Detection and Channel Equalization with Deep Learning for Low Resolution MIMO Systems

Aldebaro Klautau
Dep. of Comp. and Telecom. Eng.
Federal University of Pará
Belem, PA, 66075-110, Brazil
aldebaro@ufpa.br

Nuria González-Prelcic, Amine Mezghani and Robert W. Heath Jr.
Dep. of Elec. and Computer Eng.
The University of Texas, Austin
Austin, TX, 78712-1084, USA
{ngprelcic,amine.mezghani,rheath}@utexas.edu

***Abstract*—Deep learning (DL) provides a framework for designing new communication systems that embrace practical impairments. In this paper, we present an exploration of DL as applied to design the physical layer for MIMO systems with low resolution analog-to-digital converters. The application of DL is nontrivial thanks to the severe nonlinear distortion caused by quantization and the large dimensional MIMO channel. We investigate network architectures for channel estimation and detection. The channel estimation results indicate that the adopted DL architectures lead to good results in the large signal-to-noise ratio (SNR) regime, but are outperformed by state-of-the-art iterative message passing algorithms. For decoding, we adopted a multilabel classification architecture with implicit equalization and output size scaling linearly with the number of data symbols to be estimated. While feasible for high MIMO dimensions, the adopted DL architecture for decoding converged only for relatively small MIMO dimensions. A main conclusion of our paper is that DL still has potential but more efficient architectures are required, given the convergence problems associated with time-varying channels and 1-bit quantization.**

I. Introduction

Deep learning (DL) provides means to design communication systems accounting for practical impairments which may be difficult using standard digital communication theory [1]–[3]. In this paper, we study the application of DL techniques to MIMO systems with 1-bit analog to digital converters (ADCs). Such MIMO systems offer a low power solution thanks to power reductions that come from reducing ADCs resolution. As a result, they find application to high bandwidth millimeter wave communication and also massive MIMO systems, both places where ADCs become a dominate source of power consumption. Designing receiver algorithms including channel estimation and detection is challenging due to the nonlinearity imposed by quantization and the large dimensions involved.

State-of-the-art receiver algorithms for MIMO with low resolution ADCs are based on *message passing* (MP) [4], especially the *generalized approximate* MP (GAMP) algorithm [5] and its variants. The bilinear (BiGAMP) [6] and the *parametric* BiGAMP (PBiGAMP) [7] are particularly suitable for joint channel and detection estimation (JCD). The GAMP-based algorithms presented in [8], [9] provide a good indication of the performance that can be obtained with iterative MP for JCD in coarsely quantized massive MIMO. In [10], GAMP variants are customized to OFDM systems, tackling channel estimation of millimeter wideband channels. The results (e. g. in [8]–[10]) indicate that MP can reach close to optimal performance if the associated assumptions hold. A recent trend in DL applied to the physical layer (PHY) is to *unfold* iterative algorithms such as GAMP [11]–[15]. That work is different from our proposed work in that we use DL to do channel estimation and detection for MIMO systems with 1-bit ADCs.

DL has been used to solve other physical layer communication problems for both SISO and MIMO systems. In SISO systems, DL was applied to: equalization and detection with fixed channels in [16], detection in [17] and JCD for OFDM in [18]. In MIMO systems, DL was used for: detection with autoencoders using *one-hot* encoding [19] in full resolution 2×2 MIMO systems [20], channel estimation [15] and detection over time-varying channels [13].

In this paper, we present a preliminary study of DL architectures for channel estimation and detection in MIMO systems with low resolution receivers, with focus on identifying issues related to their scalability. This is a main issue in prior work like [20], which does not scale well with, for example, the number of antennas. While the unfolded algorithms in [11]–[14] have the potential to scale, their drawback is to rely on the assumptions of the underlying iterative algorithm. We choose to not consider unfolding, aiming at scenarios impacted by, e. g., nonlinearity at the transmitter and correlated noise, for which assumptions required by variants of GAMP do not hold. But we do use variants of GAMP for channel estimation as a baseline [10], [21].

We make two main contributions in this paper. First, we report the results of our investigation of convolutional and other network architectures for channel estimation and (separately) multilabel classification for detection. Second, we discuss practical issues associated to DL applied to these problems and the difficulties that make, for instance, detection with a large number of antennas infeasible using one-hot encoding DL architectures. We believe our paper is an important step in the direction of further applications of DL to MIMO communication systems.

II. Communication System Model

We model a coarsely *quantized* MIMO system as in Fig. 1 and assume 1-bit ADCs. The number of antennas at the receiver and transmitter are N_r and N_t, respectively. Because this paper is an initial study, we assume frequency-flat block fading, with the channel remaining constant over a block composed by T consecutive symbol intervals. We also assume perfect synchronization such that the narrowband baseband received signal $\mathbf{Y} \in \mathbb{C}^{N_\mathrm{r}\times T}$ over the block interval can be written as

$$\mathbf{Y} = \mathbf{H}\mathbf{X} + \mathbf{W}, \tag{1}$$

where $\mathbf{H} \in \mathbb{C}^{N_\mathrm{r}\times N_\mathrm{t}}$ contains the channel coefficients, $\mathbf{X} \in \mathbb{C}^{N_\mathrm{t}\times T}$ has the transmit symbols normalized to have unity average total power per transmission, and $\mathbf{W} \in \mathbb{C}^{N_\mathrm{r}\times T}$ corresponds to additive white Gaussian noise (AWGN) with zero mean and variance σ^2 per complex-valued element. For simplicity, the pilot and data symbols are from a QPSK constellation. Investigation on how DL could eventually help the design of good pilot constellations is left for future work.The receiver implements 1-bit quantization after downconversion, with quantization applied independently to real and imaginary components. The resulting quantized signal is then

$$\tilde{\mathbf{Y}} = \mathrm{Q}(\mathbf{Y}). \tag{2}$$

The quantization operation in Eq. (2) is the main reason for the signal processing challenges associated with low resolution MIMO communication.

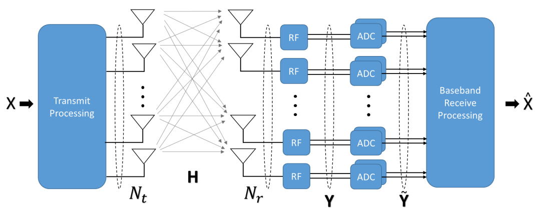

Fig. 1. Adopted quantized MIMO system model.

To facilitate channel estimation, pilot sequences are transmitted in the beginning of each block (*training phase*). The first T_t symbols are known to the receiver and organized in a matrix $\mathbf{X}_t \in \mathbb{C}^{N_\mathrm{t}\times T_\mathrm{t}}$, while the data symbols are represented by $\mathbf{X}_d \in \mathbb{C}^{N_\mathrm{t}\times T_\mathrm{d}}$, $T_\mathrm{d} = T - T_\mathrm{t}$. A similar partitioning is adopted for $\tilde{\mathbf{Y}}$. Hence, the information corresponding to the *training* and *data phases* is represented by $\mathbf{X} = [\mathbf{X}_t, \mathbf{X}_d]$ and $\tilde{\mathbf{Y}} = [\tilde{\mathbf{Y}}_t, \tilde{\mathbf{Y}}_d]$. We consider two problems. In *pilot-only channel estimation* the receiver uses $\tilde{\mathbf{Y}}_t$ and $\mathbf{X}_t$ to generate the estimate $\hat{\mathbf{H}}$. In *detection* with implicit equalization, the receiver estimates $\mathbf{X}_d$ from $\tilde{\mathbf{Y}}$.

We used the stochastic channel model adopted in [21], which allows controlling the sparsity level and is appropriate for applications to massive MIMO or millimeter wave MIMO. The L multipath components (MPCs) in the virtual (or angle) domain $\mathbf{H}_v$ are assumed to coincide with DFT bins (no leakage). Then an inverse 2D DFT generates the corresponding $\mathbf{H}$ [22].

III. Channel Estimation and Detection with DL

We first discuss channel estimation and then the detection problem. For simulations, an important detail in any DL problem, our software setup is centered on customized Python code using *Keras* with *Tensorflow* as backend. Keras and Tensorflow are among the most widely DL tools and are detailed, e. g, in [19].

A. Channel Estimation with DL

We pose the pilot-only channel estimation as a *multivariate regression* problem. The input is the binary array $\tilde{\mathbf{Y}}_t$ while the output is the complex-valued $\hat{\mathbf{H}}$. Multivariate regression is a hard problem and many times it is tackled by grouping N *multivariable* regressors, which are trained independently for simplicity. In this work we adopt a single neural network (NN) to perform multivariate regression [19]. It should be noted that the literature on NNs is very rich and precedes DL. For example, NNs for channel equalization were used in [23] and in many other communication problems. Besides, for non-perceptual data or when data is scarce, there are algorithms such as *gradient boosting* that are highly competitive with DL [19].

The NN is trained with the *mean-squared error* (MSE) loss and aims at providing the *minimum* MSE (MMSE) estimation

$$\mathbf{H}^\dagger = \min_{\hat{\mathbf{H}}} \mathbb{E}[||\mathbf{H} - \hat{\mathbf{H}}||^2]. \tag{3}$$

The 1-bit quantization means that it is not possible to estimate the norm of the channel with zero-threshold quantizers. As in [10], we assume this information can be recovered from the automatic gain control in the analog circuitry. Therefore, we suppose that both training and test data have $||\mathbf{H}||^2 = N_\mathrm{t}N_\mathrm{r}$ and normalize the channel after the MMSE estimate.

The training does not require knowledge of the distribution $p(\mathbf{H})$ over channels, but access to a reasonable number of realizations (to compose a rich training set from e. g. measurement data) or a software routine to draw samples from this distribution on-the-fly. In contrast, state-of-art GAMP-based algorithms consider the receiver knows the distribution $p(\mathbf{H})$ of channels but not its realizations. Knowing distributions for AMP (even if not their parameters) and having large datasets for DL, are similar in the sense that both are manifestations of access to a potentially infinite amount of data. One distinction is that GAMP variants leverage the analytical expression of $p(\mathbf{H})$ as a highly compact representation of knowledge about the channels. When trained using Eq. (3), the NN is expected to find its own way of representing all relevant information contained in $p(\mathbf{H})$. Similarly, we train a NN under different noise conditions (*multi-condition* training) and expect it to learn and generalize on the conditions of interest.

Both noise multi-condition training and the time-variant channel are challenging for the stochastic gradient descent (SGD) used in DL. The network training with SGD may not converge even with advanced Keras' optimizers such as *Adam*. Most SGD routines obtain the gradient estimate by averaging the individual gradients of a set of B examples called *mini* *batch*. Having $B > 1$ often helps convergence by averaging the noise out and may be essential when the SNR imposed during training is low. But in the case of time-varying channels (examples corresponding to eventually distinct channels $\mathbf{H}$), SGD may not find a reasonable average direction even if σ^2 is small. For improved performance we do not change the channel within a mini batch. The procedure can be interpreted as keeping the channel constant not only over a block of T consecutive symbol intervals, but $B \times T$. Using this method, we evaluated different NNs for DL-based channel estimation, which are detailed in Section IV.

It is instructive to observe that for 1-bit MIMO there are $M = 2^{2N_\text{r}T_\text{t}}$ distinct received pilots $\tilde{\mathbf{Y}}_t$ and channel estimation could be implemented as a look-up table. The input would be the index corresponding to $\tilde{\mathbf{Y}}_t$, and $\hat{\mathbf{H}}$ the complex-valued output with dimension $N_\text{t}N_\text{r}$. Such look-up tables are not feasible when large dimensions are involved but inspire machine learning algorithms (see, e. g., [24]). For instance, $N_\text{r} = 200$ and $T_\text{t} = 50$ were adopted in [8], which leads to more than 10^{6000} possible binary arrays. We want to train NNs that implement this mapping with reasonable complexity.

B. Detection with DL

In a machine learning framework, it is natural to pose detection as a *multiclass* or *multilabel classification* problem. On similar conditions, it is often the case that the level of difficulty increases from multiclass classification, multilabel classification, up to multivariate regression. One of the most popular DL architectures for detection is the *autoencoder*, which in previous work is applied as a multiclass problem (see, e. g., [20]). This is often accomplished by encoding inputs and outputs with integers $1, \ldots, Q$ using one-hot encoding or embeddings, and the *categorical cross-entropy* as the loss function [19]. But this is not feasible when the involved dimensions are large. For example, in 1-bit MIMO systems the number of distinct data blocks is $Q = 2^{2N_\text{r}T_\text{d}}$.

An alternative to alleviate the dimensionality scaling is to pose the problem as multilabel classification, adopting other encoding schemes and, e. g, a *binary cross-entropy* as the loss function. In this case the network output activation function can be the *sigmoid*, instead of the *softmax* activation used for multiclass classification. However, as mentioned, training a multilabel classifier is often harder than a multiclass.

Besides the scaling problem, training a 1-bit MIMO system as an autoencoder requires backpropagating gradients through the quantization layer and channel. Examples of proposed solutions in recent literature to the issue of gradient backpropagation through the channel in *end-to-end* learning are: not optimizing the transmitter [25], iterating between supervised learning of the receiver and reinforcement learning of transmitter [26], calculating approximate gradients [27] and using surrogate models such as generative adversarial networks (GAN) [28], [29]. We wanted to focus on the quantization layer and, in the end-to-end simulations, used a channel layer that assumed knowledge of $\mathbf{H}$ and that could backpropagate the gradient [20]. It remained to deal with the derivative of a 1-bit quantizer (*sign*) function being undefined at the origin and zero elsewhere. We implemented customized quantization layers on Keras, using sigmoid-like functions with very steep transitions and also passing the gradient unchanged. When assuming practical MIMO dimensions, however, the autoencoder training did not converge. Even for small dimensions (e. g., 2×2 MIMO), obtaining convergence required considerable parameter tuning. As a result, we obtained our numerical results using another architecture.

The adopted architecture (called *multilabel*) is based on posing decoding as a multilabel classification problem. It is not end-to-end and is trained to perform detection using implicit channel equalization. The target output $\mathbf{X}_d$ is represented as an array of bits with dimension $2N_\text{t} \times T_\text{d}$. The input is composed by the received symbols $\tilde{\mathbf{Y}}$ organized as a $N_\text{r} \times T$ binary array. The loss is the binary cross-entropy and the output activation is the sigmoid.

A special case of the *multilabel* architecture corresponds to restricting the network to process a single data vector together with all pilots. The input and output dimensions are $N_\text{r} \times (T_\text{t} + 1)$ and $2N_\text{t} \times 1$, respectively. This does not change the transmit block structure, which can have $T_\text{d} > 1$. But for the DL processing, a sliding window over $\tilde{\mathbf{Y}}_d$ feeds the NN with a single data vector of dimension N_r. This is motivated by the fact that the channel is memoryless and the strategy allows to decrease the computational cost. This special case is denoted as *multilabel_Td1* given that it is equivalent to having $T_\text{d} = 1$.

IV. Simulation Results

The detection performance is assessed by the bit error rate (BER) between $\mathbf{X}_d$ and $\hat{\mathbf{X}}_d$, using Monte Carlo simulations for various SNRs. The SNR is defined as $1/\sigma^2$. Channel estimation is assessed by the *normalized mean-squared error* $\text{NMSE} = \mathbb{E}\left[\|\mathbf{H} - \hat{\mathbf{H}}\|_2^2 / \|\mathbf{H}\|_2^2\right]$ between the channel $\mathbf{H}$ and its estimation $\hat{\mathbf{H}}$. We used a software routine to draw samples from $p(\mathbf{H})$ on-the-fly. More specifically, we implemented channel *generators* as instances of Keras' Sequence class for the *sparse* channel model with the number of MPCs uniformly distributed from 1 to $N_\text{r}N_\text{t}$. The NNs were trained with the Adam optimizer.

A. Pilot-only MIMO channel estimation

We used the MSE as the loss function and a customized Keras layer that normalizes the channel estimator output to have a norm $\sqrt{N_\text{t}N_\text{r}}$. We tested with a set of 200 channels, disjoint from training data. We used $N_\text{r} = 64$, $N_\text{t} = 8$ and trained the networks with different noise conditions. More specifically, for each training instance a SNR value was randomly draw from a uniform distribution.

Dense, convolutional and residual networks were evaluated [19] and the best results were obtained with no more than five layers. The results did not improve with *batch normalization* nor *dropout*. For convolutional layers, the kernel dimensions were dependent on the dimensions of the input vectors and we did not use *max pooling*. We also tested conditional GANs. For GANs, the information from random noise latent variables and received pilots $\tilde{\mathbf{Y}}_t$ were inputs to *embedding* layers, which had their outputs concatenated. We also tested with only $\tilde{\mathbf{Y}}_t$ as the generator's input, but the GANs did not converge and their results are not reported here.

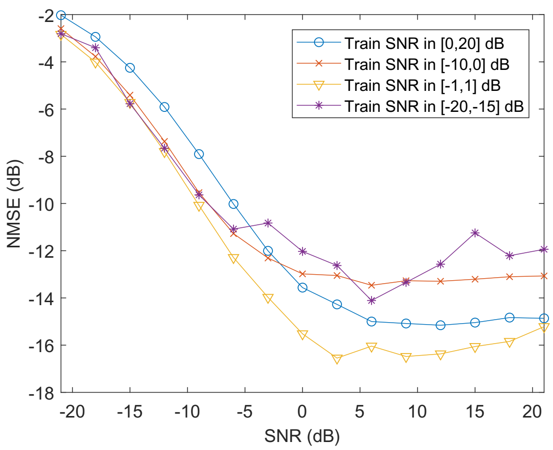

Fig. 2. Impact of the adopted SNR range during multi-condition training on channel estimation for 8×64 quantized MIMO.

Fig. 2 illustrates an aspect of multi-condition training using four residual networks trained with $T_{\mathrm{t}} = 256$, $B = 5$ and distinct supports for the SNR distribution: $[0, 20]$, $[-10, 0]$, $[-1, 1]$ and $[-20, -15]$ dB. In the test stage the SNR is fixed, and varied from -21 to 21 dB. The results in Fig. 2 indicate that in this scenario, training with SNRs in the range $[-1, 1]$ dB leads to the best results, outperforming even the training with $[0, 20]$ dB in the high SNR regime. All these networks had approximately 10^6 parameters, consisting of five layers with a skip connection from the second to the fourth layer. The training SNR range of $[-1, 1]$ dB was adopted in all other simulations in this work.

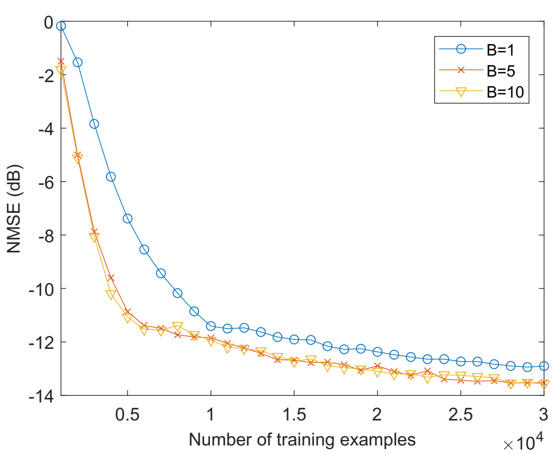

Fig. 3. Impact of different mini batch sizes B on convergence of channel estimation for 8×64 quantized MIMO using test SNR within $[-1, 1]$ dB.

Fig. 3 depicts the impact of mini batch size B on convergence. The networks used the same architecture as those of Fig. 2 and the test SNR used the same range as the training SNR: $[-1, 1]$ dB. Each training procedure used 3×10^4 channel realizations. In this case $B = 1$ converged to a NMSE 0.7 dB higher than the obtained with $B = 5$ or $B = 10$. In other simulations we considered B an hyperparameter and tried to find a reasonable value for each scenario.

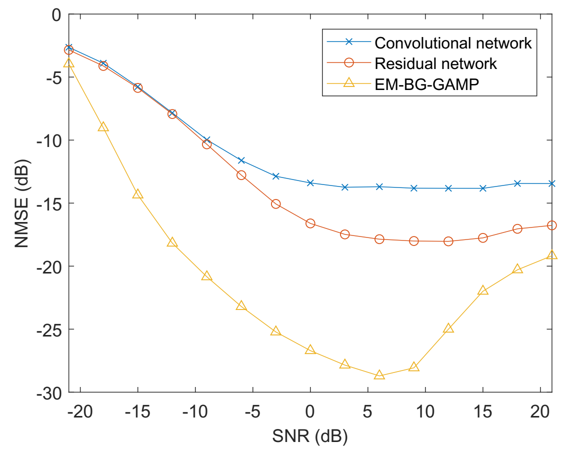

Fig. 4. Comparison of two DL architectures and EM-BG-GAMP for 4×64 quantized MIMO channel estimation.

Fig. 4 compares the results of convolutional and residual architectures with the EM-BG-GAMP, which uses the expectation-maximization (EM) algorithm with a Bernoulli-Gaussian prior [21]. In this case, $N_{\mathrm{r}} = 64$, $N_{\mathrm{t}} = 4$, $T_{\mathrm{t}} = 256$. The NNs were trained with $B = 3$ and had approximately 10^6 parameters. The adopted simulation setup matches the GAMP assumptions and EM-BG-GAMP performs extremely well, as expected from [21].

B. Low-dimension MIMO detection

Training the *multilabel* architecture has proved challenging. The network is required to perform implicit equalization for time-varying channels and learn under various SNR levels using binary inputs. We were not able to obtain convergence for larger systems and present results for 2×2 MIMO.

Fig. 5 shows detection results for the *multilabel* architecture and its special case *multilabel_Td1*. Both used $N_{\mathrm{r}} = N_{\mathrm{t}} = 2$, $T_{\mathrm{t}} = 3$ pilot vectors and $T_{\mathrm{d}} = 2$ data vectors. Recall that *multilabel_Td1* is equivalent to processing with $T_{\mathrm{d}} = 1$. In this case, the *multilabel* networks observed, besides the pilots, $T_{\mathrm{d}} = 2$ data vectors and had to predict both. The *multilabel_Td1* networks had to predict a single data vector. The networks used dense layers and $B = 32$. We varied the number of neurons, leading to the approximate total number P of parameters in the networks was 10^4, 2.5×10^5 or 10^6. The results in Fig. 5 indicate that *multilabel_Td1* can achieve with 250 thousand parameters a performance better than the one obtained with *multilabel* using one million parameters. It can also be noted that, while the data vectors are beneficial in JCD using GAMP variants [8], in this scenario the *multilabel* networks were not capable of benefiting from the extra information to internally produce a better channel equalization and detection. It remains

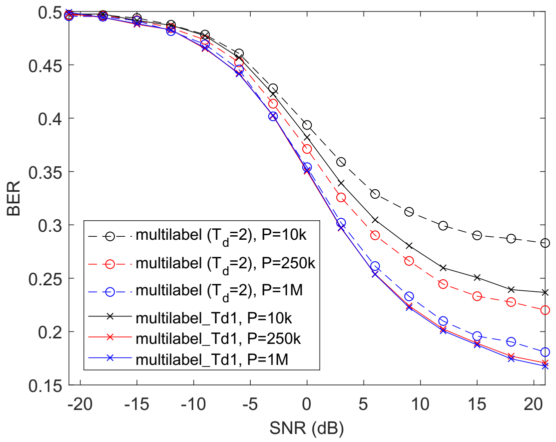

Fig. 5. BER for 1-bit DL-based low-dimension MIMO detection, where P is the total number of model parameters. The usual logarithmic scale for BER was not used given the small dynamic range.

as future work to develop alternative architectures and training methods, which would enable to check whether this behavior is observed in systems with larger dimensions.

V. Conclusions

In this paper, we explored some of the challenges of applying DL techniques to 1-bit MIMO systems. We focused on the issue of designing architectures that can cope with the relatively large dimensions. For channel estimation, SGD is able to converge even with varying-time channels if the mini batch size is properly tuned. DL-based detection of 1-bit MIMO with large numbers of receivers, without unfolding, requires further research. End-to-end learning has issues with backpropagating through the channel and quantization layer, and the proposed multilabel architecture presented convergence problems. The convergence can be improved with DL-based detection using subnetworks specialized on channel equalization and SNR estimation, as will be presented in an upcoming paper.

Acknowledgment

This material is based upon work supported in part by the National Science Foundation under Grant No. ECCS-1711702 and Grant No. CNS-1731658, as well as gifts from Nokia Bell Labs, Toyota ITC and Huawei. The work of A. Klautau was supported in part by CNPq, Brazil (201493/2017-9/PDE).

References

[1] Y. Yang, Y. Li, K. Li, S. Zhao, R. Chen, J. Wang, and S. Ci, "DECCO: Deep-learning Enabled Coverage and Capacity Optimization for Massive MIMO Systems," *IEEE Access*, pp. 1–1, 2018.

[2] T. Wang, C. K. Wen, H. Wang, F. Gao, T. Jiang, and S. Jin, "Deep learning for wireless physical layer: Opportunities and challenges," *China Communications*, vol. 14, no. 11, pp. 92–111, Nov. 2017.

[3] K. Kim, J. Lee, and J. Choi, "Deep Learning Based Pilot Allocation Scheme (DL-PAS) for 5G Massive MIMO System," *IEEE Communications Letters*, vol. 22, no. 4, pp. 828–831, Apr. 2018.

[4] F. Krzakala et al, *Statistical physics, optimization, inference, and message-passing algorithms*. Oxford University Press, 2016.

[5] S. Rangan, "Generalized approximate message passing for estimation with random linear mixing," in *2011 IEEE International Symposium on Information Theory Proceedings*, Jul. 2011, pp. 2168–2172.

[6] J. T. Parker, P. Schniter, and V. Cevher, "Bilinear Generalized Approximate Message Passing;, Part I: Derivation," *IEEE Transactions on Signal Processing*, vol. 62, no. 22, pp. 5839–5853, Nov. 2014.

[7] J. T. Parker and P. Schniter, "Parametric Bilinear Generalized Approximate Message Passing," *IEEE Journal of Selected Topics in Signal Processing*, vol. 10, no. 4, pp. 795–808, Jun. 2016.

[8] C. K. Wen, C. J. Wang, S. Jin, K. K. Wong, and P. Ting, "Bayes-Optimal Joint Channel-and-Data Estimation for Massive MIMO With Low-Precision ADCs," *IEEE Transactions on Signal Processing*, vol. 64, no. 10, pp. 2541–2556, May 2016.

[9] P. Sun, Z. Wang, R. W. Heath, and P. Schniter, "Joint channel-estimation/decoding with frequency-selective channels and few-bit ADCs," in *Proc. 51st Asilomar Conference on Signals, Systems, and Computers*, Oct. 2017, pp. 1824–1828.

[10] J. Mo, P. Schniter, and R. W. Heath, "Channel Estimation in Broadband Millimeter Wave MIMO Systems With Few-Bit ADCs," *IEEE Transactions on Signal Processing*, vol. 66, no. 5, pp. 1141–1154, Mar. 2018.

[11] N. Samuel, T. Diskin, and A. Wiesel, "Deep MIMO detection," in *Proc. IEEE 18th International Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*, Jul. 2017, pp. 1–5.

[12] X. Tan, W. Xu, Y. Be'ery, Z. Zhang, X. You, and C. Zhang, "Improving Massive MIMO Belief Propagation Detector with Deep Neural Network," *arXiv:1804.01002 [cs, eess, math, stat]*, Apr. 2018.

[13] H. He, C.-K. Wen, S. Jin, and G. Y. Li, "A Model-Driven Deep Learning Network for MIMO Detection," *arXiv:1809.09336 [cs, math]*, Sep. 2018.

[14] M. Borgerding, P. Schniter, and S. Rangan, "AMP-Inspired Deep Networks for Sparse Linear Inverse Problems," *IEEE Transactions on Signal Processing*, vol. 65, no. 16, pp. 4293–4308, Aug. 2017.

[15] H. He, C.-K. Wen, S. Jin, and G. Y. Li, "Deep Learning-based Channel Estimation for Beamspace mmWave Massive MIMO Systems," *arXiv:1802.01290 [cs, math]*, Feb. 2018.

[16] W. Xu, Z. Zhong, Y. Be'ery, X. You, and C. Zhang, "Joint Neural Network Equalizer and Decoder," *arXiv:1807.02040 [cs, eess, math]*, Jul. 2018.

[17] A. Felix, S. Cammerer, S. Dörner, J. Hoydis, and S. ten Brink, "OFDM-Autoencoder for End-to-End Learning of Communications Systems," *arXiv:1803.05815 [cs, eess, math]*, Mar. 2018.

[18] H. Ye, G. Y. Li, and B. Juang, "Power of Deep Learning for Channel Estimation and Signal Detection in OFDM Systems," *IEEE Wireless Communications Letters*, vol. 7, no. 1, pp. 114–117, Feb. 2018.

[19] F. Chollet, *Deep Learning with Python*. Manning Publications, 2017.

[20] T. J. O'Shea, T. Erpek, and T. C. Clancy, "Physical layer deep learning of encodings for the MIMO fading channel," in *Proc. 55th Annual Allerton Conference on Communication, Control, and Computing*, Oct. 2017, pp. 76–80.

[21] J. Mo, P. Schniter, N. G. Prelcic, and R. W. Heath, "Channel estimation in millimeter wave MIMO systems with one-bit quantization," in *Proc. 48th Asilomar Conference on Signals, Systems and Computers*, Nov. 2014, pp. 957–961.

[22] A. M. Sayeed, "Deconstructing multiantenna fading channels," *IEEE Transactions on Signal Processing*, vol. 50, no. 10, pp. 2563–2579, 2002.

[23] S. Chen, G. J. Gibson, C. F. N. Cowan, and P. M. Grant, "Adaptive equalization of finite non-linear channels using multilayer perceptrons," *Signal Processing*, vol. 20, no. 2, pp. 107–119, Jun. 1990.

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

[25] S. Dörner, S. Cammerer, J. Hoydis, and S. t Brink, "Deep Learning Based Communication Over the Air," *IEEE Journal of Selected Topics in Signal Processing*, vol. 12, no. 1, pp. 132–143, Feb. 2018.

[26] F. A. Aoudia and J. Hoydis, "End-to-End Learning of Communications Systems Without a Channel Model," *arXiv:1804.02276 [cs, math, stat]*, Apr. 2018.

[27] V. Raj and S. Kalyani, "Backpropagating Through the Air: Deep Learning at Physical Layer Without Channel Models," *IEEE Communications Letters*, pp. 1–1, 2018.

[28] T. J. O'Shea, T. Roy, N. West, and B. C. Hilburn, "Physical Layer Communications System Design Over-the-Air Using Adversarial Networks," *arXiv:1803.03145 [cs, eess]*, Mar. 2018.

[29] H. Ye, G. Y. Li, B.-H. F. Juang, and K. Sivanesan, "Channel Agnostic End-to-End Learning based Communication Systems with Conditional GAN," *arXiv:1807.00447 [cs, math]*, Jul. 2018.