Neural Network Aided Computation of Mutual Information for Adaptation of Spatial Modulation

Anxo Tato*, Carlos Mosquera*, Pol Henarejos†, Ana Pérez-Neira†‡

* atlanTTic Research Center, Universidade de Vigo, Galicia, Spain

† Centre Tecnològic de Telecomunicacions de Catalunya (CTTC), Castelldefels, Spain

‡ Dept. of Signal Theory and Communications, Universitat Politècnica de Catalunya (TSC-UPC)

Email: {anxotato, mosquera}@gts.uvigo.es, {pol.henarejos, ana.perez}@cttc.cat

Abstract

Index Modulations, in the form of Spatial Modulation or Polarized Modulation, are gaining traction for both satellite and terrestrial next generation communication systems. Adaptive Spatial Modulation based links are needed to fully exploit the transmission capacity of time-variant channels. The adaptation of code and/or modulation requires a real-time evaluation of the channel achievable rates. Some existing results in the literature present a computational complexity which scales quadratically with the number of transmit antennas and the constellation order. Moreover, the accuracy of these approximations is low and it can lead to wrong Modulation and Coding Scheme selection. In this work we apply a Multilayer Feedforward Neural Network to compute the achievable rate of a generic Index Modulation link. The case of two antennas/polarizations is analyzed in depth, showing not only a one-hundred fold decrement of the Mean Square Error in the estimation of the capacity as compared with existing analytical approximations, but also a fifty times reduction of the computational complexity. Moreover, the extension to an arbitrary number of antennas is explained and supported with simulations. More generally, neural networks can be considered as promising candidates for the practical estimation of complex metrics in communication related settings.

Index Terms

Mutual Information, Capacity, Index Modulation, Spatial Modulation, Polarized Modulation, Neural Networks, MFNN, Machine Learning, Adaptive Communications, ACM, Link Adaptation.

I. INTRODUCTION

The evaluation of the achievable physical layer rate of a given modulation scheme is an important theoretical problem with high relevance in practice. Most modern communication standards, e.g., [1], [2] or [3], incorporate some sort of Adaptive Coding and Modulation (ACM) mechanism, generically known as link adaptation. This consists typically on varying the modulation order and/or the coding rate of the channel encoder to track the varying channel conditions. The ultimate goal is to adjust the transmitted bit rate to the information that the channel can support for a given bit error probability.

Link adaptation makes it necessary for the transmitter to estimate somehow the mutual information (MI) between the transmit and received waveforms on a per-frame basis, so that the most efficient Modulation and Coding Scheme (MCS) can be chosen. In most cases, the receiver computes some metric related to the MI and sends it back to the transmitter end. This metric can be in the form of the average or effective Signal to Interference and Noise Ratio (SINR), or some Channel Quality Indicator (CQI) specifically suited to the set of MCS available to the transmitter [4]. In essence, the receiver must estimate the maximum amount of information that can be transmitted reliably through the channel; for all this, the estimation of the MI plays an instrumental role.

A particular family of modulation schemes, known as Index Modulations (IM) [5], have attracted a great deal of interest in the last few years. Among others, we can cite Spatial Modulation (SM) [6]-[7] or Polarized Modulation (PMod) [8]. SM and its more sophisticated variants are proposed for next generation of wireless networks due to several advantages. In comparison to single-antenna techniques, the spectral efficiency increases, with a simpler and more energy efficiency hardware than in other multi-antenna techniques [5]. As potential application, recent works such as [9] present experimental results with compact reconfigurable antennas for using SM in the uplink of Internet of Things (IoT) devices in 5G networks with a high number of antennas on

the base station side. Another interesting version of IM is that studied in [8], where the authors propose the use of PMod to increase the spectral efficiency of next generation mobile satellite communications; if Multiple-Input-Multiple-Output (MIMO) signal processing techniques are applied to Dual Polarization (DP) satellite systems, the performance of single-antenna (or single polarization) links can be notably enhanced. DP schemes were also highlighted in [10] as a means to improve the satellite coverage in remote areas to serve the increasing number of IoT devices.

In this paper we present a novel method to compute the mutual information without Channel State Information at the Transmitter (CSIT) of a 2×2 SM system, and show how to generalize it to an arbitrary number of antennas. The results are also valid for other types of IM, like PMod. The study is conducted for conventional SM, with only one Radio Frequency (RF) chain at the transmitter and a unique active antenna at each time instant. If more antennas can simultaneously transmit, the corresponding scheme is known as Generalized SM (GSM), which can achieve a higher spectral efficiency. An extension of the results in this paper to GSM is addressed in [11], where we have included results for transmitters with up to three RF chains.

The mutual information calculation can be useful, for example, in adaptive SM systems to select a proper MCS. Results requiring numerical integration or Monte Carlo simulations can be found in the literature [12], [13]. One value of this work is that it explores a radically new approach to solve an essential problem in the practical application of Information Theory: the mutual information of non-conventional modulations is computed by means of Machine Learning (ML) tools.

There is a growing number of contributions on the application of ML to the physical layer of communications, see the recent surveys [14] and [15]. In particular, Neural Networks (NN) have been successfully used for channel estimation and equalization [16], signal recognition and modulation classification [17], [18], detection in MIMO Generalized SM [19], and learning of physical layer parameters in Cognitive Radio [20], among others. In [21] and [22] NNs are applied to perform link adaptation in multicarrier systems. In [23], a Multilayer Feedforward Neural Network (MFNN) is used to predict the performance of a WiFi cell. A deep NN is proposed in [24] to decide the optimal power allocation in a wireless resource management problem. In this latter reference the NN is used to obtain the optimal power allocation values,

much more efficiently (by speeding up the computational time in several orders of magnitude) than the baseline iterative algorithm which solves the corresponding non-convex optimization problem. Besides NNs, Support Vector Machines (SVM) have been also studied for the selection of physical layer parameters in communication settings with a large number of degrees of freedom [25].

Recently, some results on the application of ML to SM systems have appeared. For example, [26] proposes the use of deep neural network to perform power allocation and transmit antenna selection on a SM system. In [27], a deep neural network is used to select the optimum codebook in adaptive SM, i.e., the particular constellation employed by each antenna. This recent work is related with older SM publications which deal with adaptive SM systems, such as [28] and [29]. The method we propose here for computing the mutual information of a SM link is a perfect complement of the referred previous schemes, which use a target spectral efficiency which needs to be estimated somehow. The neural network aided mutual information calculation enables the online update of the target bit rate so that other algorithms can select the optimum codebook.

The current work applies a one-hidden layer MFNN as a facilitator scheme to compute the MI in an adaptive 2×2 SM link, based on some specifically selected input features which can be easily obtained from the MIMO channel matrix, together with the Signal to Noise Ratio (SNR). To the best of our knowledge, it is the first time that a NN is proposed to estimate the MI of a channel. In the particular scenario of SM, the evaluation of the capacity, needed for adaptation purposes, is numerically demanding when needed on-the-fly.

The shallow NN proposed to calculate the MI of a generic SM system, valid also for PMod, outperforms recent approximations found in the literature such as [13] and [30], both in terms of estimation accuracy and computational complexity. In order to avoid the numerical evaluation of the involved integrals, these references provide two different approximations of the MI for a specific symbol constellation, with a complexity scaling with the square of the constellation size and the number of antennas. As opposed, the proposed solution has a much lower complexity, which is independent of the size of the constellation.

The main contribution of this work is the accurate evaluation of the MI of spatial modulations, which have resisted so far those attempts to obtain simple expressions. Moderate size standard neural networks will be seen to be up to the task provided that a careful extraction of channel

parameters and training are performed. Moreover, due to the lower complexity of the proposed method, the receiver can compute the MI more often and then follow faster channel variations, so that the adaptation speed is not necessarily limited by the complexity of the computation of the channel capacity.

The rest of the paper is structured as follows. Section II explains our system model and introduces the reader into SM; Section III provides some additional motivation by presenting an example of an adaptive SM system which can be improved with the evaluation of MI. Then, Section IV presents the expressions to compute the mutual information of SM. It also replicates the analytical expressions existing in the literature to approximate the MI, and to be used for benchmarking purposes. Afterwards, in Section V a brief introduction to Multilayer Feedforward Neural Networks is included before dealing with their specific application to the evaluation of the MI of a 2×2 SM for different constellations. Section VI presents the simulation results in detail for the case of two dimensions. Then, Section VII explains how to generalize the method to obtain the MI of systems with a higher number of antennas. Lastly, the main conclusions are drawn in Section VIII.

Notation: Upper (lower) boldface letters denote matrices (vectors). $(\cdot)^H$, $(\cdot)^t$, $\mathbf{I}_N$ and $\mathbf{1}$, denote Hermitian transpose, transpose, $N \times N$ identity matrix and vector of ones, respectively. $\|\cdot\|$ applied to vectors denotes the Euclidean norm. $\mathbb{E}[\cdot]$ is the expected value operator. $\circ$ and $\oslash$ denote the Hadamard (pointwise) matrix product and division. $\Re\{\cdot\}$, $\Im\{\cdot\}$, $(.)^*$ and $|\cdot|$ denote the real part, imaginary part, conjugate and absolute value of a complex number, respectively.

II. System model

Traditional digital modulation schemes transmit information modulating only the amplitude, phase and/or frequency of a sinusoidal carrier. However, Index Modulations (IM) benefit from the fact that the transmitter has several building blocks, being these antennas, polarizations or subcarriers, for example, to map additional bits of information to the block selected to transmit the conventional modulated signal [31]. As illustration, consider SM with N_t transmit antennas: in addition to the $\log_2(M)$ bits to index each symbol s in a constellation of M elements, $\log_2(N_t)$ bits can be used to select which of the N_t antennas is active at a given instant to transmit the symbol. Similarly, PMod carries information by means of the transmit polarization. In this paper,

firstly we will focus our attention in the $N_t = 2$ case, with just one bit selecting the active transmit dimension; the extension to $N_t > 2$ is also addressed later. Hereafter, SM is used rather than IM, keeping in mind that results apply to generic IM regardless of the physical interpretation of the dimensions, which can be antennas, polarization, or frequencies among others.

The system model of a 2×2 SM for a given discrete time instant is

$$\mathbf{y} = \sqrt{\gamma}\mathbf{H}\mathbf{x} + \mathbf{w} \tag{1}$$

where $\mathbf{y} \in \mathbb{C}^{2\times 1}$ is the received vector, γ the average Signal to Noise Ratio (SNR), $\mathbf{H} \in \mathbb{C}^{2\times 2}$ the channel matrix, $\mathbf{x} \in \mathbb{C}^{2\times 1}$ the transmitted signal and $\mathbf{w} \sim \mathcal{CN}(\mathbf{0}, \mathbf{I}_2)$ the Additive White Gaussian (AWGN) noise vector. Since we consider a SM system with a single RF chain, $\mathbf{x}$ has only one component $s \in \mathbb{C}$ different from zero (component l), so that (1) can be also expressed as

$$\mathbf{y} = \sqrt{\gamma}\mathbf{h}_l s + \mathbf{w} \tag{2}$$

where $\mathbf{h}_l$ denotes the l-th column of $\mathbf{H}$, $l \in \{1, 2\}$, and s takes values from the constellation $\mathcal{S}$, which is assumed the same for all the antennas when it comes to the evaluation of MI. We also assume a unit power constraint, i.e., $\mathbb{E}\left[\mathbf{x}^H\mathbf{x}\right] = \mathbb{E}\left[|s^2|\right] = 1$.

This paper addresses the evaluation of the MI in an SM link for which the transmit symbol s comes from the same constellation $\mathcal{S}$ regardless of the transmit antenna. This value represents an upper bound for the maximum rate which can be transmitted reliably; in this regard, our focus is not on error metrics, which will depend on the specific encoding scheme and receiver implementation, but rather on information quantities which will be estimated by the neural network.

Fig. 1 shows the block diagram of a fully flexible adaptive SM system with several degrees of freedom in the link adaptation. On the one hand, the transmitter can modify the coding rate r of the channel encoder for adapting the level of protection of the information bits. On the other hand, the transmitter can also select individually the constellation order M_l of the symbols sent by each antenna l, as proposed in [28]. The receiver, assumed to have perfect Channel State Information (CSI), estimates the SNR γ and the channel matrix $\mathbf{H}$; both are used by the Adaptation Unit to select a physical layer configuration. Then, the receiver informs the transmitter

about the specific coding rate r and codebook $\mathcal{C}$ to use in subsequent transmissions. Following [27], codebook refers to the set of constellations used by each of the transmit antennas[1].

Before delving into the evaluation of mutual information and channel transmission capacity, next section motivates the need for these metrics in adaptive schemes. By means of an example, we illustrate how our evaluation of the MI can complement some of the existing adaptation schemes proposed in the literature.

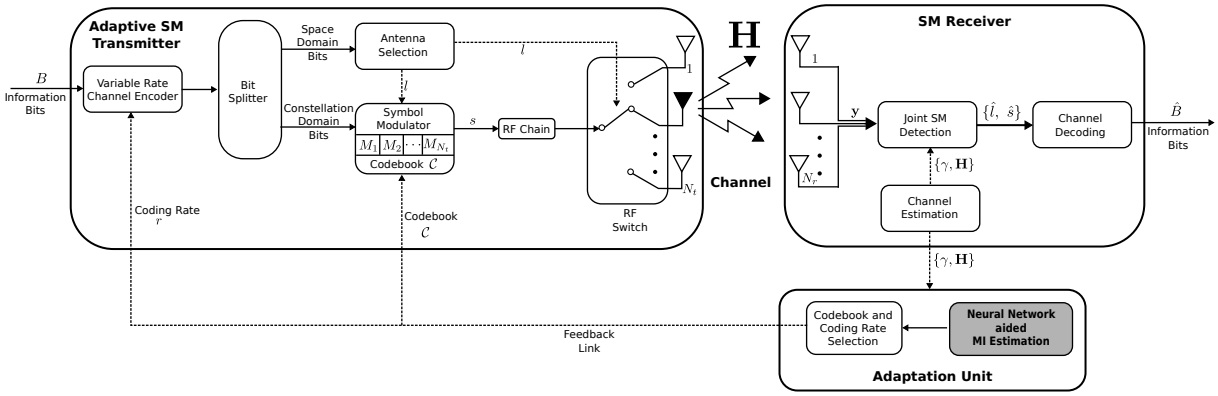


Fig. 1: Block diagram of an adaptive Spatial Modulation system with Neural Network aided MI calculation at the receiver.

III. Adaptive Spatial Modulation

SM offers many degrees of freedom that can potentially be exploited for the adaptation of the physical layer parameters in response to the channel conditions. Yang et al. propose in [28] the individual tuning of the constellation order of the symbols sent from each antenna. Similarly, works such as [27] use the term SM codebook, $\mathcal{C}$, to denote the set of the constellations associated to each antenna. In this way, for a system with N_t antennas, the codebook $\mathcal{C} = \{\mathbf{\Omega}_1, \ldots, \mathbf{\Omega}_{N_t}\}$ indicates that the constellation $\mathbf{\Omega}_i$ is used by the i-th transmit antenna.

The receiver, which is assumed to have full CSI, is typically responsible for choosing the optimum codebook $\mathcal{C}^*$ and feeding this information back to the adaptive SM transmitter. In [28] and [29],

[1]For the evaluation of the mutual information and channel capacity, we will consider that all the antennas are to pick symbols from the same set $\mathcal{S}$ at a given channel use. The channel capacity can be used then as an upper bound for the achievable rate also for those cases for which the constellation to use depends on the selected antenna.

the authors select the codebook based on the received minimum distance $d_{\min}(\mathbf{H}, \mathcal{C})$ of each codebook $\mathcal{C}$, which is defined as

$$d_{\min}(\mathbf{H}, \mathcal{C}) = \min_{\mathbf{x}_i, \mathbf{x}_j, \mathbf{x}_i \neq \mathbf{x}_j} \|\mathbf{H}(\mathbf{x}_i - \mathbf{x}_j)\|^2. \tag{3}$$

They use the minimum distance as aapproximated performance metric to the bit error probability. $\mathbf{x}_i$ and $\mathbf{x}_j$ in (3) are vectors with a single non-zero entry, which denote all possible transmit signals for the codebook $\mathcal{C}$. Note that the distance depends also on the channel matrix $\mathbf{H}$. The optimum $\mathcal{C}^*$ is chosen to maximize the minimum distance among all the codebooks which provide a given target spectral efficiency m:

$$\mathcal{C}^* = \arg\max_{\mathcal{C} \in \mathbf{\Psi}} d_{\min}(\mathbf{H}, \mathcal{C}) \text{ subject to } \mathcal{C} \text{ with } \frac{1}{N_t} \sum_{k=1}^{N_t} m_k = m. \tag{4}$$

In (4) $\mathbf{\Psi}$ denotes the set of all possible codebooks, and m_k is the spectral efficiency of the constellation $\mathbf{\Omega}_k$ used in the antenna k with the codebook $\mathcal{C}$.

Link adaptation schemes as those detailed in [27] and [28] require as input the target spectral efficiency, the parameter m in (4). However, they do not specify how to set the value of m that rules the codebook selection. The method that we present later in this paper allows to compute the maximum achievable spectral efficiency C in an affordable way, given the involved complexity in the evaluation of the transmission capacity of spatial modulation. Then, from the values of spectral efficiency attainable with the set of codebooks $\mathbf{\Psi}$ of the system, we propose to select the closest value to the calculated capacity C and use it as the target spectral efficiency m in (4).

For illustration purposes, we show some preliminary simulations which make use of the method exposed later in this paper to evaluate the maximum transmission capacity for each channel realization. In particular, the adaptive SM link includes a single RF chain transmitter which uses 1, 2 or 4 antennas to transmit the spatial information and a receiver with 4 antennas. The available constellations range from BPSK to 16QAM, with the codebook selection performed as in [28], except that the parameter m is computed dynamically for each channel realization from the maximum transmission capacity of the eight different configurations: SISO (one antenna), the six possible 2×4 SM systems and a 4×4 SM system.

Fig. 2 shows the results of the simulations described in the previous paragraphs. 200 Rayleigh distributed channel matrices are generated for each value of SNR and, for each particular channel

state, the transmission of $5,000$ symbols is simulated. The optimal codebook for each particular channel state, i.e., for a fixed $(\gamma, \mathbf{H})$, is chosen with (4) and used during $5,000$ transmissions. One application of the results in this paper is the ability of adapting the target spectral efficiency to the channel state. Thus, for the 200 simulated matrices, curves in Fig. 2 show the minimum, maximum and average values of the target spectral efficiency m, respectively, for each SNR. The average throughput, computed taking into account the symbol error probability of each simulation, is also shown. As the channel conditions improve, the target efficiency increases until it reaches a maximum value of 6 bits/s/Hz, which is obtained by using the 16QAM by all the antennas.

The simulation results in Fig. 2 illustrate how the evaluation of the SM capacity, as described in the next sections, can complement existing SM link adaptation algorithms for selecting the codebook, like those introduced in [27] and [28]. Our estimation of the SM capacity allows to update the target spectral efficiency m and select codebooks that provide higher data rates as the capacity of the channel increases, as can be seen in Fig. 2. Next section will include the expressions to compute the capacity and the MI of a SM system by means of Monte Carlo simulations. Then, Section V will detail how a MFNN can be used to obtain the MI of SM with high accuracy and low complexity.

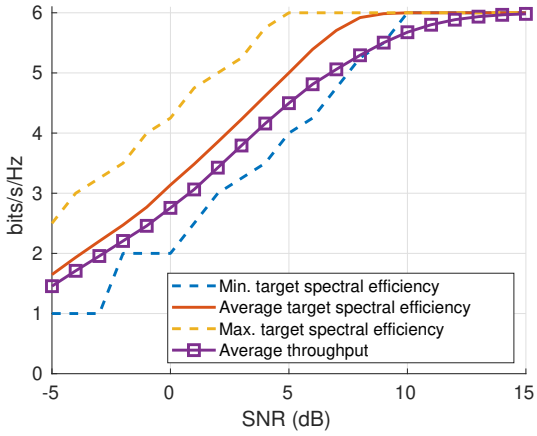


Fig. 2: Average throughput as compared with the target spectral efficiency reference values in an adaptive SM system. The codebook selection is based on the application of the channel capacity to set the target spectral efficiency.

IV. Capacity and Mutual Information

The expression of the Spatial Modulation (SM) capacity conditioned to a given realization of the channel matrix $\mathbf{H}$, measured in bits per channel use, *bpcu*, is given by

$$C = \max_{f_X(x)} I(\mathbf{x};\ \mathbf{y}\ |\mathbf{H}) = \max_{f_S(s), f_L(l)} I(s, l;\ \mathbf{y}\ |\mathbf{H})\ [bpcu], \tag{5}$$

where $I(\mathbf{x}; \mathbf{y}|\mathbf{H})$ is the Mutual Information (MI) between the two random variables $\mathbf{x}$ and $\mathbf{y}$ conditioned to $\mathbf{H}$, and the maximization is performed for all possible distributions of the transmitted signal $\mathbf{x}$ [32]. In (5), $f_X(x)$, $f_S(s)$ and $f_L(l)$ denote the probability density functions (PDF) of the complex transmit signal $\mathbf{x}$, the complex transmit symbol s, and the hopping index l which selects the antenna used to transmit the symbols, respectively. The transmitter is expected to operate without CSIT (it does know neither $\mathbf{H}$ nor γ), so it will select either index $l = \{1, 2\}$ with the same probability. The capacity is achieved in (5) when the transmit symbols belong to a Gaussian codebook [12], i.e., when $s \sim \mathcal{CN}(0, 1)$.

The MI in (5) can be expressed, as a function of the entropies of the involved random variables, as

$$C = I(\mathbf{x};\ \mathbf{y}\ |\mathbf{H}) = h(\mathbf{y}|\mathbf{H}) - h(\mathbf{y}|\mathbf{x}, \mathbf{H}) = h(\mathbf{y}|\mathbf{H}) - h(\mathbf{w}), \tag{6}$$

where $h(\cdot)$ is used for the differential entropy, and $h(\mathbf{w})$ is simply written as

$$h(\mathbf{w}) = \log_2 \det(\pi e \mathbf{I}_2). \tag{7}$$

As in [12], the received vector $\mathbf{y}$ follows a Gaussian distribution of the form

$$\mathbf{y} \sim \frac{1}{2}\sum_{l=1}^{2} \mathcal{CN}\left(\mathbf{0},\ \mathbf{H}\mathbf{K}_l\mathbf{H}^H + \mathbf{I}_2\right) \triangleq \frac{1}{2}\sum_{l=1}^{2} \mathcal{CN}\left(\mathbf{0},\ \mathbf{\Phi}_l\right), \tag{8}$$

where $\mathbf{K}_1 = \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix}$ and $\mathbf{K}_2 = \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix}$. With this, the entropy of $\mathbf{y}$ in (6) reads as

$$h(\mathbf{y}|\mathbf{H}) = -\frac{1}{2}\sum_{l=1}^{2} \int_{\mathbf{y}} \mathcal{CN}(\mathbf{0}, \mathbf{\Phi}_l) \log_2 \left(\frac{1}{2}\sum_{l'=1}^{2} \mathcal{CN}(\mathbf{0}, \mathbf{\Phi}_{l'})\right). \tag{9}$$

It is then clear that the computation of the unconstrained capacity C of (6) requires the numerical evaluation of the above integral; even for the case of constrained capacity with finite alphabets, as we will see next, the complexity of this computation can be too demanding for a receiver updating the estimate of the link capacity for adaptation purposes.

Practical communication links use symbols from a constellation $\mathcal{S}$ with a finite alphabet, which makes the constrained capacity the practical metric of interest. Hereafter, we will refer to the corresponding mutual information, or constrained capacity of SM, simply as total mutual information, I_T, since this includes the information carried by both antenna selection index l and constellation symbol s:

$$I_T = I(\mathbf{x};\ \mathbf{y}\ |\mathbf{H})|_{s\in\mathcal{S}} = I(s,l;\ \mathbf{y}\ |\mathbf{H})|_{s\in\mathcal{S}}\,. \tag{10}$$

The particularization of (10) for a 2×2 SM system and a constellation $\mathcal{S}$ with M symbols has been made in [13], and it is replicated in (11) for the sake of completeness. Note that we assume that the same constellation $\mathcal{S}$ is used in all the antennas.

The evaluation of (11) to obtain the MI can be made by means of numerical integration or by resorting to Monte Carlo simulations. The latter entail a much lower complexity since the number of function evaluations required for a given accuracy in high-dimensional integrals increases exponentially with the number of dimensions (receive antennas) [33]. Thus, the Monte Carlo scheme turns out to be more appropriate than traditional numerical integration for the evaluation of expressions such as (11).

In this article we will use Monte Carlo integration to compute the *true* value of (11) for reference purposes, by generating random values of the noise $\mathbf{w}$. The number of noise samples is chosen to keep the variance of the Monte Carlo estimation of I_T under 10^{-5}, which was achieved in all cases with $5,000$ samples.

$$I_T = \log_2(2M) - \frac{1}{2M}\sum_{s\in\mathcal{S}}\sum_{l=1}^{2}\mathbb{E}_{\mathbf{W}}\left\{\log_2\left(\sum_{s'\in\mathcal{S}}\sum_{l'=1}^{2} e^{-\gamma\left\|\mathbf{h}_l s-\mathbf{h}_{l'}s'+\frac{\mathbf{w}}{\sqrt{\gamma}}\right\|^2+\|\mathbf{w}\|^2}\right)\right\} \tag{11}$$

In an effort to find more convenient expressions of I_T to handle in practice, some results have been presented in the literature as approximations to the mutual information (11). On the one side, Guo et al [30] used the Jensen's inequality and corrected the ensuing bias to get

$$I_{T_{\text{Jensen}}} = -\log_2\left(\frac{\sum_{\Delta_x\in\mathcal{D}} e^{-\frac{1}{2}\Delta_x^H\mathbf{H}^H\mathbf{H}\Delta_x}}{(2M)^2}\right). \tag{12}$$

Here $\mathcal{D}$ is a set with $(2M)^2$ vectors in $\mathbb{C}^{2\times 1}$ of the form

$$\Delta_x = \sqrt{\gamma} \cdot (\mathbf{h}_l s_k - \mathbf{h}_{l'} s_{k'}) \tag{13}$$

for $l, l' = 1, 2$ and $k, k' = 1, 2, \ldots, M$, where $\mathbf{h}_l$ are the columns of the channel matrix, and $s_k \in \mathcal{S} \subset \mathbb{C}$ the symbols of the constellation.

A different approach resorting to the Taylor Series Expansion was followed in [13], yielding expression (14) as an approximation of I_T. The interested reader is referred to [13] for the definitions of each element of the equation. One drawback of both (12) and (14) is that the computational complexity of the MI calculation increases with the square of the constellation order M and the number of antennas N_t.

$$I_{T_{Taylor}} = \log_2\left(\frac{2M}{\mathfrak{G}(\mathcal{D}_{sl})}\right) + \mathfrak{A}\left(\frac{\log_2\left(\mathfrak{G}_{sl}\left(D_{s,l,s',l'}{}^{D_{s,l,s',l'}}\right)\right)}{\mathfrak{A}_{sl}\left(D_{s,l,s',l'}\right)} + \frac{\gamma}{\log(2)\mathcal{D}_{sl}^2}\sum_{m=1}^{2}\left(\mathcal{D}_{m,sl,\mathfrak{R}}^2 + \mathcal{D}_{m,sl,\mathfrak{I}}^2\right)\right) \tag{14}$$

This paper develops a more efficient and accurate scheme, based on a neural network, to compute the MI of an SM system, which avoids the quadratic complexity increment with the constellation cardinality. This is especially relevant for practical use, given the need to estimate on the fly the channel capacity for link adaptation purposes. The proposed scheme to estimate the achievable rate is based on a simple NN with only one hidden layer, and which provides different outputs, one per constellation in case several are available. The computational burden is much lower than that of any other previously known alternatives, so the MI can be updated more often and faster variations of the channel conditions can be tracked as a benefit.

In [11], we have presented a particular application of this work to obtain the unconstrained capacity, i.e., for Gaussian symbols, of a GSM system. In GSM, contrary to SM, there can be multiple active antennas simultaneously. The unconstrained capacity calculated in [11] is an upper bound of the MI for any given constellation. It turns out that SM is usually proposed for the uplink of IoT devices with complexity constraints [9], so that we will provide in this work simulations for constellations up to order 16. However, if higher dense constellations such as 64QAM or 256QAM are used, results from [11] can be applied to evaluate the unconstrained capacity instead.

V. Neural Network-based Mutual Information Estimation

The evaluation of the Mutual Information (MI) (11) can be interpreted as a non-linear mapping from the channel matrix $\mathbf{H}$ and the SNR γ to the MI. Multilayer Feedforward Neural Networks (MFNNs), well-known for their fitting capabilities of non-linear functions [34]-[35], will be used to estimate I_T in (11). In particular, the MFNN to be employed, a one hidden layer network, is detailed in Fig. 3. The input features, $\mathbf{x} = [x_1, x_2, \ldots, x_F]^t$, need to be extracted by means of a function $f(\cdot)$ from the channel matrix $\mathbf{H}$ and the SNR γ, as $\mathbf{x} = f(\gamma, \mathbf{H})$. The input variable selection is highly relevant for the performance of the learning process of the network. Later we will detail how the *feature extraction* is applied based on our *domain knowledge*, that is, our knowledge of the particular problem we are addressing. Alternatively, a more complex deep neural network with several hidden layers could be used, so that the relevant features for this problem are learned by the intermediate layers, and all the channel matrix coefficients are used directly as inputs without any processing. However, as detailed later, we found this solution underperforming the one hidden layer with carefully selected input features.

In the following, the variables in blue will denote the internal parameters of the neural network, and $\mathbf{a}_i, i = 0, 1, 2$ will be the intermediate internal variables at the i-th layer. Each of the F neural network inputs goes through a linear preprocessing block to adjust the neurons input to the range $[-1, +1]$. This initial scaling is expressed as

$$\mathbf{a}_0 = \mathbf{g}_0 \circ (\mathbf{x} - \mathbf{x}_0) - \mathbf{1} \ \in \mathbb{R}^{F\times 1} \tag{15}$$

with the gain $\mathbf{g}_0 \in \mathbb{R}^{F\times 1}$ and the offset $\mathbf{x}_0 \in \mathbb{R}^{F\times 1}$.

The hidden layer is made of N neurons, also named processing units, each applying a weighted linear combination of its inputs, a bias and a non-linear function, also known as activation function:

$$\mathbf{a}_1 = g(\mathbf{W}_1 \cdot \mathbf{a}_0 + \mathbf{b}_1) \in \mathbb{R}^{N\times 1}. \tag{16}$$

The matrix $\mathbf{W}_1 \in \mathbb{R}^{N\times F}$ and the vector $\mathbf{b}_1 \in \mathbb{R}^{N\times 1}$ collect the weights and the offsets. As activation function we will use the hyperbolic tangent:

$$g(x) = \frac{2}{1 + e^{-2x}} - 1.$$

The output layer of K neurons applies a linear processing of the form

$$\mathbf{a}_2 = \mathbf{W}_2 \cdot \mathbf{a}_1 + \mathbf{b}_2 \ \in \mathbb{R}^{K\times 1} \tag{17}$$

for matrix $\mathbf{W}_2 \in \mathbb{R}^{K\times N}$ and vector $\mathbf{b}_2 \in \mathbb{R}^{K\times 1}$. Finally, there is a last stage to accommodate the range of the network outcome:

$$\mathbf{y} = (\mathbf{a}_2 + \mathbf{1}) \oslash \mathbf{g}_3 + \mathbf{y}_0 \ \in \mathbb{R}^{K\times 1} \tag{18}$$

with the gain $\mathbf{g}_3 \in \mathbb{R}^{K\times 1}$ and the offset $\mathbf{y}_0 \in \mathbb{R}^{K\times 1}$. The output vector is expressed as $\mathbf{y} = [y_1, y_2, \ldots, y_K]^t$.

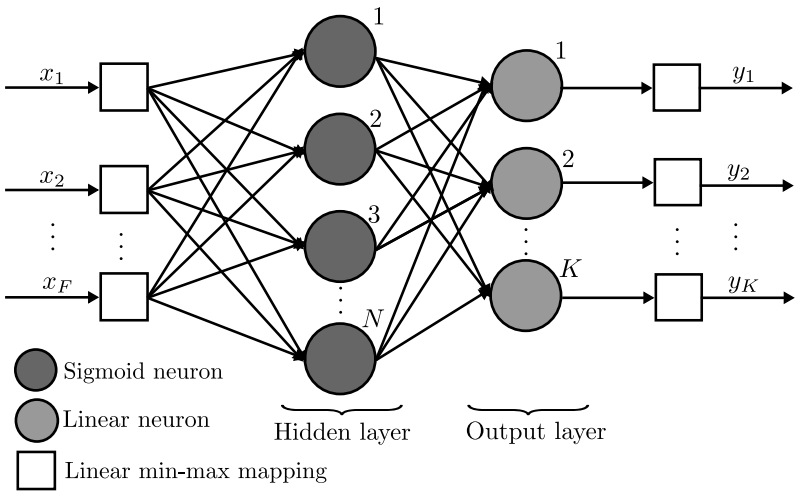


Fig. 3: Diagram of the neural network.

The different parameters of the network –weights and biases in (15)-(18)– will be obtained under supervised learning. They are initialized with the Nguyen-Widrow algorithm [36], which generates random values for the weights and the biases for a balanced distribution of input values across the neurons. The Levenberg-Marquardt (LM) backprogation algorithm [37] will be used to minimize the Mean Squared Error (MSE) on the test set:

$$\text{MSE} = \frac{1}{L}\sum_{\ell=1}^{L} \|\mathbf{y}(\ell) - \mathbf{t}(\ell)\|^2, \tag{19}$$

with $\mathbf{y}(\ell)$ and $\mathbf{t}(\ell)$ the network output and the true MI values, respectively, for the training tuples $(\mathbf{x}(\ell), \mathbf{t}(\ell))$, $\ell = 1, \ldots, L$. The true MI values are obtained by numerical evaluation of (11) for K different symbol constellations. The training of the network is performed off-line, so that

receiver terminals only need to evaluate (15)-(18) for adaptation purposes. The computational complexity of the NN will be evaluated later in comparison to that of previous methods presented in the literature [13]-[30].

Hereafter, we focus on the MI calculation of a 2×2 SM system and next section will provide simulation results for this particular case. However, the underlying philosophy applies to SM with a higher number of antennas, not only 2. Section VII will show how to extend the results to obtain the MI of SM systems with a larger number of antennas. Remarkably, the neural network entails always a degree of complexity about two orders of magnitude lower compared to that of the analytical approximations, even for SM systems with high number of antennas/dimensions.

A. Input Variable Selection

The output of the network in Fig. 3 is the MI estimation for K different constellations, as an approximation of the true MI function (11). This depends on the channel matrix $\mathbf{H}$ and the SNR γ; in the following we will see how to pre-process these values for a better network performance.

For a Maximum Likelihood receiver, the pairwise error probability (PEP) between (s, l) and $(\hat{s}, \hat{l})$ is given by [38],[39]

$$P_e(s, \hat{s}, l, \hat{l}) = Q\left(\sqrt{\frac{\gamma}{2}} \|\mathbf{h}_l \cdot s - \mathbf{h}_{\hat{l}} \cdot \hat{s}\|\right) \tag{20}$$

for the AWGN case. This PEP depends on the distance among supersymbols $\mathbf{h}_l \cdot s$. The set $\mathcal{X}$ of $2M$ different supersymbols for a given channel matrix $\mathbf{H}$ and the transmitted constellation $\mathcal{S} = \{s_k,\ k = 1, 2, \ldots, M\}$ is

$$\mathcal{X} = \{\mathbf{h}_l s_k,\ l = 1, 2,\ k = 1, 2, \ldots, M\}. \tag{21}$$

The MI will be also affected by all the involved distances, as inferred from (11). For convenience, we put together the squared distances among all the pairs of supersymbols under the matrix $\mathbf{D}$ of size $2M \times 2M$, with the respective entries given by

$$\mathbf{D}[(l-1)M + k, (l'-1)M + k'] = \|\mathbf{h}_l s_k - \mathbf{h}_{l'} s_{k'}\|^2 \tag{22}$$

where $l, l' = \{1, 2\}$ and $k, k' = \{1, 2, \ldots, M\}$. Matrix $\mathbf{D}$ can be also expressed as

$$\mathbf{D} = \begin{pmatrix} \|\mathbf{h}_1\|^2 \mathbf{D}_S & \mathbf{D}_L \\ \mathbf{D}_L^t & \|\mathbf{h}_2\|^2 \mathbf{D}_S \end{pmatrix}. \tag{23}$$

The $M \times M$ matrix $\mathbf{D}_S$ on the diagonal is a function of the symbols in the constellation $\mathcal{S}$:

$$\mathbf{D}_S[k,k'] = |s_k - s_{k'}|^2, \tag{24}$$

whereas the $M \times M$ matrix $\mathbf{D}_L$ contains all the distances between supersymbols of different antennas/polarizations:

$$\begin{aligned} \mathbf{D}_L[m,n] &= \|\mathbf{h}_1 s_m - \mathbf{h}_2 s_n\|^2 \\ &= \|\mathbf{h}_1 s_m\|^2 + \|\mathbf{h}_2 s_n\|^2 - 2\Re\{s_m^* s_n \mathbf{h}_1^H \mathbf{h}_2\}. \end{aligned} \tag{25}$$

Note that the matrix $\mathbf{D}_L$ can be expressed as the sum of four rank-1 matrices:

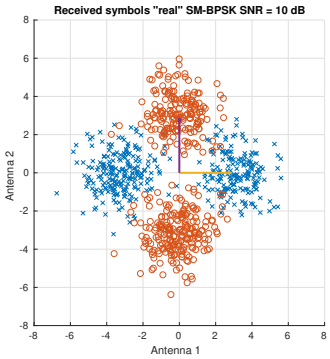
$$\begin{aligned} \mathbf{D}_L = \|\mathbf{h}_1\|^2 \begin{pmatrix} |s_1|^2 \\ \vdots \\ |s_M|^2 \end{pmatrix} \mathbf{1}^t + \|\mathbf{h}_2\|^2 \, \mathbf{1} \begin{pmatrix} |s_1|^2 & \dots & |s_M|^2 \end{pmatrix} \\ - \mathbf{h}_1^H \mathbf{h}_2 \begin{pmatrix} s_1^* \\ \vdots \\ s_M^* \end{pmatrix} \begin{pmatrix} s_1 & \dots & s_M \end{pmatrix} \\ - \mathbf{h}_1^t \mathbf{h}_2^* \begin{pmatrix} s_1 \\ \vdots \\ s_M \end{pmatrix} \begin{pmatrix} s_1^* & \dots & s_M^* \end{pmatrix}. \end{aligned} \tag{26}$$

With this, we have that rank$\{\mathbf{D}_L\} \leq 4$. Even further, if the constellation of symbols $\{s_n\}$ is known, then only four real values are required to describe the dependence of $\mathbf{D}_L$ and $\mathbf{D}$ with the channel matrix $\mathbf{H}$, namely, $\|\mathbf{h}_1\|^2, \|\mathbf{h}_2\|^2$ and the real and imaginary parts of the scalar product $\mathbf{h}_1^H \mathbf{h}_2$. Alternatively, the scalar product can be expressed as [40]

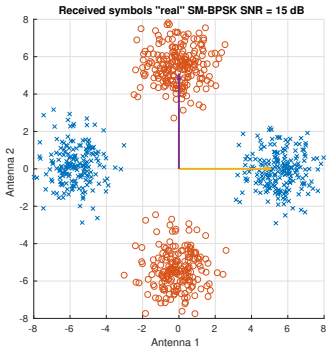
$$\mathbf{h}_1^H \mathbf{h}_2 = \|\mathbf{h}_1\| \cdot \|\mathbf{h}_2\| \cdot \cos\Theta_H \cdot e^{i\varphi} \tag{27}$$

where $\Theta_H \in [0, \pi/2]$ and $\varphi \in [-\pi, \pi]$ denote, respectively, the Hermitian angle and the Kasner's pseudo-angle between two complex vectors. Thus, the four values $(\|\mathbf{h}_1\|, \|\mathbf{h}_2\|, \Theta_H, \varphi)$ serve to characterize the matrix $\mathbf{D}$.

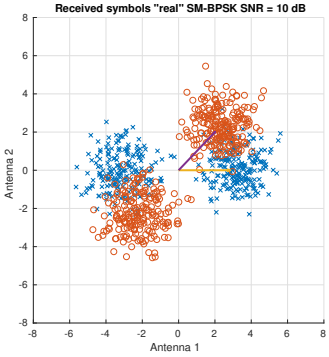
For illustration purposes, Fig. 4 shows the received symbols for a real BPSK case. Two different SNR values and two different channel matrices are employed to display the clouds of received symbols. Both the SNR and the angle between the column vectors of $\mathbf{H}$ determine the distance among the different color clouds.



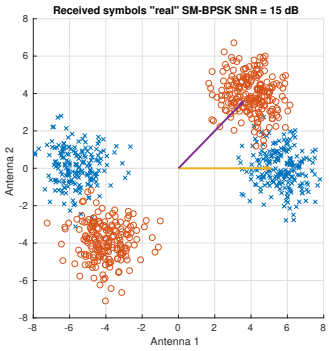
(a) Orthogonal columns, SNR= 10 dB.



(b) Orthogonal columns, SNR= 15 dB.



(c) Non-orthogonal columns, SNR= 10 dB.



(d) Non-orthogonal columns, SNR= 15 dB.

Fig. 4: Received constellation for 2×2 SM-BPSK system where transmitted symbols, channel matrix and noise are real-valued.

The impact of the two angles Θ_H and φ in the final MI can be grasped with the aid of Fig. 5, which shows a 3D representation of the MI as function of both angles. Monte Carlo simulations were run for a QPSK constellation, with both columns having the same norm, $\|\mathbf{h}_1\| = \|\mathbf{h}_2\| = 1$, and $\gamma = 2$. With this, the structure of the channel matrix $\mathbf{H}$ is the following:

$$\mathbf{H} = \begin{pmatrix} 1 & \cos\Theta_H e^{i\varphi} \\ 0 & \sin\Theta_H \end{pmatrix}. \tag{28}$$

It can be seen that the MI has a strong dependance with the Hermitian angle. If $\Theta_H = \pi/2$ the two columns are orthogonal and the MI is maximum, whereas for $\Theta_H = 0$ the columns are considered parallel, and the MI is reduced. Moreover, the Kasner's pseudoangle φ only affects the MI significantly when Θ_H is close to zero, creating a small ripple, due to the radial

symmetry of the constellation. However, for $\Theta_H > \pi/3$, the MI is barely affected by the Kasner's pseudoangle. Instead, if $\Theta_H = 0$, the phase φ determines to which extent the receiver can tell which antenna transmitted the observed symbol.

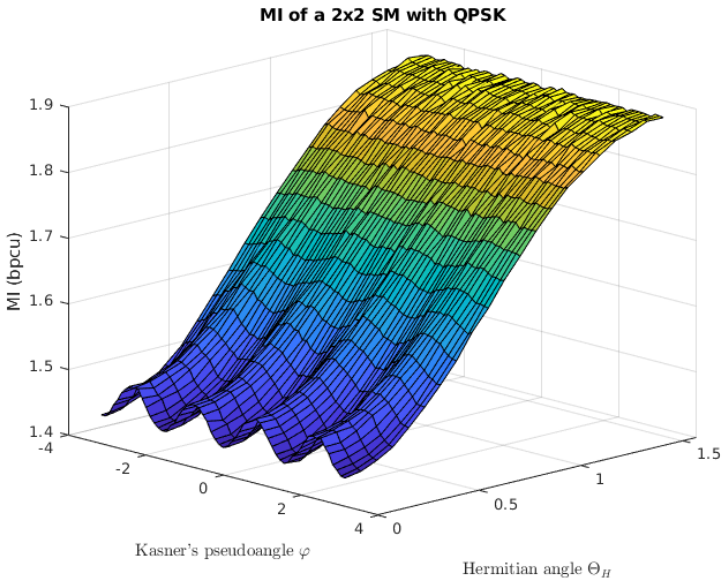


Fig. 5: MI of a 2×2 SM link with QPSK constellation as a function of the two angles for unit-valued columns norms and 3 dBs of SNR.

After extensive training cases, we have observed that performance can be enhanced if, as part of the input parameters, the distances among the supersymbols are also included. Four distances, $\{d_i\}, i = 1, \ldots, 4$, are used; this is the number of different entries of matrix $\mathbf{D}_L$ in (25) when a QPSK constellation is employed. It turns out that these four quantities suffice for the neural network to compute a good estimate of the MI for other constellations too, such as 8PSK and 16QAM, even though the number of different entries of the matrix is higher. Table III in Section VI supports this claim by showing the performance of the neural network when estimating the MI for 8PSK and 16QAM, in addition to QPSK, without exploiting additional distance values, thus keeping the number of input features fixed.

B. Neural Network Operation

We will use a unique MFNN to estimate the mutual information I_T for different symbol constellations. $K = 3$ outputs will provide the estimate of the mutual information for QPSK, 8PSK

and 16QAM constellations. Following the above considerations, the input features $\mathbf{x} = f(\gamma, \mathbf{H})$ will correspond to different options to characterize the distance matrix $\mathbf{D}$. Table I depicts the different feature sets that will be used for testing the performance of the network. Essentially, four inputs is the lowest number of inputs to test following the previous discussion[2]. Values are sorted in ascending order given the invariance of the capacity to the labeling of the dimensions and symbols. As to the number of neurons, $N = 10$ and $N = 20$ will be used in the simulations.

Option	**Input features** F	**Description of the features**
i	4	$\mathbf{x} = \left[\text{ sort } \left([\gamma \Vert \mathbf{h}_1 \Vert^2,\ \gamma \Vert \mathbf{h}_2 \Vert^2] \right),\ \Re \left\{ \frac{\mathbf{h}_1^H \mathbf{h}_2}{\Vert \mathbf{h}_1 \Vert \cdot \Vert \mathbf{h}_2 \Vert} \right\},\ \Im \left\{ \frac{\mathbf{h}_1^H \mathbf{h}_2}{\Vert \mathbf{h}_1 \Vert \cdot \Vert \mathbf{h}_2 \Vert} \right\} \right]^t$
ii	4	$\mathbf{x} = \left[\text{ sort } \left([\gamma \Vert \mathbf{h}_1 \Vert^2,\ \gamma \Vert \mathbf{h}_2 \Vert^2] \right),\ \Theta_H,\ \varphi \right]^t$
iii	6	$\mathbf{x} = \left[\text{ sort } \left([\gamma \Vert \mathbf{h}_1 \Vert^2,\ \gamma \Vert \mathbf{h}_2 \Vert^2] \right),\ \text{sort}([\gamma d_1,\ \gamma d_2,\ \gamma d_3,\ \gamma d_4]) \right]^t$
iv	8	$\mathbf{x} = \left[\text{ sort } \left([\gamma \Vert \mathbf{h}_1 \Vert^2,\ \gamma \Vert \mathbf{h}_2 \Vert^2] \right),\ \text{sort}([\gamma d_1,\ \gamma d_2,\ \gamma d_3,\ \gamma d_4]),\ \Re \left\{ \frac{\mathbf{h}_1^H \mathbf{h}_2}{\Vert \mathbf{h}_1 \Vert \cdot \Vert \mathbf{h}_2 \Vert} \right\},\ \Im \left\{ \frac{\mathbf{h}_1^H \mathbf{h}_2}{\Vert \mathbf{h}_1 \Vert \cdot \Vert \mathbf{h}_2 \Vert} \right\} \right]^t$
v	8	$\mathbf{x} = \left[\text{ sort } \left([\gamma \Vert \mathbf{h}_1 \Vert^2,\ \gamma \Vert \mathbf{h}_2 \Vert^2] \right),\ \text{sort}([\gamma d_1,\ \gamma d_2,\ \gamma d_3,\ \gamma d_4]),\ \Theta_H,\ \varphi \right]^t$

TABLE I: Different alternatives for selecting the NN input features.

Training of the network will be based on extensive amount of data, by generating a large number or random channels for different values of SNR. The reference true capacity values for the different constellations will be obtained after computing (11) by the Monte Carlo method.

VI. Simulation results

For performance evaluation, a dataset of $50,000$ realizations of the channel matrix $\mathbf{H}$ is used, randomly generated following a unit-variance Rayleigh distribution, i.e., $h_{ij} \sim \mathcal{CN}(0, 1)$. Each channel matrix is associated with a different SNR whose value in decibels is drawn from a uniform random variable between -20 and 20 dB. The true MI with QPSK, 8PSK and 16QAM constellations of each realization of $(\gamma, \mathbf{H})$ is calculated with a Monte Carlo simulation using (11) with $5,000$ realizations of the complex Gaussian noise $\mathbf{w}$. We limit ourselves to these low order constellations, which are more likely to be used with SM. However, other constellations, like 64QAM, could be easily added to the system at the expense of increasing the time required for obtaining the dataset with Monte Carlo simulations -note the two summations over all the constellation symbols in (11).

[2]Note that the number of real values to characterize $\mathbf{H}$ and γ is nine.

The dataset is divided into two independent parts. $7,500$ samples (15%) are reserved for the final test of the performance of the MFNN and the analytical approximations (equations (12 and (14)). The remaining $35,000$ samples (70%) and $7,500$ samples (15%) are employed for training and validation of the neural network, respectively.

Firstly, the impact of the selection of the input features in the performance of the MFNN is evaluated. For this, several NNs are trained using in each case one of the sets of inputs detailed in Table I. Fig. 6 shows some histograms with the statistical distribution of the features obtained from the dataset. The distances and the norms include the SNR term and are shown in dB, whilst the unit of the angles is the radian.

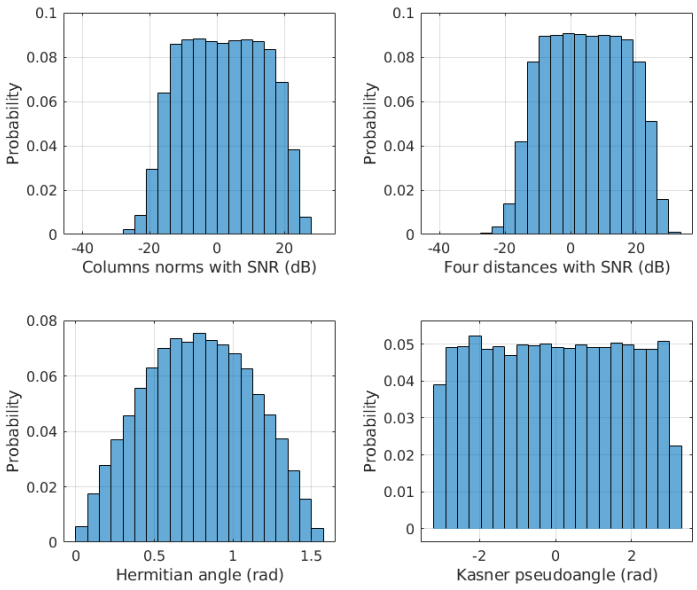


Fig. 6: Histograms with the distribution of the features (norms, distances and angles) in the dataset of Rayleigh distributed channel matrices and uniformly distributed SNR.

Then, the global MSE obtained with the trained NNs when calculating the three MI values is obtained by using the entries of the dataset reserved for testing. Table II collects the values of MSE for five different selections of the input features and for both numbers of neurons ($N = 10$ and $N = 20$). It shows the best MSE in the testing dataset after 10 trainings with different NN parameters initialization. If the NNs are fed directly with the real and imaginary parts of the channel matrix coefficients, the MSE is very high, in the order of 10^{-1}. Nevertheless, at least

two orders of magnitude improvement is achieved when the NNs are fed with the input features detailed in Table I.

Input features option	**Number of features F**	**Global MSE (10 neurons)**	**Global MSE (20 neurons)**
i) Columns norm and projection	4	$1.78 \cdot 10^{-3}$	$6.98 \cdot 10^{-4}$
ii) Columns norm and angles	4	$4.29 \cdot 10^{-4}$	$3.36 \cdot 10^{-4}$
iii) Columns norm and distances	6	$1.79 \cdot 10^{-4}$	$5.21 \cdot 10^{-5}$
iv) Columns norm, distances and projection	8	$1.33 \cdot 10^{-4}$	$4.96 \cdot 10^{-5}$
v) Columns norm, distances and angles	8	$1.00 \cdot 10^{-4}$	$2.97 \cdot 10^{-5}$

TABLE II: Comparison of the global MSE obtained with the neural network for different input features.

In Table II, it can be observed how the Hermitian angle Θ_H and the Kasner's pseudoangle φ, options (ii) and (v), improve the NN estimation with respect to the use of the real and imaginary parts of the projection, options (i) and (iv). Furthermore, the addition of the four distances to the set of inputs, cases (iii), (iv) and (v), serves to reduce the MSE as compared to cases (i) and (ii). Finally, the MSE reaches a minimum value of about $3 \cdot 10^{-5}$ when the four distances and the two column norms are combined with the two angles for the 20 neurons MFNN.

Secondly, the two NNs with 10 and 20 neurons and the input features selection (v), are compared with the analytical approximations from the literature, (12) and (14), in Table III. Both Taylor and Jensen based approximations have a similar MSE, around 10^{-2}, which is outperformed by all the NN reported in Table II. Moreover, when we compare the analytical approximations with the best NNs of the table, the improvement in the MSE is about 100 and 600 times with a NN of 10 and 20 neurons, respectively.

As noted previously, the calculation of the MI could be addressed with a deep neural network, a MFNN with several hidden layers, using as inputs the channel matrix coefficients (scaled by the SNR) directly. With this approach, the network extracts the relevant features at the intermediate layers, so that the last layer computes the MI. We have tested this approximation for a number of layers ranging from one to ten, and a number of neurons per layer between 20 and 50. It was found that a deep network with at least three layers of 20 neurons can perform better than the Taylor and Jenson approximations, yielding an MSE in the order of 10^{-3}. However, the deep networks do not overcome the peformance of the one-hidden layer MFNN which the input

	Global MSE	QPSK		8PSK		16QAM	
		3σ	Max. error	3σ	Max. error	3σ	Max. error
Taylor approximation (14)	$1.87 \cdot 10^{-2}$	0.330	0.523	0.370	0.492	0.392	0.558
Jensen based approximation (12)	$1.21 \cdot 10^{-2}$	0.229	0.300	0.291	0.498	0.300	0.741
MFNN option (v) with 10 neurons	$1.00 \cdot 10^{-4}$	0.020	0.153	0.026	0.140	0.034	0.120
MFNN option (v) with 20 neurons	$2.97 \cdot 10^{-5}$	0.016	0.067	0.015	0.046	0.018	0.105

TABLE III: Comparison of the performance of the MFNN with the analytical approximations of the literature for calculating the MI of a 2×2 SM.

features of row (v) of Table I. In addition, the training of these deep networks is much more time consuming, and the learning algorithm has more difficulties to converge to those parameter values providing a good performance.

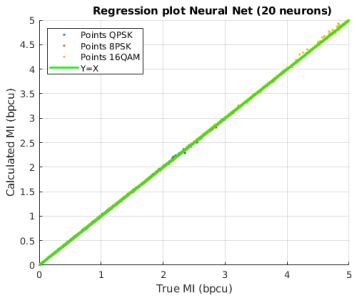
As shown in Table III, the analytical approximations suffer a maximum error of 0.741, for the 16-QAM constellation, which is reduced to 0.105 in the case of the MFNN with 20 neurons. The improvement is even more noticeable with the 3σ value: a little bit more than 0.300 for the analytical approximations, and 20 times smaller with the neural network. This significant accuracy improvement achieved by the MFNN makes it possible for the physical layer to operate with a transmission rate much closer to the channel transmission capacity. We should note that there is not much more room for improvement since the MSE with the MFNN of 20 neurons is of the same order of magnitude than the variance in the estimation of the MI with the Monte Carlo simulations, 10^{-5}.

Fig. 7 shows a graphical view of the estimated MI values: the scatter plot of the values of the true MI (X axis) are shown together with the values of the MI computed with each method (Y axis) for the three constellations, QPSK, 8PSK and 16QAM. The green line Y=X sets the perfect match of the MI. It can be seen that the analytical approximations provide better results for lower values of MI, while for MIs close to their maximum (3, 4 or 5, depending on the constellation), they have a noticeable positive bias. Remarkably, both NNs with 10 and 20 neurons in the hidden layer, match the true value of the MI almost perfectly, clearly outperforming the analytical approximations.

The accuracy achieved by the MFNN has a direct impact on the implementation of adaptive

(a) Taylor approximation (14).

(b) Jensen based approximation (12.)



(c) Neural network option (v), 20 neurons.

Fig. 7: Comparison of the scatter plots (true MI vs calculated MI) of the analytical approximation from the literature and the MFNNs with 20 neurons for the three constellations (QPSK, 8PSK and 16QAM) in a 2×2 SM system.

SM links. The quality of the tracking of the MI allows to use smaller back-off margins for the selection of the coding rate or the codebooks; large errors make it necessary to use highly conservative margins in the selection of the physical layer configuration to guarantee a prescribed error decoding metric, thus reducing the transmission rate.

Finally, the ergodic MI of an 2×2 SM system with QPSK, 8PSK and 16QAM constellations under Rayleigh fading is shown in Fig. 8. For each value of SNR, 100 realizations of the channel matrix are generated, similarly to the NN dataset, and the true MI in each case is calculated with a Monte Carlo simulation with $1,000$ realizations of the noise. Afterwards, the ergodic MI for each SNR point is calculated by averaging the instantaneous values of the MI. The true ergodic MI, shown with circles, is compared with that obtained by averaging the instantaneous MI calculated with each method, the two analytical approximations and the 10 neurons neural network. As it can be seen, the neural network matches perfectly the true ergodic MI, which is overestimated by the other methods for moderate values of the SNR.

The accuracy improvement of the MFNN as compared with the analytical approximations in Fig. 8 might seem small. Nevertheless, under block fading with an almost static channel matrix during the transmission of a frame, the instantaneous error for a single channel matrix is more relevant. In this latter case, the plots of Fig. 7 and the data of Table III demonstrate that the previous analytical approximations have a non-negligigle positive bias, whereas the MFNNs do

not, making our ML-based approach a better method for adaptation purposes.

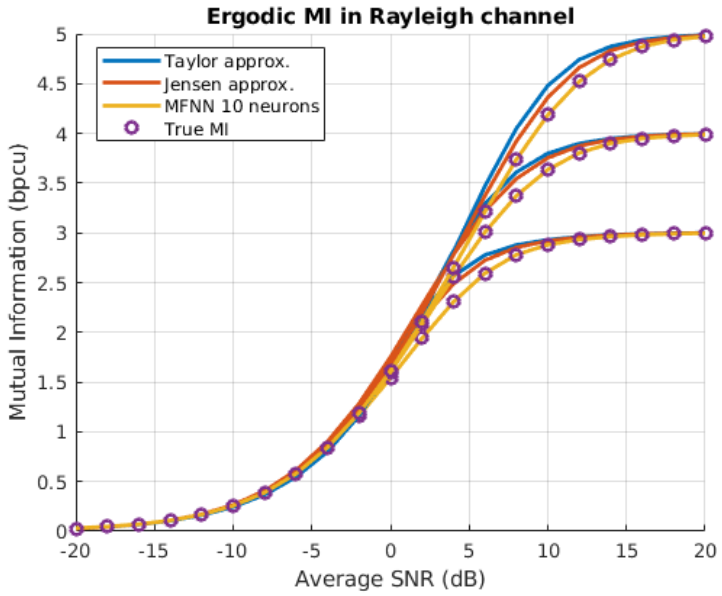


Fig. 8: Ergodic MI of a 2×2 SM system obtained after averaging the instantaneous MI calculated with each method for a Rayleigh channel (QPSK, 8PSK and 16QAM).

In addition to the estimation accuracy, the computational complexity is key for practical implementation: the MI computation must be done at the receive side, which has knowledge of the SNR γ and the channel matrix $\mathbf{H}$. This evaluation must be such that an on-line tracking of the channel state is made to report the selected physical layer configuration back to the receiver.

In this regard, Table IV shows the computational complexity of each method after counting the number of mathematical operations required to compute the MIs for the three constellations. In the case of the analytical approximations, the table shows a lower bound of the number of operations since it only counts the most demanding instructions, which are repeated $(2M)^2$ times. In the case of the NN, all the required operations are counted, including the preprocessing of γ and $\mathbf{H}$ to calculate the inputs of the NN.

The numbers in Table IV reveal that the MFNN is not only more accurate, but also less computationally demanding. The MFNN requires about 90 times fewer multiplications and non linear operations than the analytical approximations. This is in line with the required time to compute with Matlab® the three MIs for all the testing dataset, in a computer equipped with an i7-4510U 2 GHz processor. From another point of view, the laptop only allows to make an

	Taylor approximation	**Jensen based approximation**	**MFNN option (v) 20 neurons**
Real products	$7,168$	$32,800$	368
$\exp(\cdot)$	672	$1,344$	20
$\log_2(\cdot)$	112	3	-
Other non-linear operations	$1,344$	-	3
Calculation time for $7,500$ channel realizations	**41 s**	**76 s**	**0.80 s**
Calculation time for one channel realization	**5.5 ms**	**10.1 ms**	**0.1 ms**

TABLE IV: Comparison of complexity and computational time of the MFNN and the two methods of the literature. Total number of operations for computing the three MI values (QPSK, 8PSK and 16QAM) in a 2×2 SM system are given, as well as the computational time required for one and $7,500$ channel realizations with Matlab® running in a laptop.

estimation of the MIs every 5.5 ms with the Taylor approximations, which gets reduced to 0.1 ms with the NN. With respect to the off-line training duration, each training of the NN took typically less than 3 minutes.

VII. Extension to a higher number of antennas

This paper is focused mainly on a SM system with two dimensions in order to study thoroughly the impact of the different input features selection. However, this approach can be easily extended to compute the MI in scenarios with a higher number of antennas. This section aims to explain how to apply the same philosophy to obtain the MI of a 4×4 and 8×8 SM systems, providing also some clues to keep the complexity bounded for higher numbers of antennas.

Let us recall Table I, which portraits several selections of the NN input features for the case of 2 antennas. As can be seen, the performance improves from the top to the botton; in order to avoid too a large number of input features in systems with a higher number of antennas, we propose to apply option (ii) as a trade-off between performance and complexity. This option makes use of the channel column norms (scaled by the SNR) and the angles. Whilst the number of norms increases only linearly with the transmit antennas, the number of angles raises rapidly with N_t since there are $2\binom{N_t}{2}$ angles, a tuple (Θ_H, φ) for each possible combination of two transmit antennas. For example, in a system with 16 antennas there are 120 pairs of angles. However, we have found out that it is not necessary to give the values of all the angles explicitly to the

neural network. A few values characterizing the statistical distribution of the angles suffice for the NN to estimate the MI with an MSE similar to those values reported in Table II.

The MI evaluation in a SM system with 4×4 antennas can be easily done with an MFNN trained with the proper dataset, obtained now with 4×4 Rayleigh matrices, and using as input features the four values of $\gamma\|\mathbf{h}_l\|^2$ and the six pairs of angles (θ_H, φ). In the case of an 8×8 IM system we propose to reduce the $\binom{8}{2} = 28$ pairs of angles to just Q values per type of angle (Hermitian and Kasner). These Q values are the quantiles of the angles distribution for Q probabilities taken from 0 to 1 at equal steps. For example, for $Q = 5$ the distribution of the angles is characterized by the minimum, the 25th percentile, the 50th percentile (the median), the 75th percentile and the maximum. Therefore, the MFNN for obtaining the set of MI of an 8×8 IM system has as input features the 8 columns norms $\gamma\|\mathbf{h}_l\|^2$, and the Q quantiles of the Hermitian angle Θ_H and the Kasner's pseudoangle φ, respectively.

For testing purposes, we have generated two additional datasets, one with $50,000$ 4×4 Rayleigh matrices and another with $25,000$ 8×8 Rayleigh matrices. Again, each channel matrix has associated a random SNR value between -20 and 20 dB, and we have calculated the MI of each pair $(\gamma, \mathbf{H})$ for several constellations (QPSK, 8PSK and 16QAM) using Monte Carlo simulations. Following the same procedure of training and testing described in Section VI, we have obtained two trained MFNNs for calculating the three MI values of 4×4 and 8×8 SM systems, respectively.

Table V sums up the results obtained with these two neural networks. Note that for the 8×8 system we have used only $Q = 5$ quantiles. If we compare the results of 4 and 8 antennas provided in Table V with those obtained with the best network for the 2 antennas scenario from Table III, using 20 neurons and input features option (v), the NN performs a little worse in the setup with more antennas, although the errors are of the same magnitude. However, a fair comparison with the NN for 2×2 SM which uses the same type of input features, i.e., option (ii) from Table II, reveals that the MSE is slightly better when the number of dimensions grows.

Fig. 9 intends to show graphically the accuracy of the proposed MFNNs to calculate the MI of these 4×4 and 8×8 SM systems. Firstly, Fig. 9a contains a scatter-plot with the MI computed with the MFNN (Y axis), and the true MI (X axis) of a 8×8 SM system with the three constellations, QPSK, 8PSK and 16QAM, similarly to Fig. 7. It can bee seen how all the points

	NN input features	Global MSE	QPSK		8PSK		16QAM	
			3σ	Max. error	3σ	Max. error	3σ	Max. error
SM 2×2 (20 neurons), option (ii)	$2+2\times1=4$	$3.36\cdot10^{-4}$	0.078	0.436	0.041	0.342	0.034	0.322
SM 4×4 (20 neurons)	$4+2\times6=16$	$2.40\cdot10^{-4}$	0.047	0.169	0.050	0.200	0.041	0.137
SM 8×8 ($Q=5$, 20 neurons)	$8+2\times5=18$	$5.06\cdot10^{-5}$	0.022	0.050	0.023	0.061	0.018	0.046

TABLE V: Performance of the MFNNs for obtaining the MI for QPSK, 8PSK and 16QAM of a SM system with 2, 4 and 8 transmit and receive antennas, using as features the column norms and the angles.

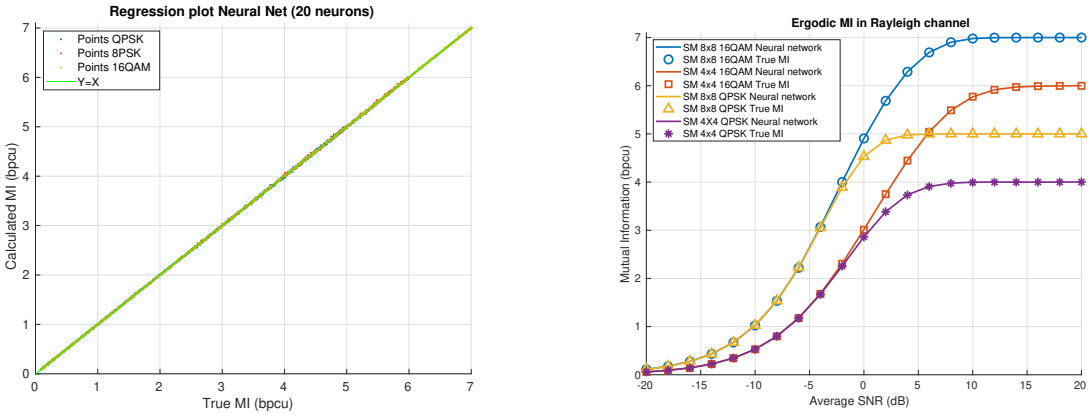
are very close to the green line, which represents the perfect match. This behaviour, similar in the 4×4 case, could be expected from the numerical results of Table V. Due to space restrictions, we did not include here the results of the analytical approximations for these systems with more than two antennas; however, they perform significantly worse than the neural network, as shown previously for the 2×2 system.

On the other hand, Fig. 9b shows the ergodic MI of SM systems with 4 and 8 transmit and receive antennas and two different constellations, QPSK and 16QAM, under a Rayleigh channel. For each point of SNR the ergodic MI is obtained by averaging the MI of 100 channel realizations, evaluated with the neural network (continuous line), and with the Monte Carlo method with $1,000$ realizations of the noise (markers). It can be seen how the ergodic MI obtained by means of the neural network matches perfectly the true ergodic MI, which is calculated by means of costly Monte Carlo simulations.

Lastly, the accuracy gain of the neural network as compared with the analytical approximations of the literature comes along with a complexity reduction also for SM systems with a higher number of antennas. In Table VI, we provide some numbers regarding the computational complexity of the proposed method to calculate the three MIs (QPSK, 8PSK and 16QAM) of an 8×8 SM system compared with the two analytical approximations. Similarly to the 2×2 case, the Taylor and Jensen approximations require here about 55 and 100 times more computational time for obtaining the MIs for a channel realization, respectively. Therefore, the neural network is again not only more accurate, but also more computationally efficient for the evaluation of the MI in SM systems with 4 and 8 transmit and receive antennas.

	Taylor approximation	Jensen based approximation	MFNN Q=5, 20 neurons
Real products	$372,736$	$6,195,200$	$1,695$
$\exp(\cdot)$	$10,752$	$21,504$	20
$\log_2(\cdot)$	448	3	-
Other non-linear operations	$21,504$	-	84
Calculation time for $7,500$ channel realizations	**637 s**	**$1,143$ s**	**11.2 s**
Calculation time for one channel realization	**85 ms**	**152 ms**	**1.5 ms**

TABLE VI: Comparison of complexity and computational time of the MFNN and the two methods of the literature. Total number of operations for computing the three MI values (QPSK, 8PSK and 16QAM) in a 8×8 SM system are provided, together with the computational time for one and $7,500$ channel realizations with Matlab® running in a laptop (the MFNN has 20 neurons, and uses as input features the column norms and the angles with $Q = 5$).



(a) Scatter plot of the MI of a SM 8x8 system calculated with the neural network which uses only the column norms and the angles (Q=5, 20 neurons).

(b) Ergodic MI of two 4×4 and 8×8 SM systems with QPSK and 16QAM constellations.

Fig. 9: Performance of the MFNN for computing the MI of SM systems with more than two antennas, showing the MI scatter plot of a SM 8×8 system in Fig. 9a, and the ergodic MI of SM 4×4 and SM 8×8 systems in Fig. 9b.

VIII. Conclusions

The implementation of next generation adaptive Spatial Modulation links requires practical mechanisms to estimate the mutual information for a given signalling strategy. An accurate and timely computation of this constrained capacity serves to adapt the constellation order, and apply

a fine tuning of the coding rate of the channel encoder, providing a better fit to the maximum achievable rate. The method proposed in this paper to calculate the constrained capacity is a very simple Multilayer Feedforward Neural Network, which can obtain the Mutual Information for different symbol constellations simultaneously. The neural network is more accurate and less computationally demanding than the analytical approximations existing in the literature, and provides a valuable reference achievable rate for different adaptive Spatial Modulation schemes.

IX. Acknowledgments

This work was funded by the Xunta de Galicia (Secretaria Xeral de Universidades) under a predoctoral scholarship (cofunded by the European Social Fund) and it was partially funded by the Agencia Estatal de Investigación (Spain) and the European Regional Development Fund (ERDF) under project MYRADA (TEC2016-75103-C2-2-R). It was also funded by the Xunta de Galicia and the ERDF (Agrupación Estratéxica Consolidada de Galicia accreditation 2016-2019). Furthermore, this work has received funding from the Spanish Agencia Estatal de Investigación under project TERESA, TEC2017-90093-C3-1-R (AEI/FEDER,UE); and from the Catalan Government (2017 SGR 891 and 2017 SGR 1479).

References

[1] LTE; evolved universal terrestrial radio access (E-UTRA); physical layer procedures. *ETSI TS 136 213 V14.2.0 (2017-04)*.

[2] IEEE Standard for Information technology– Telecommunications and information exchange between systems. Local and metropolitan area networks– Specific requirements–Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications–Amendment 4: Enhancements for Very High Throughput for Operation in Bands below 6 GHz. *IEEE Std 802.11ac-2013*, pages 1–425, Dec 2013.

[3] Satellite component of UMTS (S-UMTS); family SL satellite radio interface. *ETSI TS 102 744*, Oct. 2015.

[4] Krishna Sayana and Jeff Zhuang. Link performance abstraction based on mean mutual information per bit (MMIB) of the LLR channel, 2007.

[5] E. Basar. Index modulation techniques for 5G wireless networks. *IEEE Communications Magazine*, 54(7):168–175, July 2016.

[6] R. Mesleh, H. Haas, C. W. Ahn, and S. Yun. Spatial modulation - a new low complexity spectral efficiency enhancing technique. In *2006 First International Conference on Communications and Networking in China*, pages 1–5, Oct 2006.

[7] P. Yang, M. Di Renzo, Y. Xiao, S. Li, and L. Hanzo. Design Guidelines for Spatial Modulation. *IEEE Communications Surveys Tutorials*, 17(1):6–26, Firstquarter 2015.

[8] P. Henarejos and A. I. Perez-Neira. Dual Polarized Modulation and Reception for Next Generation Mobile Satellite Communications. *IEEE Transactions on Communications*, 63(10):3803–3812, Oct 2015.

[9] D.-T. Phan-Huy, Y. Kokar, K. Rachedi, P. Pajusco, A. Mokh, T. Magounaki, R. Masood, C. Buey, P. Ratajczak, N. Malhouroux-Gaffet, J. . Conrat, J. . Prvotet, A. Ourir, J. De Rosny, M. Crussire, M. Hlard, A. Gati, T. Sarrebourse, and M. Di Renzo. Single-Carrier Spatial Modulation for the Internet of Things: Design and Performance Evaluation by Using Real Compact and Reconfigurable Antennas. *IEEE Access*, 7:18978–18993, 2019.

[10] Viktor Nikolaidis, Nektarios Moraitis, and Athanasios G. Kanatas. Statistical characterization of an urban dual-polarized MIMO LMS channel. *International Journal of Satellite Communications and Networking*, 36(6):474–488, 2018.

[11] A. Tato, C. Mosquera, P. Henarejos, and A. Prez-Neira. Neural Network Aided Computation of Generalized Spatial Modulation Capacity. In *2019 27th European Signal Processing Conference (EUSIPCO)*, pages 1–5, Sep. 2019.

[12] T. Lakshmi Narasimhan and A. Chockalingam. On the capacity and performance of generalized spatial modulation. *IEEE Communications Letters*, 20(2):252–255, Feb 2016.

[13] P. Henarejos, A. Perez-Neira, A. Tato, and C. Mosquera. Channel Dependent Mutual Information in Index Modulations. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 3261–3265, April 2018.

[14] Nauman Ahad, Junaid Qadir, and Nasir Ahsan. Neural networks in wireless networks: Techniques, applications and guidelines. *Journal of Network and Computer Applications*, 68:1 – 27, 2016.

[15] O. Simeone. A very brief introduction to machine learning with applications to communication systems. *IEEE Transactions on Cognitive Communications and Networking*, 4(4):648–664, Dec 2018.

[16] M. M. A. Moustafa and S. H. A. El-Ramly. Channel estimation and equalization using backpropagation neural networks in OFDM systems. In *2009 IFIP International Conference on Wireless and Optical Communications Networks*, pages 1–4, April 2009.

[17] M. T. E. A. Elsoufi, X. Ying, W. Jun, and T. Bin. Fletcher-Reeves learning approach for high order MQAM signal modulation recognition. In *2016 7th International Conference on Information and Communication Systems (ICICS)*, pages 74–79, April 2016.

[18] M. Mirmohammadsadeghi, S. S. Hanna, and D. Cabric. Modulation classification using convolutional neural networks and spatial transformer networks. In *2017 51st Asilomar Conference on Signals, Systems, and Computers*, pages 936–939, Oct 2017.

[19] A. Marseet and F. Sahin. Application of complex-valued convolutional neural network for next generation wireless networks. In *2017 IEEE Western New York Image and Signal Processing Workshop (WNYISPW)*, pages 1–5, Nov 2017.

[20] P. V. R. Ferreira, R. Paffenroth, A. M. Wyglinski, T. M. Hackett, S. G. Biln, R. C. Reinhart, and D. J. Mortensen. Multiobjective Reinforcement Learning for Cognitive Satellite Communications Using Deep Neural Network Ensembles. *IEEE Journal on Selected Areas in Communications*, 36(5):1030–1041, May 2018.

[21] J. Kassab and S. Nagaraj. Adaptive modulation in an OFDM communications system with artificial neural networks. In *2009 International Joint Conference on Neural Networks*, pages 1547–1551, June 2009.

[22] Sandanalakshmi R and Kadhiravan D. Modified EESM Link Adaptation Method with Multiple Constellation for Future Wireless Networks. *IJRCCT*, 2(7), 2013.

[23] N. Baldo and M. Zorzi. Learning and Adaptation in Cognitive Radios Using Neural Networks. In *2008 5th IEEE Consumer Communications and Networking Conference*, pages 998–1003, Jan 2008.

[24] H. Sun, X. Chen, Q. Shi, M. Hong, X. Fu, and N. D. Sidiropoulos. Learning to optimize: Training deep neural networks for wireless resource management. In *2017 IEEE 18th International Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*, pages 1–6, July 2017.

[25] A. Rico-Alvarino and R. W. Heath. Learning-Based Adaptive Transmission for Limited Feedback Multiuser MIMO-OFDM. *IEEE Transactions on Wireless Communications*, 13(7):3806–3820, July 2014.

[26] P. Yang, Y. Xiao, M. Xiao, Y. L. Guan, S. Li, and W. Xiang. Adaptive spatial modulation MIMO based on machine learning. *IEEE Journal on Selected Areas in Communications*, 37(9):2117–2131, Sep. 2019.

[27] V. Saxena, B. Cavarec, J. Jaldn, M. Bengtsson, and H. Tullberg. A learning approach for optimal codebook selection in spatial modulation systems. In *2018 52nd Asilomar Conference on Signals, Systems, and Computers*, pages 1800–1804, Oct 2018.

[28] P. Yang, Y. Xiao, Y. Yu, and S. Li. Adaptive spatial modulation for wireless MIMO transmission systems. *IEEE Communications Letters*, 15(6):602–604, June 2011.

[29] P. Yang, Y. Xiao, Y. Yu, L. Li, Q. Tang, and S. Li. Simplified adaptive spatial modulation for limited-feedback MIMO systems. *IEEE Transactions on Vehicular Technology*, 62(6):2656–2666, July 2013.

[30] S. Guo, H. Zhang, J. Zhang, and D. Yuan. On the mutual information and constellation design criterion of spatial modulation MIMO systems. In *2014 IEEE International Conference on Communication Systems*, pages 487–491, Nov 2014.

[31] E. Basar, M. Wen, R. Mesleh, M. Di Renzo, Y. Xiao, and H. Haas. Index Modulation Techniques for Next-Generation Wireless Networks. *IEEE Access*, 5:16693–16746, 2017.

[32] Thomas M. Cover and Joy A. Thomas. *Elements of Information Theory*. Wiley-Interscience, 1991.

[33] Norbert Schorghofer. *Lessons in Scientific Computing: Numerical Mathematics, Computer Technology, and Scientific Discovery*. Routledge, 2018.

[34] Ken-Ichi Funahashi. On the approximate realization of continuous mappings by neural networks. *Neural networks*, 2(3):183–192, 1989.

[35] Kurt Hornik. Approximation capabilities of multilayer feedforward networks. *Neural Networks*, 4(2):251 – 257, 1991.

[36] D. Nguyen and B. Widrow. Improving the learning speed of 2-layer neural networks by choosing initial values of the adaptive weights. In *1990 IJCNN International Joint Conference on Neural Networks*, pages 21–26 vol.3, June 1990.

[37] M. T. Hagan and M. B. Menhaj. Training feedforward networks with the Marquardt algorithm. *IEEE Transactions on Neural Networks*, 5(6):989–993, Nov 1994.

[38] P. Yang, Y. Xiao, L. Li, Q. Tang, Y. Yu, and S. Li. Link Adaptation for Spatial Modulation With Limited Feedback. *IEEE Transactions on Vehicular Technology*, 61(8):3808–3813, Oct 2012.

[39] G. Zafari, M. Koca, and H. Sari. Dual-Polarized Spatial Modulation Over Correlated Fading Channels. *IEEE Transactions on Communications*, 65(3):1336–1352, March 2017.

[40] K. Scharnhorst. Angles in complex vector spaces. *Acta Applicandae Mathematica*, 69(1):95–103, Oct 2001.