Rx are distributed over a wide area, and hence, the set of scatterers is likely to be different for each transmit/receive antenna.
- The channel matrix $\mathbf{H} \in \mathbb{C}^{N_{\mathrm{t}} \times N_{\mathrm{r}}}$, with entries $h_{ij}, i \in \{1,\dots,N_{\mathrm{t}}\}, j \in \{1,\dots,N_{\mathrm{r}}\}$, embodies small-scale fading, which is assumed to be static during each coherence time and changes independently from one coherence time to another. We consider block-fading Rayleigh channel to reflect a rich scattering environment.
- We assume channel reciprocity, i.e., in each coherence time, the channel coefficients in the uplink are the conjugate of those in the downlink. This assumption requires TDD operation and appropriate calibration of the hardware chains.
- We consider a block-fading channel where the channel remains constant for T symbol periods (SPs), i.e., the coherence time length. Since we operate in TDD, as depicted in Fig. 1, the coherence time is shared between uplink pilots for CSIT estimation, downlink transmission, and uplink transmission, where T^{up}, T^{dl}, and $T^{\mathrm{up}\prime}$ represent their allocated time in SPs, respectively.
- We let d_j denote the data symbol associated with the jth data stream intended for the jth Rx's antenna, which satisfies $\mathbb{E}[|d_j|^2] = 1$. In each SP, the Tx sends $\mathbf{d} = [d_1 \dots d_{N_{\mathrm{r}}}]^{\mathrm{T}}$ from the N_{t} antennas, where the symbols $d_j, j \in \{1,\dots,N_{\mathrm{r}}\}$, are mutually independent.

A. CSIT Estimation through Uplink Training

The Tx computes the channel estimate $\hat{\mathbf{H}} \in \mathbb{C}^{N_{\mathrm{t}} \times N_{\mathrm{r}}}$ using uplink pilots, as depicted in Fig. 1, that will be used subsequently to define the precoder for the downlink transmission phase that encompasses T^{dl} SPs. The quality of the channel estimate depends mainly on the pilot length, the particular channel estimation technique that is being employed, and hardware impairments, e.g., uncalibrated hardware chains in this context [26]. In our work, we use the Gauss-Markov formulation to model $\hat{\mathbf{H}}$ in order to reflect these imperfections [27], ranging from perfect estimation to a completely inaccurate estimation. In this setting, the channel estimate $\hat{\mathbf{H}}$ is obtained using the *actual* channel $\mathbf{H}$ as

$$\hat{\mathbf{H}} = \tau \mathbf{H} + \sqrt{1-\tau^2}\mathbf{E} \tag{1}$$

where the scalar $\tau \in [0,1]$ specify the quality/accuracy of the instantaneous CSI — $\tau = 1$ corresponds to perfect CSIT while $\tau = 0$ indicates that $\hat{\mathbf{H}}$ is completely incorrect and uncorrelated with the actual CSI —, and $\mathbf{E} \in \mathbb{C}^{N_{\mathrm{t}} \times N_{\mathrm{r}}}$ represents the random error [1], [29] where each term follows a circularly symmetric normal distribution $\mathcal{CN}(0,1)$.

B. Downlink Transmission

Once the channel estimate $\hat{\mathbf{H}}$ is obtained, the Tx uses it to precode pilot and data symbols to transmit to the Rx during $T_{\mathrm{p}}^{\mathrm{dl}}$ and $T_{\mathrm{d}}^{\mathrm{dl}}$ SPs, respectively, as depicted in Fig. 1.

In each SP, the Tx sends N_{r} symbols to the Rx, which could be pilot or data symbols. Hence, for the nth SP, the received data signal y_j at the jth antenna of the Rx can be expressed as

$$y_j[n] = \mathbf{h}_j^{\mathrm{H}} \mathbf{x}_{\mathbf{d}}[n] + z_j[n] \tag{2}$$

$\mathbf{h}_j = [h_{1j} \dots h_{N_{\mathrm{r}}j}]^{\mathrm{T}} \in \mathbb{C}^{N_{\mathrm{t}} \times 1}$ is the channel from the N_{t} transmit antennas to the jth receive antenna, $\mathbf{x}_{\mathbf{d}}[n] \in \mathbb{C}^{N_{\mathrm{t}} \times 1}$ is the precoded transmitted signal with power $\rho_{\mathrm{d}} \geq 0$, and $z_j[n]$ is the additive white Gaussian noise (AWGN) at the jth receive antenna with variance σ_z^2.

The above model can be rewritten in a more compact form by collecting the received signal at all Rx's antennas:

$$\mathbf{y}[n] = \mathbf{H}^{\mathrm{H}} \mathbf{x}_{\mathrm{d}}[n] + \mathbf{z}[n], \tag{3}$$

where $\mathbf{z}[n] \in \mathbb{C}^{N_{\mathrm{r}} \times 1}$ collects the independent AWGN components of all antennas. Consequently we define the transmit SNR as $\eta = \frac{\rho_{\mathrm{d}} N_{\mathrm{t}}}{\sigma_z^2}$.

In our work, we consider the case when the Tx uses the regularized zero-forcing (RZF) [30], which is a linear block-level precoder, as well as the case of a conventional symbol-level precoder, i.e., the constructive interference for sum power minimization (CISPM) approach in [31].

The RZF precoding main goal is to maximize the sum of the SINR. It minimizes the interference signal while optimizing the received power. It is considered as a precoding approach between the matched filter (MF) and zero-forcing (ZF) precoders, which can be expressed as a linear combination of MF and ZF precoders [32]. The RZF precoding matrix can be expressed as

$$\mathbf{W} = \hat{\mathbf{H}}^{\mathrm{H}} \left(\hat{\mathbf{H}}^{\mathrm{H}} \hat{\mathbf{H}} + \alpha \mathbf{I}_{N_{\mathrm{t}}} \right)^{-1} \tag{4}$$

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/OJCOMS.2021.3079643, IEEE Open Journal of the Communications Society

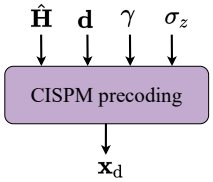


Fig. 2: Structure of the CISPM precoding scheme.

where α is a regularization parameter whose value is fixed during the transmission and $\mathbf{I}_{N_\mathrm{t}}$ denotes the $N_\mathrm{t} \times N_\mathrm{t}$ identity matrix.

We note that, in order to respect the power constraint $\mathbb{E}[\|\mathbf{W}\mathbf{d}\|^2] = 1$, we normalize $\mathbf{W}$ as follows: $\widehat{\mathbf{W}} = \frac{\mathbf{W}}{\|\mathbf{W}\|_\mathrm{F}}$. Hence, using this notation, the RZF precoded signal for the nth SP can be expressed as

$$\mathbf{x}_\mathrm{d}[n] = \sqrt{\rho_\mathrm{d}}\widehat{\mathbf{W}}\mathbf{d}[n]. \tag{5}$$

As for the constructive interference for sum power minimization symbol-level precoding scheme, it is designed to exploit the multi-user interference for power gains. This scheme propels the Rx's received signals deeper into the correct detection region of the desired symbol for each receive antenna. The CISPM precoded signal for the nth SP can be computed as

$$\begin{aligned}\mathbf{x}_\mathrm{d}(\mathbf{d}, \hat{\mathbf{H}}, \boldsymbol{\gamma}, \sigma_z) = & \arg\min_{\mathbf{x}} ||\mathbf{x}||^2 \\ & \text{subject to} \\ & \mathrm{Re}\{\hat{\mathbf{h}}_j^\mathrm{H}\mathbf{x}\} \trianglelefteq \sigma_z\sqrt{\gamma_j}\mathrm{Re}\{d_j\},\ \forall j \\ & \mathrm{Im}\{\hat{\mathbf{h}}_j^\mathrm{H}\mathbf{x}\} \trianglelefteq \sigma_z\sqrt{\gamma_j}\mathrm{Im}\{d_j\},\ \forall j,\end{aligned} \tag{6}$$

where $\hat{\mathbf{h}}_j \in \mathbb{C}^{N_\mathrm{t}\times 1}$ is the channel estimate from the N_t transmit antennas to the jth receive antenna, $\gamma_j \geq 0$ is the target SINR for the jth receive antenna with $\boldsymbol{\gamma} = [\gamma_1 \dots \gamma_{N_\mathrm{r}}]^\mathrm{T} \in \mathbb{R}^{N_\mathrm{r}\times 1}$ representing the target SINR for all Rx's antennas, and the operator $\trianglelefteq$ denotes[2] the correct detection region [35].

As depicted in Fig. 2, the optimization problem in eq. (6) takes inputs: the estimated CSIT $\hat{\mathbf{H}}$, the symbols to transmit $\mathbf{d}$, the target SINR at the N_r antennas $\boldsymbol{\gamma}$, and the noise standard deviation σ_z. The objective function's aim is to minimize the transmit power subject to some constructive interference constraints that are applied to each receive antenna. The constraints' aim is to place the real/imaginary parts of the noiseless received signal at the jth antenna, $\hat{\mathbf{h}}_j^\mathrm{H}\mathbf{x}$, in the detection region corresponding to the real/imaginary parts of the jth intended symbols to transmit. Specifically, with a minimum value of $\sigma_z\sqrt{\gamma_j}$ to guarantee a specific target SINR for each receive antenna. In other words, this scheme propels the Rx's received signals deeper into the correct detection region of the desired symbol.

Thus, the CISPM scheme minimizes the transmit power while guaranteeing a certain target SINR at the Rx through constructive interference constraints. Contrary to the RZF scheme where the precoding matrix $\widehat{\mathbf{W}}$ is used for the entire coherence time, in the SLP approach, for each SP, the precoding module directly designs the transmitted signal vector $\mathbf{x}_\mathrm{d}$ based on both the CSIT $\hat{\mathbf{H}}$ and the input data symbols $\mathbf{d}$. Even though SLP schemes are computed for every SP, an efficient implementation in hardware has been proposed in [36].

[2]For further detailed information, the reader should refer to [33], [34].

III. Learning-based MIMO detection frameworks

In this section, we propose two ML detection frameworks, where the Rx uses the transmitted precoded pilot symbols as training data to accurately hard/soft decode the transmitted symbols. We note that these precoded pilots are already used in some communication standards [28] for the purpose of SINR estimation. Thus, we propose to leverage this existing knowledge at the Rx to train our learning-based detectors. Both of these frameworks are comprised of training and inference phases. In the following, we use RZF precoding as an example for a complete explanation. However, the proposed detection frameworks are valid for any precoding scheme used at the Tx. We stress that in our work we propose to use machine learning only at the Rx, whereas at the Tx we assume conventional precoding where RZF is used as an example for the precoding used at the Tx. The proposed learning-based frameworks are generic and function with any ML or channel coding algorithm, thus mathematical derivations of these algorithms are omitted in this section.

A. Learning-based framework for the proposed MIMO soft detection scheme

As pointed out earlier, the detection occurs at the Rx during the downlink part of the coherence time T^dl, where the training phase takes place during the T_p^dl SPs followed by the inference phase in the remaining T_d^dl slots. We note that these pilots might be interleaved with data and do not have to be sequential.

As depicted in Fig. 3, our learning-based soft detector encompasses two steps: 1) training phase, where the ML model is trained using the T_p^dl precoded pilot symbols; 2) inference phase, where probability densities are estimated and employed to calculate the LLRs which are then fed to a soft decoder. Below we detail each phase.

*1) **Training phase**:* In each SP of T_p^dl, the Tx sends N_r pilot symbols, $\mathbf{p} = [p_1 \dots p_{N_\mathrm{r}}]^\mathrm{T}$ satisfying $\mathbb{E}[|p_j|^2] = 1$, which are pseudo-random sequences with each symbol corresponding to one Rx antenna. The corresponding received pilot signal at the jth antenna during the nth SP, $y_j^\mathrm{p}[n] \in \mathbb{C}$, can be written as

$$y_j^\mathrm{p}[n] = \mathbf{h}_j^\mathrm{H}\mathbf{x}_\mathbf{p}[n] + z_j[n], \tag{7}$$

where $\mathbf{x}_\mathbf{p} \in \mathbb{C}^{N_\mathrm{t}\times 1}$ is the RZF precoded transmitted pilot signal with power $\rho_\mathrm{p} \geq 0$ such that $\mathbf{x}_\mathbf{p} = \sqrt{\rho_\mathrm{p}}\widehat{\mathbf{W}}\mathbf{p}$.

Thus, collecting all of the received pilot signals at all antennas, the equivalent received pilot signals during the nth SP, $\mathbf{y}^\mathrm{p}[n] \in \mathbb{C}^{N_\mathrm{r}\times 1}$, can be expressed as

$$\mathbf{y}^\mathrm{p}[n] = \mathbf{H}^\mathrm{H}\mathbf{x}_\mathrm{p}[n] + \mathbf{z}[n]. \tag{8}$$

As depicted in Fig. 3, for the nth SP, the Rx obtains $\mathbf{y}^\mathrm{p}[n]$ and $\mathbf{p}$. The Rx creates a single set that collects together the received pilot signal $\mathbf{y}^\mathrm{p}[n]$ and the pilot symbols $\mathbf{p}_\mathrm{b}$

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/OJCOMS.2021.3079643, IEEE Open Journal of the Communications Society

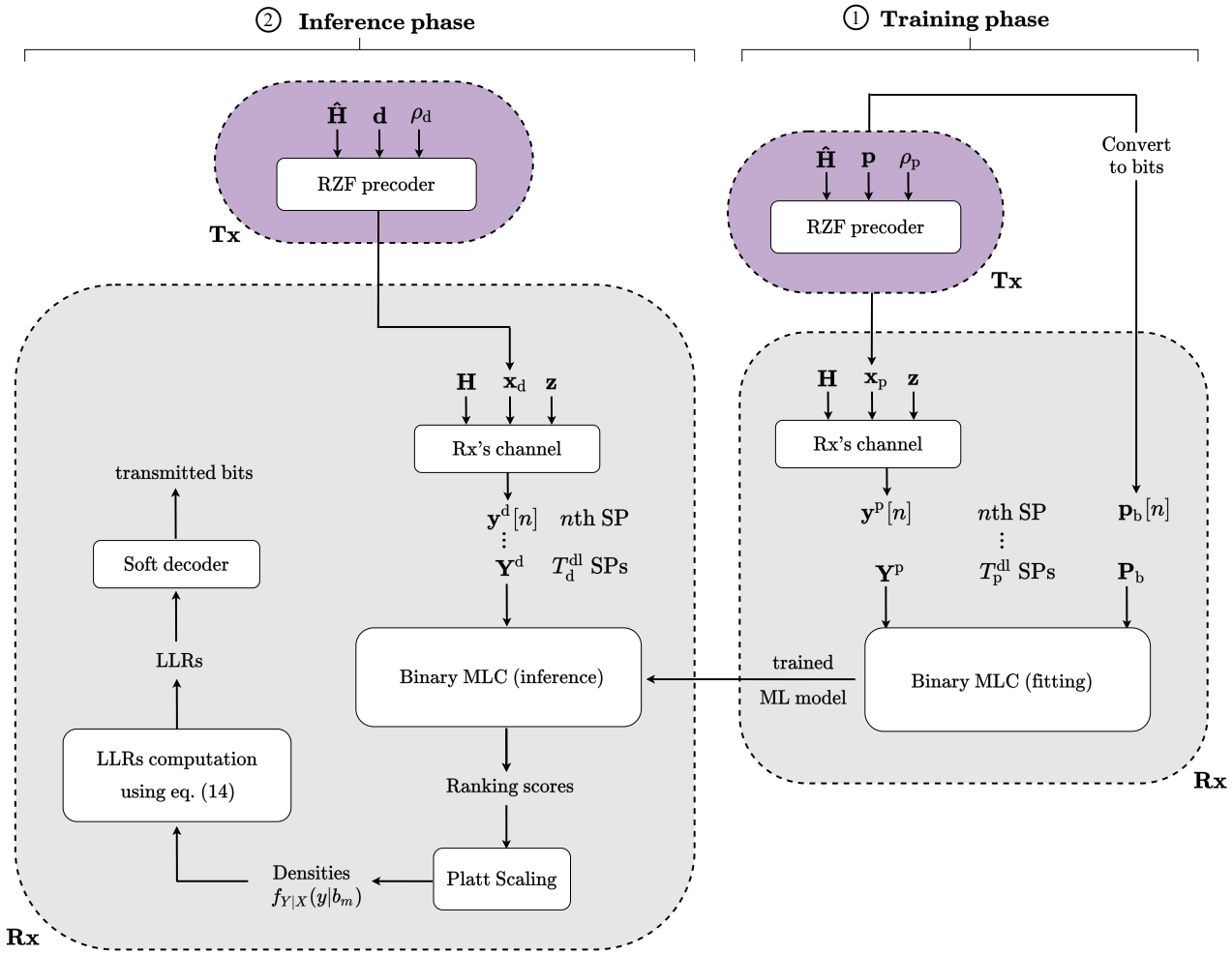


Fig. 3: Overview of the proposed learning-based soft detector.

corresponding to the bit representation of $\mathbf{p}$, for the LLRs to be computed on a per-bit basis. Specifically, the subscript b in $\mathbf{p}_\mathrm{b}$ stands for "bit", where the pilot symbols are represented in bits, e.g., $p_j \in \{\text{“00”}, \text{“01”}, \text{“11”}, \text{“10”}\}$ in the case of QPSK modulation.

As illustrated in Fig. 3, to perform the training, the Rx first collects all the T_p^dl received pilot signals $\mathbf{y}^\mathrm{p}$ with their pilot symbols $\mathbf{p}_\mathrm{b}$ in one set, which is referred to as the *training set*, which can be written as

$$\mathcal{D}_\mathrm{p}^\mathrm{s} = \{\mathbf{Y}^\mathrm{p}, \mathbf{P}_\mathrm{b}\}, \tag{9}$$

where $\mathbf{Y}^\mathrm{p} \in \mathbb{C}^{T_\mathrm{p}^\mathrm{dl} \times N_\mathrm{r}}$ are the received pilot symbols at the Rx during T_p^dl SPs and $\mathbf{P}_\mathrm{b} \in \mathbb{C}^{T_\mathrm{p}^\mathrm{dl} \times N_\mathrm{r}}$ are the transmitted pilot symbols for T_p^dl SPs. In the ML context, $\mathbf{Y}^\mathrm{p}$ and $\mathbf{P}_\mathrm{b}$ are referred to as features[3] and labels, respectively. This class of ML problems is named supervised ML. Specifically, this ML problem is considered as a binary multi-label classification (MLC) problem [37] as more than one label is used for each example and the format of the label is binary; each row in the training set $\mathcal{D}_\mathrm{p}^\mathrm{s}$ represents an example. In our case, an example is any received pilot signal $\mathbf{y}^\mathrm{p}$ in a given SP.

Hence, as depicted in Fig. 3, the training dataset is inputed to the MLC fitting module that will output the so-called "trained ML model". Our goal is to generate a well-fitted model to accurately predict new features, that are of similar nature to the ones used in the training, by minimizing the *bias*. A high bias leads to underfitting, i.e., the model is unable to predict well the labels in the training, whereas overfitting manifests when the model predicts very well the training data but poorly the data outside of the training set [38].

*2) **Inference phase**:* This phase, on the other hand, takes place during the T_d^dl SPs of the downlink part of the coherence time. In particular, as depicted in Fig. 3, in each SP of T_d^dl, the Tx sends N_r data symbols. The overall received data signals at all antennas of the Rx during the nth SP, $\mathbf{y}^\mathrm{d}[n] \in \mathbb{C}^{N_\mathrm{r} \times 1}$, can be expressed as

$$\mathbf{y}^\mathrm{d}[n] = \mathbf{H}^\mathrm{H}\mathbf{x}_\mathrm{d}[n] + \mathbf{z}[n]. \tag{10}$$

As depicted in Fig. 3, the Rx first collects the T_d^dl received data signals in one set $\mathcal{D}_\mathrm{d} = \mathbf{Y}^\mathrm{d} \in \mathbb{C}^{T_\mathrm{d}^\mathrm{dl} \times N_\mathrm{r}}$. In ML terminology, the set $\mathcal{D}_\mathrm{d}$ is the test/evaluation dataset.[3]

Generally, the goal of classification is to predict labels. In this context, however, we are not interested in the predicted

[3]We note that the features in $\mathbf{Y}^\mathrm{p}$ are complex-valued and are not suitable for common ML algorithms. This is generally addressed by considering real and imaginary parts separately.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/OJCOMS.2021.3079643, IEEE Open Journal of the Communications Society

labels (bits) but rather in the corresponding predicted densities (soft outputs). These soft outputs are used subsequently to compute the log-likelihood ratios (LLRs), which indicate the reliability of the predicted bits.

Before tackling the computation of these LLRs, we first provide an overview of LLRs computation by recalling some fundamental definitions in binary detection [39]. Let X be a binary RV, acting as the correct hypothesis, with possible values $\{b_0, b_1\}$ and a priori probabilities p_0 and p_1. Herein, X models one bit in a transmitted symbol. Let Y be an RV with conditional probability density $f_{Y|X}(y|b_m)$ that is finite and non-zero for all $y \in \mathbb{R}$ and $m \in \{0, 1\}$. In our context, Y models the received signal at the Rx's antenna for a given SP. We note that the conditional densities $f_{Y|X}(y|b_m), m \in \{0, 1\}$, are called *likelihoods*. The marginal density of Y is given by $f_Y(y) = p_0 f_{Y|X}(y|b_0) + p_1 f_{Y|X}(y|b_1)$. Hence, the *a posteriori* probability of X can be expressed as

$$f_{X|Y}(b_m|y) = \frac{p_m f_{Y|X}(y|b_m)}{f_Y(y)}, \tag{11}$$

where $m \in \{0, 1\}$. To maximize the probability of correct detection, the maximum a posteriori (MAP) rule can be written as

$$\frac{p_0 f_{Y|X}(y|b_0)}{f_Y(y)} \underset{\tilde{X}=b_1}{\overset{\tilde{X}=b_0}{\gtrless}} \frac{p_1 f_{Y|X}(y|b_1)}{f_Y(y)}, \tag{12}$$

where $\tilde{X}$ denotes the decision on the RV X. Rearranging (12) and canceling $f_Y(y)$, we obtain the *likelihood ratio*

$$\Lambda(y) = \frac{f_{Y|X}(y|b_0)}{f_{Y|X}(y|b_1)} \underset{\tilde{X}=b_1}{\overset{\tilde{X}=b_0}{\gtrless}} \frac{p_1}{p_0}, \tag{13}$$

where the quantity $\frac{p_1}{p_0}$ is called the *threshold* and depends only on the a priori densities. Hence, the log-likelihood ratio $\mathrm{LLR}(y)$ can be expressed as follows:

$$\mathrm{LLR}(y) = \ln\left[\frac{f_{Y|X}(y|b_0)}{f_{Y|X}(y|b_1)}\right]. \tag{14}$$

Therefore, in order to compute the LLRs, we need to first calculate the densities in eq. (14).

As depicted in Fig. 3, to obtain the densities in eq. (14), we feed the test dataset to the binary MLC inference module along with the trained ML model. However, the binary MLC inference module does not output densities, but rather predicted labels with their ranking scores. These scores are uncalibrated values that do not constitute probability densities; these scores signify the confidence level of the inference. That is, if $s(b_0)$ and $s(b_1)$ represent the scores of the predicted bit's possibilities b_0 and b_1, respectively, and if $s(b_0) \leq s(b_1)$, then $f_{Y|X}(y|b_0) \leq f_{Y|X}(y|b_0)$. Fortunately, there are existing methods to convert these ranking scores into densities [40]. In Section IV, we discuss the details of an efficient implementation that estimates densities from ranking scores, which will be used subsequently in the experiments.

Once the likelihoods $f_{Y|X}(y|b_m)$ are obtained, the LLRs can be computed using eq. (14), after which the Rx can simply feed the computed LLRs to the soft decoder (e.g., a Viterbi decoder [41]) to obtain the transmitted data.

B. Learning-based framework for the proposed MIMO hard detection scheme

The proposed learning-based hard MIMO detector is also comprised of two phases: 1) training phase, where the ML model is trained using the precoded pilot symbols as training data; 2) inference phase, where the module directly predicts the coded bits (in the form of decimals), which are then mapped into bits to finally be fed to a conventional hard decoder to recover the transmitted bits.

*1) **Training phase**:* As illustrated in Fig. 4, during the nth SP within $T_{\mathrm{p}}^{\mathrm{dl}}$, the Tx sends $\mathbf{p}[n] \in \mathbb{C}^{N_{\mathrm{r}} \times 1}$ which becomes $\mathbf{x}_{\mathrm{p}}[n] \in \mathbb{C}^{N_{\mathrm{t}} \times 1}$ after precoding. Subsequently, the Rx receives at all its antennas, $\mathbf{y}^{\mathrm{p}}[n] \in \mathbb{C}^{N_{\mathrm{r}} \times 1}$, as detailed in eq. (8).

In addition to the $\mathbf{y}^{\mathrm{p}}[n]$, the Rx also has the knowledge of the pilot symbols $\mathbf{p}[n]$. We denote the decimal representation of $\mathbf{p}[n]$ by $\mathbf{p}_{\mathrm{h}}[n]$, where the subscript h stands for "hard"; e.g., $p^j \in \{0, 1, 2, 3\}$ in the case of QPSK modulation.

As illustrated in Fig. 4, the Rx creates a single set that maps the received signal $\mathbf{y}^{\mathrm{p}}[n]$ to the corresponding pilot symbols $\mathbf{p}_{\mathrm{h}}[n]$. Thus, the training set $\mathcal{D}_{\mathrm{p}}^{\mathrm{h}}$ is

$$\mathcal{D}_{\mathrm{p}}^{\mathrm{h}} = \{\mathbf{Y}^{\mathrm{p}}, \mathbf{P}_{\mathrm{h}}\}, \tag{15}$$

where $\mathbf{P}_{\mathrm{h}} \in \mathbb{C}^{T_{\mathrm{p}}^{\mathrm{dl}} \times N_{\mathrm{r}}}$ are the the pilot symbols during $T_{\mathrm{p}}^{\mathrm{dl}}$ SPs. Since the labels are represented in decimals, considering a modulation order of M, for each label there are M classes. In the case of non-binary modulations, this problem is a multi-class MLC problem [37]. Thus, as depicted in Fig. 4, the training dataset $\mathcal{D}_{\mathrm{p}}^{\mathrm{h}}$ is fed to the multi-class MLC fitting module that in sequence outputs a trained ML model, which will be used thereafter in the inference phase.

*2) **Inference phase**:* As depicted in Fig. 4, for the nth SP, the Tx sends the data symbols $\mathbf{d}[n]$ to the Rx in the form of the precoded signal $\mathbf{x}_{\mathrm{d}}[n]$. The corresponding received signals at the Rx's antennas is $\mathbf{y}^{\mathrm{d}}[n]$, as detailed in eq. (10).

Therefore, for $T_{\mathrm{d}}^{\mathrm{dl}}$ SPs, the overall received data signal is $\mathbf{Y}^{\mathrm{d}} = \{\mathbf{y}^{\mathrm{d}}[n]\}, n \in \{1, \ldots, T_{\mathrm{d}}^{\mathrm{dl}}\}$, which constitutes the evaluation set. Thus, for the inference, we feed the set $\mathbf{Y}^{\mathrm{d}}$ to the multi-class MLC inference module. Contrary to the proposed soft detection scheme, herein we are interested in predicting the labels, i.e., hard outputs. As depicted in Fig. 4, to obtain the labels, we feed the evaluation set as well as the previously trained model to the multi-class MLC inference module. We note that the predicted labels are in the form of decimals, i.e., the same nature of the labels used in the training phase. Once the labels are predicted, they will be first mapped into bits to obtain the coded bits, which will then be fed to a hard decoder to finally obtain the transmitted bits.

C. Scalability of the proposed ML detection frameworks to multi-user MIMO systems

Even though this work investigates the case of a single-user (SU) MIMO system, the proposed detection frameworks can be extended to multi-user (MU) MIMO systems [42].

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/OJCOMS.2021.3079643, IEEE Open Journal of the Communications Society

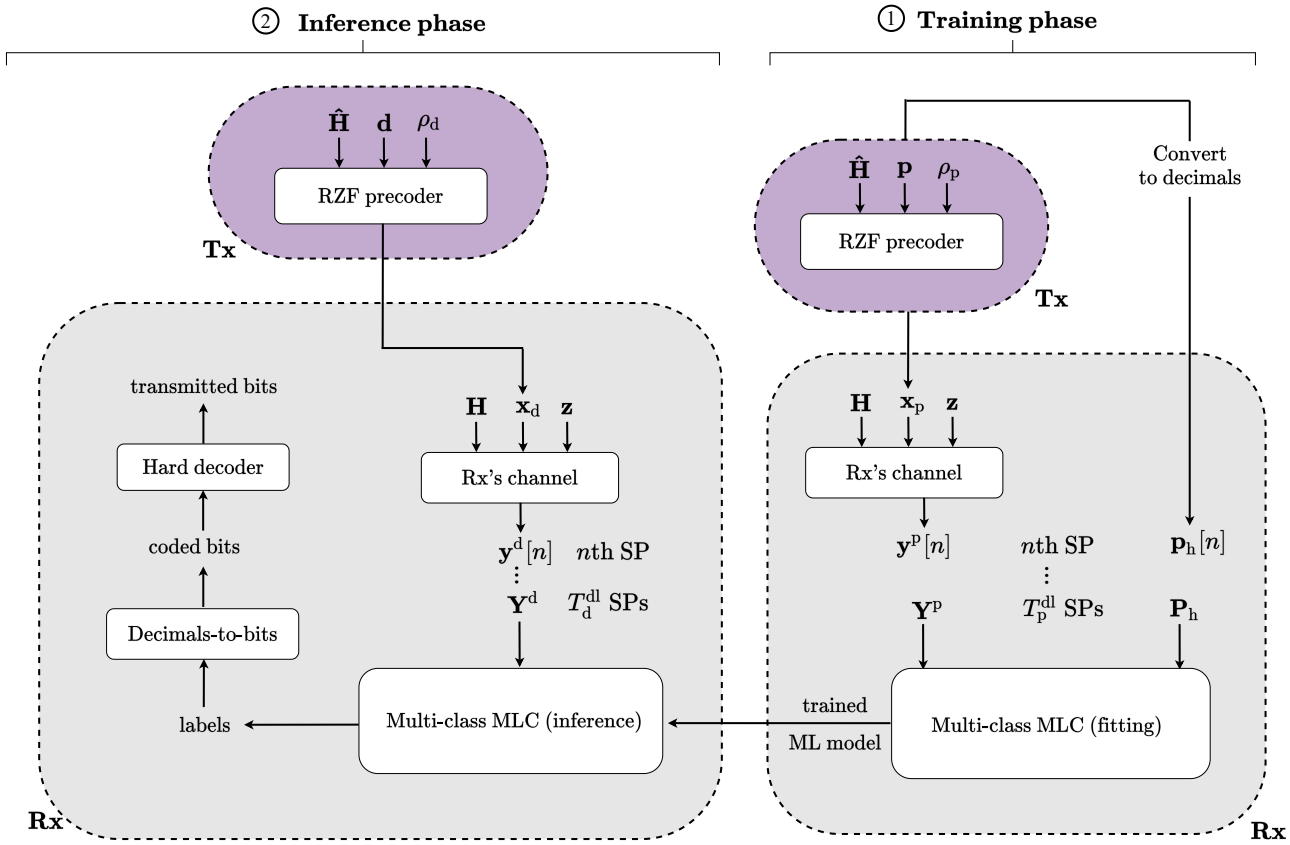


Fig. 4: Overview of the proposed learning-based hard detector.

To illustrate, let us consider an MU-MIMO system with N_t antennas at the Tx and K Rxs with N_i antennas each with N_iK total downlink streams such that $N_\mathrm{t} > N_iK$. For the nth SP, the received signal at the ith Rx, $\mathbf{y}_i[n] \in \mathbb{C}^{N_i \times 1}$, can be expressed as follows

$$\mathbf{y}_i[n] = \mathbf{H}_i^\mathrm{H}\mathbf{x}_\mathrm{d}[n] + \mathbf{z}_i[n], \tag{16}$$

$\mathbf{H}_i \in \mathbb{C}^{N_\mathrm{t} \times N_i}$ is the channel matrix from the Tx to the ith Rx and $\mathbf{z}_i[n] \in \mathbb{C}^{N_i \times 1}$ collects the independent AWGN components of all of the ith Rx's antennas. For each SP, the ith Rx receives N_i pilot symbols. Therefore, the ith Rx maps each received signal $\mathbf{y}_i[n]$ with the corresponding pilot symbols vector $\mathbf{p}_i \in \mathbb{C}^{N_i \times 1}$. Similarly to the SU scenario, the Rx first collects the received signals with their corresponding pilots for the T_p^dl SPs then construct the training and evaluation datasets using the exact same approach in Section III-A and Section III-B.

All in all, the only difference with respect to the single-user case is the second dimension of the matrices $\mathbf{Y}^\mathrm{p}, \mathbf{P}_\mathrm{b}, \mathbf{P}_\mathrm{h}, \mathbf{Y}^\mathrm{d}$ constructing the training and evaluation datasets, having N_i for the ith Rx in the MU use case instead of N_r for the single-user scenario.

IV. Lightweight implementation of the proposed detection frameworks

In this section, we first discuss our efficient implementation to solve the MLC problem for both of the proposed soft and hard detectors. For the soft detector in particular, we propose a fast algorithm to estimate the probability densities from the ranking scores, which will be used for LLRs computation, and demonstrate its efficacy by plotting the distribution of the predicted densities. Next, we discuss the considered state-of-the-art classifiers used to solve the classification problem for each label. To clarify, the proposed efficient implementation is hierarchical; we propose an implementation for the MLC problem as well as the specific classifiers used for each label. This implementation is further detailed below. We note that the chosen algorithms and approaches for the proposed learning-based detectors have been carefully investigated in terms of suitability for online learning, where Rx optimizes its detector for every coherence time.

We should mention that we did not use deep learning [43] in this context despite its high performance in several areas mainly because it requires considerable amount of training data, which is not available in our case. Indeed, in each coherence time, only a limited portion of downlink transmission is dedicated to pilots. As we consider online learning, i.e., the detector is optimized for each coherence time, the pilots of one coherence time could not be combined with the pilots of another coherence time, thus the limitation of the training data. Besides, deep learning typically requires millions of parameters to train, which makes it prohibitively expensive in our application scenario.

In the literature, the proposed methods for solving MLC problems can be grouped into two main categories: a) problem transformation methods and b) algorithm adaptation methods

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/OJCOMS.2021.3079643, IEEE Open Journal of the Communications Society

[37]. Transformation methods are those that transform the MLC problem and decompose it into multiple single-label classification (SLC) problem instances, whereas adaptation methods are designed to solve MLC problems directly. Transformation methods are simple and efficient, however, adaptation methods are designed for maximal efficiency, which usually makes them more complex and more computationally intensive compared to transformation methods.

Motivated by the online training approach, we adopt the transformation approach to solve the MLC problem because of its sufficient efficiency and low complexity. In this setting, we consider two transformation methods, binary relevance (BR) [44] and classifier chain (CC) [45]. BR is the most simple and efficient method to solve MLC problems, where multiple SLC are trained independently and their individual outputs are combined to form the multi-label output. Even though this method was designed for binary (two class) labels, as the name implies, it is also implemented for multi-class SLC problems. Despite its popularity and simplicity, its only drawback is that it does not consider label correlations. CC, on the other hand, takes into account the correlation between labels by using the outputs of the previously trained classifiers as features for the subsequent ones in the chain, except for the first classifier. In CC, a chain of SLC is constructed where each classifier, in addition to the related input label, also uses the inferences of other classifiers, thus considering the correlation between labels. We refer to these transformation implementations by "BR" and "CC" accordingly.

The time complexity of BR and CC algorithms is $\mathcal{O}\big(N_{\mathrm{r}} f_c(|\mathbf{Y}^{\mathrm{p}}|, |\mathcal{D}_{\mathrm{p}}^{\mathrm{s}|\mathrm{h}}|)\big)$ and $\mathcal{O}\big(N_{\mathrm{r}} f_c(|\mathbf{Y}^{\mathrm{p}}| + N_{\mathrm{r}}, |\mathcal{D}_{\mathrm{p}}^{\mathrm{s}|\mathrm{h}}|)\big)$, respectively, where $f_c(\cdot)$ is the complexity of the underlying classifier [46]. $f_c(\cdot)$ is heavily dependent on the classifier used and the solver used for its implementation, which makes it challenging to obtain a closed-form big $\mathcal{O}$ representation of it. For a quantitative analysis of the time complexity, we measure the total time it takes the algorithm to finish (in milliseconds), which is commonly being refereed to as *runtime* [9]. Consequently, we present the runtime complexity analysis in the following section.

Concerning the proposed soft detector, we adopt an efficient and fast method to accurately estimate the densities $f_{Y|X}(y|b_m)$ from the outputted ranking scores of the MLC module, namely, the Platt scalling approach in [47]. This method is used to transform the uncalibrated scores generated by the classification module into densities. Platt scaling works by fitting a logistic regression model to the classifier's scores. The densities $f_{Y|X}(y|b_m)$ according to the Platt scaling algorithm can be computed as

$$f_{Y|X}(y|b_m) = \frac{1}{1 + \exp(A f_y(b_m) + B)}, \tag{17}$$

where $f_y(b_m)$ is the classifier ranking score and scalars A and B are the sigmoid parameters [47] learned by the algorithm, which are calculated using a cross-entropy loss function and an internal threefold cross-validation to prevent overfitting.

Regarding the implementation of the SLCs, we have experimented with several state-of-the-art classifiers.[4] For the details about the classifiers' hyper-parameters used in this experiment, we have used the default parameters of the Python modules, where module "scikit-learn" version "0.23.1" was used to implement the "CC" method as well as the "MaxEnt", "SVM", "R_Forest", "KNN", "Decision_Tree", and "Extra_Trees" classifiers, module "scikit-multilearn" version "0.2.0" was used for the "BR" method, module "xgboost" version "1.1.1" was employed for the "XGB" classifier, and module "lightgbm" version "2.3.1" was used for the "LightGBM" classifier implementation. We note that the same implementation is used in the following section. In Table I, we compare the prediction accuracy of the proposed soft and hard detectors using the proposed implementations, considering both RZF and CISPM precoding schemes with QPSK modulation. For a fair comparison between these precoding schemes, we set the transmit SNR η and the target SINR at the jth receive antenna γ_j to the same value such that all the examined schemes have the same transmit power. For simplicity, we set $\gamma_j = \gamma, j \in \{1, \ldots, N_{\mathrm{r}}\}$. The parameters used for this experiment are: $\tau = 0.8$ (severe CSIT degradation), $N_{\mathrm{t}} = N_{\mathrm{r}} = 8$, $\eta = \gamma_j = 6$ dB such that signal powers of pilot and data signals is the same, $\rho_{\mathrm{p}} = \rho_{\mathrm{d}}$, and a frame size of 300 symbols with $T_{\mathrm{p}}^{\mathrm{up}} = T_{\mathrm{d}}^{\mathrm{up}}$. We note that these results represent the averaged results over 100 different channel realizations. We also note that this accuracy applies before channel decoding, i.e., by comparing the ML predicted labels to the actual coded transmitted symbols.

As observed in Table I, the prediction accuracy values are high despite the severe CSIT degradation ($\tau = 0.8$) and the relatively small SNR $\eta = 6$ dB employed. We also observe, for both of the precoding schemes, the CC approach achieves better results when using the Soft scheme, however, the BR approach slightly outperforms the CC approach when using the Hard scheme. Furthermore, MaxEnt classifier achieves the highest prediction accuracy amongst all classifiers regardless of the precoding scheme employed and the detector used. Therefore, in the numerical results, to achieve the highest detection performance, we adopt the CC approach for the soft detector and the BR method for the hard one, where both of them employ the MaxEnt classifier to solve the SLC problem for each label. In the numerical results, we refer to these implementations by "ML - Soft" for the soft detection scheme and "ML - Hard" for the hard detection one.

With regards to the employed likelihoods estimation method for the "ML - Soft" scheme, Fig. 5 depicts the distribution of the estimated likelihoods $f_{Y|X}(y|b_m)$ in the presence of severe CSIT degradation ($\tau = 0.8$). The parameters used for this simulation are: $N_{\mathrm{t}} = 15$, $N_{\mathrm{r}} = 8$, $\eta \in \{0, 6\}$ dB, a frame size of 2000 with QPSK modulation and RZF precoding at the Tx. We stress that a predicted probability of 0.5 indicates that the predictor is fully unsure of the predicted bit, on the other hand, a value close to 1 means the predictor is very confident that the predicted bit is a 1 whereas a probability close to 0 implies predictor is confident it is a 0. For a transmit SNR

[4]We note that we did not experiment with neural-network based classifiers because of their relative high complexity and the scarcity of the training data.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/OJCOMS.2021.3079643, IEEE Open Journal of the Communications Society

TABLE I: Prediction accuracy of the proposed learning-based detectors with several state-of-the-art classifiers when using CIPSM and RZF precoding at the Tx.

Classifiers	**CISPM**				**RZF**			
	CC		**BR**		**CC**		**BR**	
	Soft	**Hard**	**Soft**	**Hard**	**Soft**	**Hard**	**Soft**	**Hard**
MaxEnt	0.9439	0.9322	0.9398	0.9341	0.9096	0.8967	0.9058	0.8995
SVM	0.9015	0.8800	0.9011	0.8806	0.8792	0.8609	0.8793	0.8642
R_Forest	0.8419	0.8305	0.8440	0.8313	0.8337	0.8194	0.8376	0.8234
KNN	0.7874	0.7674	0.7888	0.7719	0.7712	0.7428	0.7754	0.7557
Decision_Tree	0.7579	0.7334	0.7582	0.7364	0.7410	0.7169	0.7427	0.7192
Extra_Trees	0.8207	0.8212	0.8577	0.8393	0.8116	0.8108	0.8467	0.8270
LightGBM	0.8667	0.8452	0.8673	0.8464	0.8500	0.8309	0.8495	0.8326
XGB	0.8611	0.8424	0.8611	0.8434	0.8450	0.8279	0.8456	0.8285

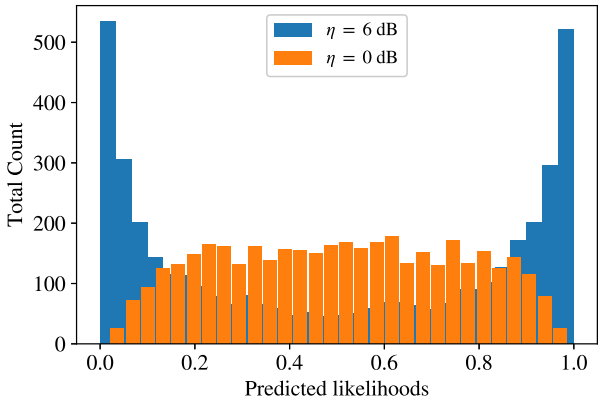


Fig. 5: Histograms of the estimated likelihoods for the ML - Soft scheme using RZF precoding with $N_{\mathrm{t}} = 15$, $N_{\mathrm{r}} = 8$, $\eta \in \{0, 6\}$ dB, a frame size of 2000, and QPSK modulation.

of 0 dB, we observe that the predicted likelihoods are spread around 0.5, which indicates a poor prediction performance. However, for $\eta = 6$ dB, the prediction accuracy is high as evidenced by the likelihoods values distributed mostly around values 0 and 1, thus demonstrating the efficacy of the adopted likelihood estimation method.

V. Numerical Results

Herein, we demonstrate the performance of the proposed detection frameworks using Monte Carlo simulations. We consider a Rayleigh flat-fading MIMO system with $N_{\mathrm{t}} = 15$ and $N_{\mathrm{r}} = 8$, with QPSK constellation and channel coding, in which we use convolutional coding [48] and Viterbi decoding [41] with the parameters in Table II. Unless otherwise specified, we use a frame size of 300 symbols, $T_{\mathrm{p}}^{\mathrm{up}} = T_{\mathrm{d}}^{\mathrm{up}}$, $\rho_{\mathrm{p}} = \rho_{\mathrm{d}}$, and $\sigma_z^2 = \alpha = 1$. Further, we consider CISPM and RZF precoding.

To our knowledge, we do not have direct competitors that addressed the same problem that we are investigating here. However, we compare our results with a *conventional* Rx that directly detects the received signals without any processing, since in the considered system, the Tx uses precoding to mitigate the channel effect. Subsequently, we refer to "Conv - Soft" to indicate soft decoding and "Conv - Hard" for the hard decoding. We also compare with another type of Rx that uses

TABLE II: Channel coding parameters used for the simulations

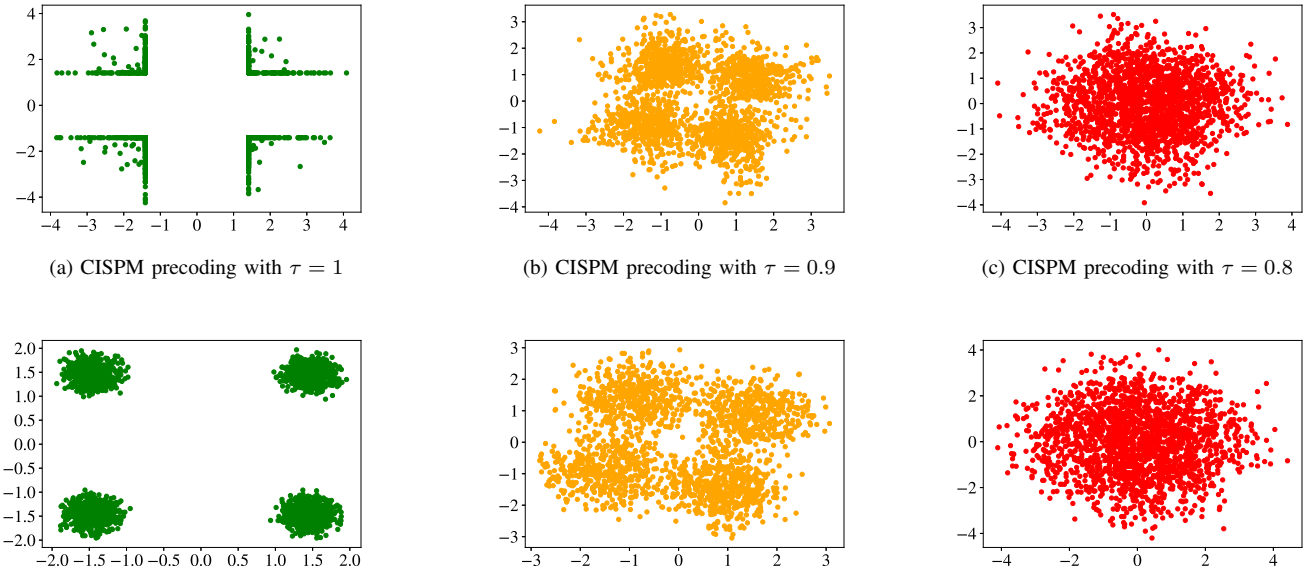
Parameters	**Values**
Code rate	½
Decoder decision technique	Hard, Soft
Number of frames	100
Trace-back length	96
Constraint length	9

the downlink precoded pilot signals to estimate the effective channel and equalize the receive data signals accordingly. Specifically, for this benchmark Rx, we adopt the least-squares (LS) method for the effective channel estimation and the linear MMSE detector that applies the SNR-regularized channel's pseudo-inverse and rounds the output to the nearest constellation point. In the following, for this Rx, we refer to "MMSE - Soft" to indicate soft decoding and "MMSE - Hard" for the hard decoding implementation.

In this work we consider several severity levels of CSIT degradation by varying the parameter τ in eq. (1). We note that a value of $\tau = 0.99$ identifies an optimistic channel imperfection, where even with such a small error in the CSI estimation, the degradation is notable [27]. In this work, we consider three scenarios of CSIT imperfections: $\tau = 1$ for perfect CSIT, $\tau = 0.9$ to indicate a moderately degraded CSIT estimate, and $\tau = 0.8$ to reflect severe CSIT degradation. We first analyze the noiseless received signal in each of these scenarios considering both RZF and CISPM precoding schemes.

Fig. 6 depicts the noiseless received signal at an Rx's antenna when using both RZF and CISPM precoding for different values of τ. We start by evaluating the case when the Tx uses CISPM precoding. Fig. 6a plots the scatterplot of the noiseless received signal in the case of perfect CSIT ($\tau = 1$). We can clearly see that the received constellation points are positioned in the corresponding detection regions; this scheme is guaranteeing a minimum target SINR value of $\gamma = 6$ dB for some transmitted symbol while propelling the rest deeper into the detection region. Therefore, we can conclude that the Tx was fully able to mitigate the ICI effect when using perfect CSIT. In the case of a degraded CSIT ($\tau = 0.9$), as depicted in Fig. 6b, we can observe that the constellation points that were mostly lying at the edge of the guaranteed SINR in Fig. 6a got deviated in all directions,

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/OJCOMS.2021.3079643, IEEE Open Journal of the Communications Society



(a) CISPM precoding with $\tau = 1$ (b) CISPM precoding with $\tau = 0.9$ (c) CISPM precoding with $\tau = 0.8$

(d) MMSE precoding with $\tau = 1$ (e) MMSE precoding with $\tau = 0.9$ (f) MMSE precoding with $\tau = 0.8$

Fig. 6: Noiseless received signal an Rx's antenna with $N_{\mathrm{t}} = 15$, $N_{\mathrm{r}} = 8$, and $\eta = \gamma = 6$ dB.

which is evidenced by the elliptic/circular shape cloud, due to the CSIT imperfections. However, in the case of severe CSIT imperfections ($\tau = 0.8$), as depicted in Fig. 6c, we can no longer straightforwardly map the received constellation points to the QPSK detection regions. This is due to the ICI effect.

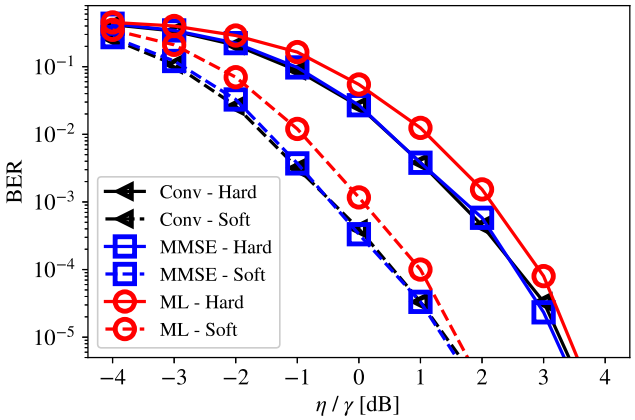
Similarly, when the Tx uses RZF precoding, we observe a similar phenomenon. Particularly, in the case of perfect CSIT ($\tau = 1$), as depicted in Fig. 6d, the noiseless received constellation also exhibits a clear separation between the 4 QPSK symbols. As opposed to ZF precoding, RZF does not fully cancel out the ICI effect, we see the constellation points forming a circular cloud in each detection region, where the cloud-effect is caused by the ICI effect, thus, this is how RZF offers a trade-off between ZF and MF precoding. Fig. 6e shows the degraded CSIT case, where we can observe that the CSIT imperfections led to deviations of the constellation points, causing the 4 "clouds" to get bigger and where their edge is near the detection regions. This means after noise adds up, some of the constellation points will cross into the opposite regions leading to detection errors. However, in the case of severe CSIT degradation ($\tau = 0.8$), the "clouds" are fully merged and centered at coordinates $(0, 0)$. The Tx in this case is not able to mitigate the ICI effect.

Fig. 7 depicts the BER as a function of η/γ [dB] in low SNR regime in the case of perfect CSIT ($\tau = 1$). Fig. 7a plots the BER as a function of η/γ. As expected, the higher the η/γ, the lower the BER, thus the better the detection performance. Conventional detectors outperform the proposed learning-based detectors, where both converge as η/γ increases. This is due to the fact that learning-based schemes loose some performance because of the training and the inference phases. The MMSE detector, however, achieves the same detection performance as the conventional detector, as the Tx have already mitigated all the ICI effect possible. As expected, for all detectors, soft decoding always outperforms hard decoding. The BER performance when using RZF precoding is depicted in Fig. 7b, where the same observations apply to the RZF case as well. Overall, we note that all detectors can achieve very low BER with very low η/γ, which indicates the effectiveness of all detectors in an unrealistic perfect CSI scenario, conventional, MMSE, and the proposed ones.

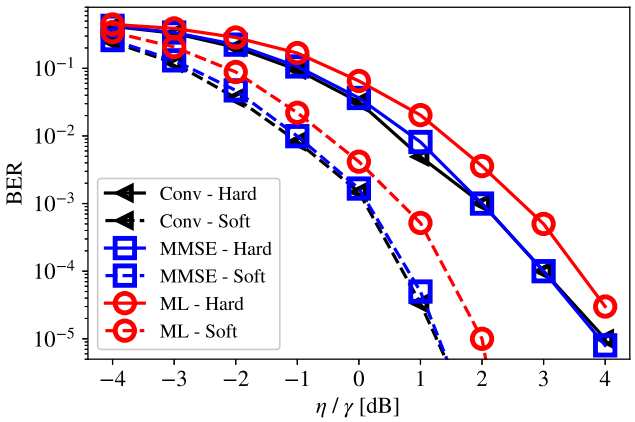
Fig. 8 plots the BER as a function of η/γ [dB] in the case of a moderately degraded CSIT ($\tau = 0.9$). In particular, Fig. 8a plots the BER as a function of η/γ in the case of CISPM precoding. Similarly, for all detectors, the higher the η/γ, the lower the BER. In particular, the conventional detector's performance flattens in high SNR, which is due to the ICI effect as depicted in Fig. 6b. Nonetheless, MMSE and the proposed detectors' BER decreases linearly with η/γ with the proposed ML detectors outperforming MMSE ones in high SNR. And as expected, for all detectors, soft decoding always outperforms hard decoding. When the Tx uses RZF precoding, as depicted in Fig. 8b, we observe a similar behavior. In particular, the proposed ML detectors outperform the MMSE detector as η/γ increase. This implies that the ML approach can leverage better the knowledge of the pilot symbols η/γ, contrary to the MMSE detector where the detection performance that improves linearly with the increase of η/γ.

Fig. 9 plots the BER as a function of η/γ [dB] in the case of severe CSIT degradation ($\tau = 0.8$). In particular, Fig. 9a plots the BER as a function of η/γ in the case of CISPM precoding. As expected, the BER when using the conventional detector is high, even in high SNR regime. However, the BER gets slightly lower as η/γ increases, but flattens out in high SNR. The MMSE detector's BER, on the other hand, improves linearly with η/γ but attaining a limited detection performance in high SNR. Nonetheless, when using the proposed learning-

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/OJCOMS.2021.3079643, IEEE Open Journal of the Communications Society

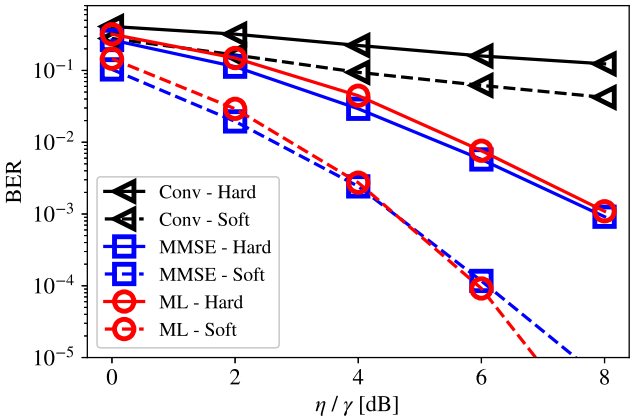


(a) CISPM precoding ($\tau = 1$)

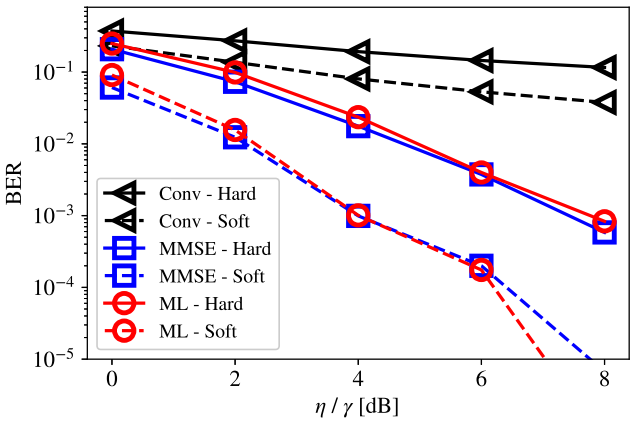


(b) RZF precoding ($\tau = 1$)

Fig. 7: BER vs. η/γ [dB], with $\tau = 1$ (perfect CSIT).



(a) CISPM precoding ($\tau = 0.9$)



(b) RZF precoding ($\tau = 0.9$)

Fig. 8: BER vs. η/γ [dB], with $\tau = 0.9$ (degraded CSIT).

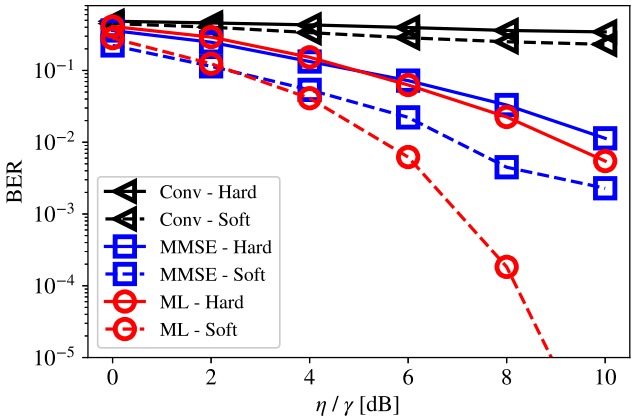
based detectors, the obtained BER drastically decreases as η/γ increases. This is due to the learning aspect of the proposed ML-based detectors; even in high ICI effect (induced by severe CSIT degradation), the ML detectors learns from the sent precoded pilots the input-output relationships and use it effectively in the inference. When the Tx uses RZF precoding, similar to Fig. 9a, for all detection schemes, the higher the η/γ, the lower the BER, with the proposed learning-based detectors immensely outperforming the conventional and MMSE detectors and soft decoding achieving better detection performance than hard decoding. In addition, severe CSIT degradation impede the correct detection when using conventional detectors even in high SNR regime. Nonetheless, the proposed learning-based detectors can achieve under 10^{-4} BER values with η/γ as low as 8 dB. Overall, we observe that RZF precoding leads to better detection performance than CISPM precoding.

To quantitatively evaluate the time complexity of the different detectors, in Fig. 10 we plot the runtime per SP as a function of N_{r} with $\eta = 6$ dB and RZF precoding at the Tx. As expected, the higher the number of antennas at the Rx, N_{r}, the higher the runtime, with soft detection consuming more computation time than hard detection for all schemes. We also observe that "Conv" and "MMSE" detectors exhibit a comparable runtime, as both are based on closed-form implementations. The runtime of the "ML - Hard" scheme is higher than the "Conv/MMSE" hard detection ones but shows a similar performance to "Conv/MMSE" soft detection methods. For the "ML - Soft" scheme, when $N_{\mathrm{r}} = 1$, it achieves the same runtime as the soft implementation of 'Conv/MMSE" schemes, whereas for $N_{\mathrm{r}} > 1$, the runtime of the "ML - Soft" is higher than the one for the soft implementation of "Conv/MMSE" schemes. This difference is due to the "CC" implementation of the "ML - Soft" scheme, where the training and inference phases of the SLC modules are performed sequentially for each received antenna/stream. Overall, the proposed ML detectors consumes marginally higher runtime than the closed-form based implementations.

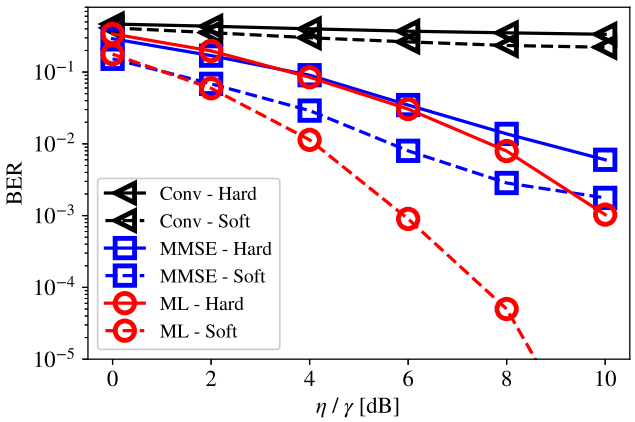
We conclude this section by summarizing the insights from the numerical results as follows:

- Soft decoding scheme always outperform hard decoding, i.e., soft values provide extra information to the decoder that allow for better restitution of the original data.
- In the case of perfect CSIT, RZF and CISPM precoding schemes are almost equivalent in performance, however, RZF clearly outperforms CISPM in imperfect CSIT

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/OJCOMS.2021.3079643, IEEE Open Journal of the Communications Society



(a) CISPM precoding ($\tau = 0.8$)



(b) RZF precoding ($\tau = 0.8$)

Fig. 9: BER vs. η/γ [dB], with $\tau = 0.8$ (severe CSIT degradation).

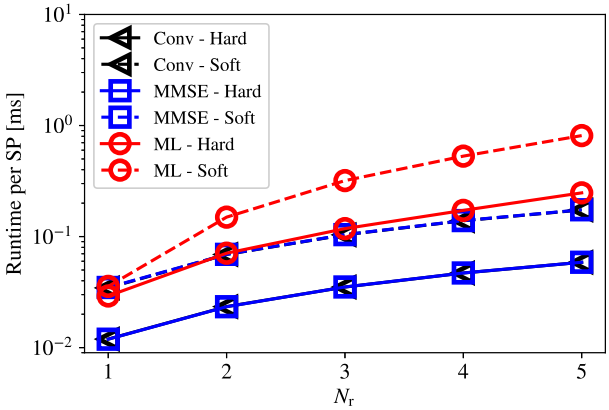


Fig. 10: Runtime per SP [ms] vs. N_r using RZF precoding and $\eta = 6$ dB.

regime.

- Conventional and MMSE detectors outperform the proposed learning based detectors in perfect CSIT, however, the former detectors are vulnerable to CSIT imperfections.
- The proposed learning-based detectors are much more robust to CSIT imperfections, thanks to their learning aspect that exploits the availability of the precoded SINR pilots in the downlink.
- Leveraging these precoded pilot symbols by using a classic detector like MMSE leads to limited BER improvement with SNR increase, as opposed to drastic BER improvement when using the proposed learning-based detectors.
- Overall, the proposed learning-based detectors achieve remarkable detection performance in severe channel conditions while having low computational complexity.

VI. Conclusions

In this paper, we investigated the impact of CSIT imperfections on coded MIMO detection in systems with precoding and without explicit CSIR knowledge. We modeled the CSIT imperfections using the Markov-Gauss formulation [27] to reflect the degradation due to channel estimation errors. In this setting, we proposed soft and hard learning-based detection frameworks that leverage the availability of downlink precoded pilots, originally intended for SINR estimation, as training data. Moreover, we proposed a lightweight implementation by using fast and efficient ML algorithms and methods to support online training. Numerical results showed that CSIT imperfections inhibits correct detection when using a conventional MIMO detector. We showed that even when a conventional Rx exploits the downlink precoded pilots to estimate the effective channel and uses the MMSE detector to compensate for ICI, the resulting performance is not good under severe CSIT degradation. However, the proposed learning-based detectors are substantially more robust to CSIT degradation, where the proposed ML scheme can achieve 4-8 dB power gain for a BER value under 10^{-4} when compared to the MMSE receiver under severe CSIT degradation, while retaining low computational complexity.

References

[1] Taesang Yoo, Eunchul Yoon, and A. Goldsmith, "MIMO capacity with channel uncertainty: does feedback help?" in *in Proc. IEEE Global Comm. Conf. (GLOBECOM)*, vol. 1, 2004, pp. 96–100 Vol.1.

[2] S. Yang and L. Hanzo, "Fifty years of MIMO detection: The road to large-scale MIMOs," *IEEE Commun. Surveys Tuts.*, vol. 17, no. 4, pp. 1941–1988, 2015.

[3] D. Gesbert, S. Hanly, H. Huang, S. Shamai Shitz, O. Simeone, and W. Yu, "Multi-cell MIMO cooperative networks: A new look at interference," *IEEE J. Sel. Areas in Commun.*, vol. 28, no. 9, pp. 1380–1408, 2010.

[4] E. G. Larsson, "MIMO detection methods: How they work [lecture notes]," *IEEE Signal Process. Mag.*, vol. 26, no. 3, pp. 91–95, 2009.

[5] Y. Sun, M. Peng, Y. Zhou, Y. Huang, and S. Mao, "Application of machine learning in wireless networks: Key techniques and open issues," *IEEE Commun. Surveys Tuts.*, vol. 21, no. 4, pp. 3072–3108, 2019.

[6] C. Jiang, H. Zhang, Y. Ren, Z. Han, K. Chen, and L. Hanzo, "Machine learning paradigms for next-generation wireless networks," *IEEE Wireless Commun.*, vol. 24, no. 2, pp. 98–105, Apr. 2017.

[7] F. A. Aoudia and J. Hoydis, "End-to-end learning for OFDM: From neural receivers to pilotless communication," vol. arXiv preprint arXiv:2009.05261, 2020. [Online]. Available: http://arxiv.org/abs/2009.05261

[8] T. O'Shea and J. Hoydis, "An introduction to deep learning for the physical layer," *IEEE Trans. Cognitive Commun. Netw.*, vol. 3, no. 4, pp. 563–575, 2017.

[9] N. Samuel, T. Diskin, and A. Wiesel, "Learning to detect," *IEEE Trans. Signal Process.*, vol. 67, no. 10, pp. 2554–2564, 2019.

[10] H. He, C.-K. Wen, S. Jin, and G. Y. Li, "A model-driven deep learning network for MIMO detection," vol. arXiv preprint arXiv:1809.09336, 2018. [Online]. Available: http://arxiv.org/abs/1809.09336

[11] M. Khani, M. Alizadeh, J. Hoydis, and P. Fleming, "Adaptive neural signal detection for massive MIMO," *IEEE Trans. Wireless Commun.*, vol. 19, no. 8, pp. 5635–5648, 2020.

[12] J. Zhang, Y. He, Y. W. Li, C. K. Wen, and S. Jin, "Meta learning-based MIMO detectors: Design, simulation, and experimental test," *IEEE Trans. Wireless Commun.*, vol. 20, no. 2, pp. 1122–1137, 2021.

[13] O. Sholev, H. H. Permuter, E. Ben-Dror, and W. Liang, "Neural network MIMO detection for coded wireless communication with impairments," in *2020 IEEE Wireless Commun. Netw. Conf. (WCNC)*, 2020, pp. 1–8.

[14] J. Ma and L. Ping, "Orthogonal AMP," *IEEE Access*, vol. 5, pp. 2020–2033, 2017.

[15] H. Liang, W. Chung, and S. Kuo, "Coding-aided K-Means clustering blind transceiver for space shift keying MIMO systems," *IEEE Trans. Wireless Commun.*, vol. 15, no. 1, pp. 103–115, 2016.

[16] L. You, P. Yang, Y. Xiao, S. Rong, D. Ke, and S. Li, "Blind detection for spatial modulation systems based on clustering," *IEEE Commun. Letters*, vol. 21, no. 11, pp. 2392–2395, 2017.

[17] Y. Jeon, S. Hong, and N. Lee, "Blind detection for MIMO systems with low-resolution ADCs using supervised learning," in *2017 IEEE Int. Conf. Commun. (ICC)*, 2017, pp. 1–6.

[18] L. V. Nguyen, D. T. Ngo, N. H. Tran, A. L. Swindlehurst, and D. H. N. Nguyen, "Supervised and semi-supervised learning for MIMO blind detection with low-resolution ADCs," *IEEE Trans. Wireless Commun.*, vol. 19, no. 4, pp. 2427–2442, 2020.

[19] G. Thiagarajan and C. R. Murthy, "Novel transmit precoding methods for Rayleigh fading multiuser TDD-MIMO systems with CSIT and no CSIR," *IEEE Trans. Vehicular Tech.*, vol. 64, no. 3, pp. 973–984, 2015.

[20] B. N. Bharath and C. R. Murthy, "Channel training signal design for reciprocal multiple antenna systems with beamforming," *IEEE Trans. Vehicular Tech.*, vol. 62, no. 1, pp. 140–151, 2013.

[21] T. L. Marzetta and B. M. Hochwald, "Fast transfer of channel state information in wireless systems," *IEEE Trans. Signal Process.*, vol. 54, no. 4, pp. 1268–1278, 2006.

[22] M. Guillaud, D. T. M. Slock, and R. Knopp, "A practical method for wireless channel reciprocity exploitation through relative calibration," in *In Proc. 8th Int. Symp. Signal Process. Appl., 2005.*, vol. 1, 2005, pp. 403–406.

[23] B. Ottersten, "Array processing for wireless communications," in *In Proc. 8th Workshop Stat. Signal Array Process.*, 1996, pp. 466–473.

[24] A. Mayouche, W. A. Martins, C. G. Tsinos, S. Chatzinotas, and B. Ottersten, "A novel learning-based hard decoding scheme and symbol-level precoding countermeasures," in *2021 IEEE Wireless Commun. Netw. Conf. (WCNC)*, 2021.

[25] ——, "Multi-antenna data-driven eavesdropping attacks and symbol-level precoding countermeasures," vol. arXiv preprint arXiv:2011.03750, 2020. [Online]. Available: http://arxiv.org/abs/2011.03750

[26] T. Schenk, *RF Imperfections in High-rate Wireless Systems*, 1st ed. Springer Netherlands, 2008.

[27] M. Rebato, L. Rose, and M. Zorzi, "Performance assessment of MIMO precoding on realistic mmWave channels," in *IEEE Int. Conf. Commun. Workshops (ICC Workshops)*, 2019, pp. 1–6.

[28] E. Alberty, S. Defever, C. Moreau, R. De Gaudenzi, A. Ginesi, R. Rinaldo, G. Gallinaro, and A. Vernucci, "Adaptive coding and modulation for the DVB-S2 standard interactive applications: Capacity assessment and key system issues," *IEEE Wireless Commun.*, vol. 14, no. 4, pp. 61–69, 2007.

[29] M. Ding and S. D. Blostein, "MIMO minimum total MSE transceiver design with imperfect CSI at both ends," *IEEE Trans. Signal Proces.*, vol. 57, no. 3, pp. 1141–1150, 2009.

[30] C. B. Peel, B. M. Hochwald, and A. L. Swindlehurst, "A vector-perturbation technique for near-capacity multiantenna multiuser communication-part i: channel inversion and regularization," *IEEE Trans. Commun.*, vol. 53, no. 1, pp. 195–202, 2005.

[31] M. Alodeh, S. Chatzinotas, and B. Ottersten, "Constructive Multiuser Interference in Symbol Level Precoding for the MISO Downlink Channel," *IEEE Trans. Signal Process.*, vol. 63, no. 9, pp. 2239–2252, May 2015.

[32] L. Rose and M. Maso, "Receiver-centric inter-cell interference cancellation in D2D-assisted networks," in *IEEE Globecom Workshops (GC Wkshps)*, 2016, pp. 1–7.

[33] A. Li, D. Spano, J. Krivochiza, S. Domouchtsidis, C. G. Tsinos, C. Masouros, S. Chatzinotas, Y. Li, B. Vucetic, and B. Ottersten, "A tutorial on interference exploitation via symbol-level precoding: Overview, state-of-the-art and future directions," *IEEE Commun. Surveys Tuts.*, vol. 22, no. 2, pp. 796–839, Secondquarter 2020.

[34] M. Alodeh, D. Spano, A. Kalantari, C. G. Tsinos, D. Christopoulos, S. Chatzinotas, and B. Ottersten, "Symbol-level and multicast precoding for multiuser multiantenna downlink: A state-of-the-art, classification, and challenges," *IEEE Commun. Surveys Tuts.*, vol. 20, no. 3, pp. 1733–1757, thirdquarter 2018.

[35] M. Alodeh, S. Chatzinotas, and B. Ottersten, "Symbol-level multiuser MISO precoding for multi-level adaptive modulation," in *IEEE Trans. Wireless Commun.*, vol. 16, no. 8, Aug 2017, pp. 5511–5524.

[36] J. Krivochiza, J. Merlano Duncan, S. Andrenacci, S. Chatzinotas, and B. Ottersten, "FPGA acceleration for computationally efficient symbol-level precoding in multi-user multi-antenna communication systems," *IEEE Access*, vol. 7, pp. 15 509–15 520, 2019.

[37] G. Tsoumakas and I. Katakis, "Multi-label classification: An overview," *Int. J. Data Warehousing Mining (IJDWM)*, vol. 3, no. 3, pp. 1–13, July 2007.

[38] A. Burkov, *The Hundred-Page Machine Learning Book*. Andriy Burkov, 2019.

[39] R. Gallager, *Chapter 8, course materials for 6.450 Principles of Digital Communications I*. MIT OpenCourseWare, Fall 2006.

[40] B. Zadrozny and C. Elkan, "Transforming classifier scores into accurate multiclass probability estimates," *in Proc. 8th ACM SIGKDD Int. Conf. Knowledge Discovery Data Mining*, July 2002.

[41] A. J. Viterbi, "A personal history of the Viterbi algorithm," *IEEE Signal Process. Magazine*, vol. 23, no. 4, pp. 120–142, July 2006.

[42] C. G. Tsinos, A. Kalantari, S. Chatzinotas, and B. Ottersten, "Symbol-level precoding with low resolution dacs for large-scale array MU-MIMO systems," in *IEEE Int. Workshop Signal Process. Adv. Wireless Commun. (SPAWC)*, 2018, pp. 1–5.

[43] L. Deng, "A tutorial survey of architectures, algorithms, and applications for deep learning," *APSIPA Trans. Signal Inf. Process.*, vol. 3, Jan 2014.

[44] M.-L. Zhang, Y.-K. Li, X.-Y. Liu, and X. Geng, "Binary relevance for multi-label learning: an overview," *Springer Front. Comput. Sci.*, vol. 12, no. 1, p. 191–202, Jan 2018.

[45] Z. Yu, Q. Wang, Y. Fan, H. Dai, and M. Qiu, "An improved classifier chain algorithm for multi-label classification of big data analysis," *IEEE 17th Int. Conf. High Performance Comput. Commun., IEEE 7th Int. Symposium Cyberspace Safety Security, 12th Int. Conf. Embedded Softw. and Syst.*, pp. 1298–1301, 2015.

[46] J. Read, B. Pfahringer, G. Holmes, and E. Frank, "Classifier chains for multi-label classification," in *Machine Learning and Knowledge Discovery in Databases*. Springer Berlin Heidelberg, 2009, pp. 254–269.

[47] J. C. Platt, "Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods," in *Advances in Large Margin Classifiers*. MIT Press, 1999, pp. 61–74.

[48] R. Johannesson and K. S. Zigangirov, *Fundamentals of Convolutional Coding*. Wiley-IEEE Press, 2015.