A New Low-Resolution Min-Sum Decoder Based on Dynamic Clipping for LDPC Codes

Huanyu He
Department of Electronic Engineering
Shanghai Jiao Tong University
Shanghai, China
hehuanyu_ee@sjtu.edu.cn

Lei Chu
Department of Electrical Engineering
Shanghai Jiao Tong University
Shanghai, China
leochu@sjtu.edu.cn

Robert Caiming Qiu
Department of Electrical Engineering
Shanghai Jiao Tong University
Shanghai, China
rcqiu@sjtu.edu.cn

***Abstract*—Compared with the sum-product algorithm (SPA) with high decoding complexity for low-density parity-check (LDPC) codes, its approximated version, the min-sum algorithm (MSA), reduces the computational complexity at the cost of slight performance degradation. In order to compensate the oversized check-node messages in the MSA, the effect of clipping on variable-node messages is investigated under two-bit resolution. Our results show that the performance of clipped MSA degrades at the high bit error rate (BER) and increases at the low BER as clipping threshold enlarges. Based on these results, we propose a modified quantized MS decoding algorithm where the adaptive clipping threshold is applied on the variable-node messages according to the number of unsatisfied checks per-iteration. The adaptive rule is designed for easy hardware implementations. Numerical results show that the proposed algorithm has considerable improvement in performance compared with the MSA under two-bit precision. And it can achieve the performance near the SPA for some short-length LDPC codes.**

***Index Terms*—Low-density parity-check codes, min-sum algorithm, clipping, quantization, error floor.**

I. Introduction

As the new 5G wireless communications systems develop, more severe challenges will be encountered such as lower latency, higher energy efficiency and larger data rates [1]–[3]. The low-density parity-check (LDPC) codes can meet the requirements in modern communication systems for codification with high reliability and spectral efficiency [1].

The LDPC code which is a linear error correcting code was firstly introduced by Gallager [4] in the 1960's and rediscovered 30 years later by MacKay [5]. The sum-product algorithm (SPA) is one of the message-passing (MP) decoding algorithms under which the performance of LDPC codes can get close to the Shannon-limit over the additive white Gaussian noise (AWGN) channel [6]. However, the SPA is not suitable for hardware realization owing to the computational complexity of the check-node processing that causes large decoding delay.

Complexity can be reduced by an approximation of the check-node calculation where the low-complexity minimum operations take the place of the hyperbolic tangent functions. This approximated version called min-sum algorithm (MSA) can obtain great complexity reduction with slight performance degradation [7]. Many researches have been devoted to increasing its performance by adding acceptable complexity to decoding.

The scaled min-sum algorithm [8] [9] uses a fixed scaling factor to compensate the oversized messages to some extent. Though scaled MSA has a noticeable performance improvement over the MSA, this approach is no longer efficient for irregular LDPC codes since irregular codes require different scaling factors for nodes with different degrees. Some researches suggested that an additional adaptive rule for updating scaling factor should be incorporated into the iterative process [10] [11]. In [10], the scaling factor is computed as a function of the number of the decoding iterations. It has been shown that an adjustable scaling factor gives a better performance than the fixed scaling factor. The decoding strategy in [11] considers different scaling factors for the output of both the variable node and the check node based on their degrees. However, the required multi-dimension optimization adds to design complexity.

When it comes to hardware realization, the decoding performance will suffer critical performance degradation since message quantization exacerbates the error floor [12]. In [12], the modified algorithm employs the dual quantizers with two domains that fit the operating regions of the input and output of the variable nodes. The quasi-uniform quantization in [13] applies different quantization step sizes based on the magnitude of messages. It uses the uniform quantization step sizes for the messages of small magnitude and the exponentially increasing step sizes for the messages of large magnitude. The early error floor induced by clipping under coarse quantization was mentioned in [17] and can be improved by applying the scaling factor to the magnitude of messages according to the correction step.

In our work, we further study the effect of clipping since the clipping is an important part of quantization. The results of simulations show that the optimal clipping threshold increases as BER decreases. Based on these results, a new decoding algorithm using adaptive clipping thresholds per-iteration is proposed. Lastly, we verify the proposed decoding algorithm over the AWGN channel. And our results show that the proposed algorithm under two-bit precision can achieve the

Dr. Qiu's work is partially supported by N.S.F. of China under Grant No.61571296 and N.S.F. of US under Grant No. CNS-1247778, No.CNS-1619250.

performance close to the unquantized SPA.

II. Problem Formulation

A. System Model

1) Clipping: Let $\mathbf{H} \in \mathbb{F}_2^{M\times N}$ be a binary LDPC parity-check matrix with M rows and N columns and let $\mathcal{C}$ be a binary LDPC code defined by

$$\mathcal{C} = \left\{ \mathbf{c} \in \mathbb{F}_2^N \left| \mathbf{H}\mathbf{c}^{\mathrm{T}} = \mathbf{0} \right. \right\} \tag{1}$$

$\mathcal{N}(m) = \{n : h_{m,n} = 1\}$ represents the variable nodes connected to the check node m and $\mathcal{M}(n) = \{m : h_{m,n} = 1\}$ represents the check nodes connected to the variable node n. We denote $l_{n,m}^i$ ($l_{m,n}^i$) by the message passed from variable node n (check node m) to check node m (variable node n) at iteration i.

We assume a binary phase-shift keying (BPSK) modulation where the binary baseband signal $\{0,1\}$ are represented by $\{1,-1\}$ and an additive white Gaussian noise (AWGN) channel. The channel output signal y_v is used to calculate the input log-likelihood ratio (LLR) defined as l_n^{ch} of the decoder. In decoding stage, the messages updating in the check nodes participate in the variable-node updating iteratively until the stopping criterion is reached. The decoding steps of the SPA and the MSA are almost the same except for the check-node updating which are respectively shown as (2) and (3):

$$l_{m,n}^i = 2\tanh^{-1}\left(\prod_{n' \in \mathcal{N}(m)\backslash n} \tanh\left(\frac{1}{2}\left| l_{n',m}^i \right| \right) \right) \tag{2}$$

$$p = \prod_{n' \in \mathcal{N}(m)\backslash n} \mathrm{sgn}\left(l_{n',m}^i \right) \cdot \min_{n' \in \mathcal{N}(m)\backslash n} \left| l_{n',m}^i \right| \tag{3}$$

Thus, the check-node message in the MSA is the overestimation of that in the SPA [7]. That is,

$$2\tanh^{-1}\left(\prod_{n' \in \mathcal{N}(m)\backslash n} \tanh\left(\frac{1}{2}\left| l_{n',m} \right| \right) \right) < \min_{n' \in \mathcal{N}(m)\backslash n} \left| l_{n',m} \right| \tag{4}$$

The modification of the MSA such as normalizing and offsetting focus on narrowing the check-node messages to make up for the overestimation [9]. Similarly, clipping on the variable-node messages can also improve the performance of the MSA by reducing the variable-node messages, therefore reducing the check-node messages at the next iteration. However, clipping draws our attention rather than normalizing or offsetting since it plays an important role in quantization.

The operation of clipping on variable-node message is defined as below:

$$\mathrm{clip}_\gamma\left(l_{n,m}^i \right) = \mathrm{sgn}\left(l_{n,m}^i \right) \cdot \min\left(l_{n,m}^i, \gamma \right) \tag{5}$$

where γ described as clipping factor is a positive constant.

2) Quantization: Considering the hardware realization, the node information passing in the decoding process will be quantized every iteration. Here, we use an uniform quantizer [13]–[16]with quantization step size Δ,

$$Q_\Delta(x) = \mathrm{sgn}(x) \cdot \left(\Delta \left\lfloor \left| \frac{|x|}{\Delta} - \frac{1}{2} \right| \right\rfloor + \frac{\Delta}{2} \right) \tag{6}$$

In the case that the clipping and the uniform quantization are applied simultaneously, we transform (6) into (7) where the quantization step size Δ is calculated from the quantization resolution q and the clipping threshold γ by $\Delta_{q,\gamma} = \frac{2\gamma}{2^q - 1}$,

$$Q_{q,\gamma}(x) = \begin{cases} \mathrm{sgn}(x) \cdot \left(\Delta \left\lfloor \left| \frac{|x|}{\Delta} - \frac{1}{2} \right| \right\rfloor + \frac{\Delta}{2} \right), & |x| < \gamma \\ \gamma, & |x| \geq \gamma \end{cases} \tag{7}$$

the operation of clipping and quantization applied on variable-node message and check-node message is defined as (8) (9), respectively,

$$l_{n,m}^i \to Q_{q,\gamma}\left(l_{n,m}^i \right) \tag{8}$$

$$l_{m,n}^i \to Q_{q,\gamma}\left(l_{m,n}^i \right) \tag{9}$$

B. Error Floor Phenomenon

Although clipping can compensate for the performance loss in the MSA, it causes an early error floor where the decoding performance degrades in the high signal-noise ratio (SNR) region [17]. The early error floor is shown in Fig.1. The BER curve of the clipped MSA falls quickly as the SNR condition becomes better. However the performance flattens in the SNR region higher than 2.5dB which is known as the error floor region. And it becomes worse than MSA at SNR of 2.75dB.

Based on the observations above, more results are simulated over a wide range of SNR with different clipping factors to study the effect of clipping. Fig.2 shows that the performance of the higher clipping threshold is worse at the high BER but improves rapidly as BER decreases which leads to a better performance at the low BER. In short, scaling up the clipping threshold can lower the error floor effect.

Furthermore, Fig.2 demonstrates that the optimal clipping factor increases as the BER decreases. Such results were also shown in [17] and we utilize to design the adaptive rule for the clipping factors that we will introduce below.

III. The Proposed Algorithm

A. The Modified Clipping MSA

The MS decoder with adaptive scaling factor can achieve better performance than that with the fixed scaling factor [10] [11]. Inspired by these works, we design an adaptive rule for the clipped MSA. As mentioned before, we apply a higher clipping threshold as the BER of the decoding output decreases per-iteration. And the clipping factor is chosen according to the number of the unsatisfied checks in our algorithm since it is correlated with the number of errors in the codeword [18].

We give an example of the proposed algorithm under finite precision that chooses one out of the three factors to clip the node messages per-iteration:

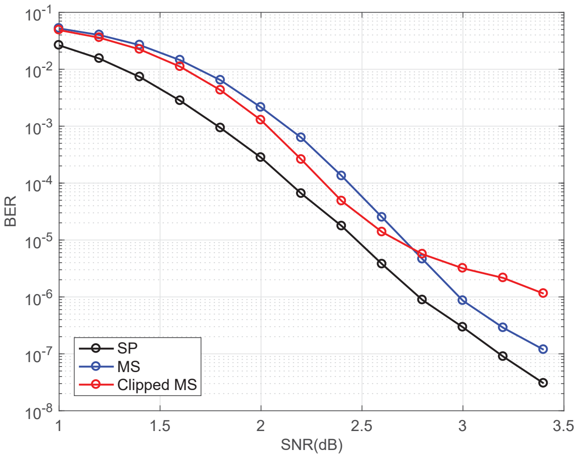

Fig. 1: The early error floor caused by clipping.

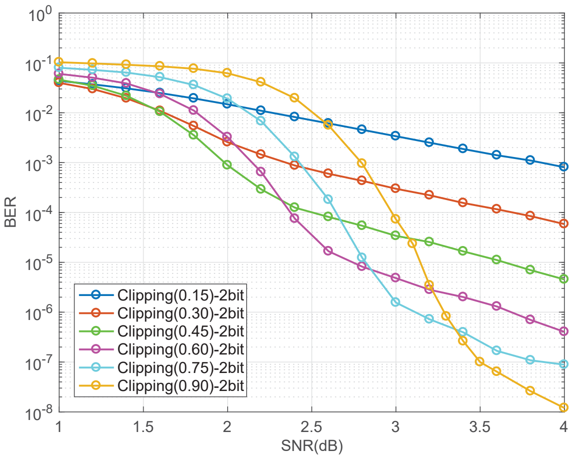

Fig. 2: The BER performance of the MSA under different clipping thresholds over AWGN channel.

1) Initialization: Set the iteration number $i = 1$. The initial values of LLR can be obtained from the channel output.

$$l_{n,m}^{0} = Q_{q,\gamma}\left(l_{n}^{ch}\right) = Q_{q,\gamma}\left(y_{v}\right) \tag{10}$$

2) Check node processing:

$$l_{m,n}^{i} = \prod_{n' \in \mathcal{N}(m) \backslash n} \operatorname{sgn}\left(l_{n',m}^{i-1}\right) \cdot \min_{n' \in \mathcal{N}(m) \backslash n} \left|l_{n',m}^{i-1}\right| \tag{11}$$

3) Posterior information calculating:

$$l_{n}^{i} = l_{n}^{ch} + \sum_{m' \in \mathcal{M}(n)} l_{m',n}^{i} \tag{12}$$

4) Hard decision:

$$\hat{\mathbf{x}} = \begin{cases} 0, & \operatorname{sgn}\left(l_{n}^{i}\right) > 0 \\ 1, & \operatorname{sgn}\left(l_{n}^{i}\right) < 0 \end{cases} \tag{13}$$

If all checks are satisfied i.e. $\mathbf{H}\hat{\mathbf{x}}^{\mathrm{T}} = \mathbf{0}$ or the maximum number of iterations I is reached, terminate the decoding and output $\hat{\mathbf{x}}$. Otherwise, choose corresponding clipping factor according to:

$$\gamma = \begin{cases} \gamma_1, & if \quad 0 < \mathbf{1}^{M} \cdot \mathbf{H}\hat{\mathbf{x}}^{\mathrm{T}} \leq \tau_1 \\ \gamma_2 & if \quad \tau_1 < \mathbf{1}^{M} \cdot \mathbf{H}\hat{\mathbf{x}}^{\mathrm{T}} \leq \tau_2 \\ \gamma_3 & if \quad \tau_2 < \mathbf{1}^{M} \cdot \mathbf{H}\hat{\mathbf{x}}^{\mathrm{T}} \leq M \end{cases} \tag{14}$$

where τ_1 and τ_2 are the two decision thresholds for the selection of three clipping factors γ_1, γ_2, γ_3.

5) Variable node processing:

$$l_{n,m}^{i} = l_{n}^{i} - l_{m,n}^{i} \tag{15}$$

$$l_{n,m}^{i} \rightarrow Q_{q,\gamma}\left(l_{n,m}^{i}\right) \tag{16}$$

where γ described as clipping threshold is a positive constant and q represents quantization resolution.

Go to step 2) and increase i by one.

Note that, the quantized variable-node message that belongs to quantization labels $\{\pm Q_1, \pm Q_2, \cdots, \pm Q_{2^{q-1}}\}$ is a q-bit message. And the check-node message is also a q-bit message belongs to $\{\pm Q_1, \pm Q_2, \cdots, \pm Q_{2^{q-1}}\}$ according to (11). Therefore, we omit the quantization on the check-node message to make the algorithm concise.

B. The Analysis of Performance and Complexity

By choosing the optimal clipping factor at different BER, the proposed algorithm improves the overall performance over a wide range of SNR. The study on the density evolution of the node message show that the magnitude of the message increases as the decoding process iteratively executes [13]. In comparison with the clipped MSA which uses a constant clipping factor, the proposed algorithm scales up the clipping factor iteratively to avoid the clipping distortion and decrease the quantization noise.

From the aspect of complexity, the calculation of the number of the unsatisfied checks in the hard decision step in our proposed algorithm adds little complexity. Compared with the algorithms in [10] [11] which require extra storage for different scaling factors per-iteration. However, in our proposed algorithm we do not need to store sequences of scaling factors for each iteration and only a few clipping factors, for instance 3 or 4 in our simulations can achieve noticeable improvement.

IV. Simulations

We experiment with the irregular LDPC codes with parameters $(n = 576, m = 432)$ and $(n = 1152, m = 864)$ taken from [19] according to standard IEEE 802.16e. These codes are labeled as I [(576, 432) LDPC code] and II [(1152, 864) LDPC code] for future reference. The optional clipping thresholds $[\gamma_1, \gamma_2, \gamma_3]$ for code I and code II are set to be $[0.75, 0.60, 0.45]$ and $[0.90, 0.60, 0.45]$ .The predefined thresholds $[\tau_1, \tau_2]$ for both code I and II are $[10, 20]$. The maximum number of iterations is set to be 10.

A. Quantization Resolution

In this part, we verify the performance enhancement of the proposed algorithm for different LDPC codes under two-bit precision and infinite precision.

The results of code I are shown in Fig.3. The adaptive clipping threshold outperforms the fixed clipping threshold under two-bit precision. To be more specific, when BER is 10^{-4}, the proposed algorithm has similar performance to the best clipped MSA with $\gamma = 0.45$. It achieves a 0.1-dB performance gain compared with the best clipped MSA with $\gamma = 0.60$ and even gets close to the unquantized SPA at BER of 10^{-6}. It improves the decoding performance by approximately 0.2 dB than the best clipped MSA with $\gamma = 0.75$ at BER of 10^{-7}. Under infinite precision, the proposed algorithm has better performance than the MSA over a wide range of BER. At BER of 10^{-6}, a significant performance gain of about 0.4 dB compared with the MSA is observed for the proposed algorithm.

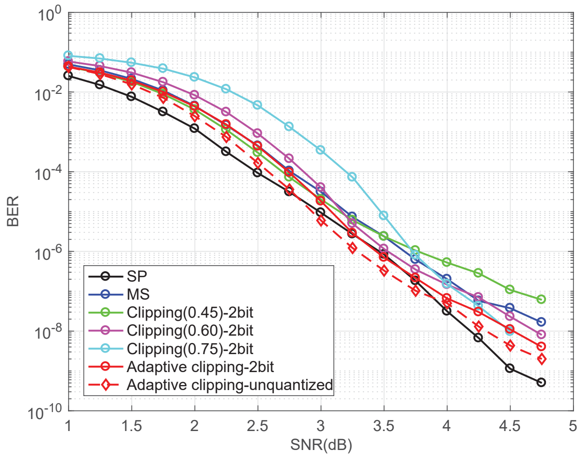

Fig. 3: The BER performance for $(576, 432)$ LDPC code under finite and infinite precision.

B. Code Length

In this part, we study the performance of the proposed algorithm for code II with larger block length and the results are shown in Fig.4. It demonstrates that the overall performance of the decoder with adaptive clipping threshold is remarkably improved compared with the other three decoders with fixed clipping threshold under two-bit precision. Especially at the low BER, for instance, at BER of 10^{-6}, the proposed algorithm improves the decoding performance by approximately 0.5 dB than the best clipped MSA with $\gamma = 0.90$. Under infinite precision, the proposed algorithm has good error correcting performance that approaches the SPA when BER is $10^{-5} \sim 10^{-8}$. It tells that the proposed algorithm is effective for different LDPC codes of short or medium length.

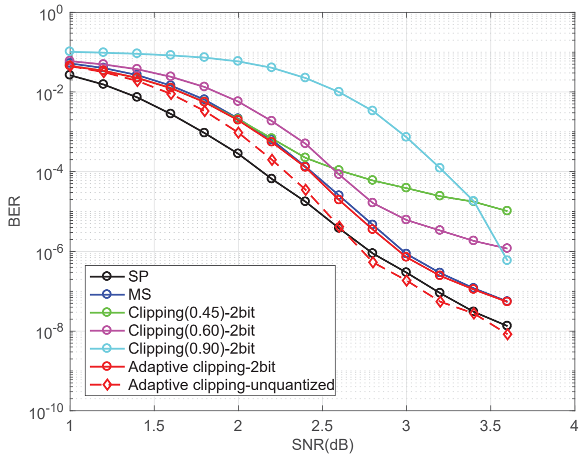

Fig. 4: The BER performance for $(1152, 864)$ LDPC code under finite and infinite precision.

C. Clipping Thresholds

In this part, we study the effect of the number of the optional clipping factors. The performance of the 4 clipping factors $[0.90, 0.75, 0.60, 0.45]$ compared with the 3 clipping factors $[0.75, 0.60, 0.45]$ for code I is shown in Fig.5. The proposed algorithm with 3 optional clipping factors somehow alleviates the error floor but the performance flattening can still be observed when SNR > 3.75dB. The proposed algorithm with 4 clipping factors has a better performance than 3 clipping factors especially in the error floor region. In addition, not only the error floor is further lowered but the performance before the error floor region has a slight improvement. In conclusion, the proposed algorithm is easy to expand capability by adding one or two more clipping factors with little complexity increased.

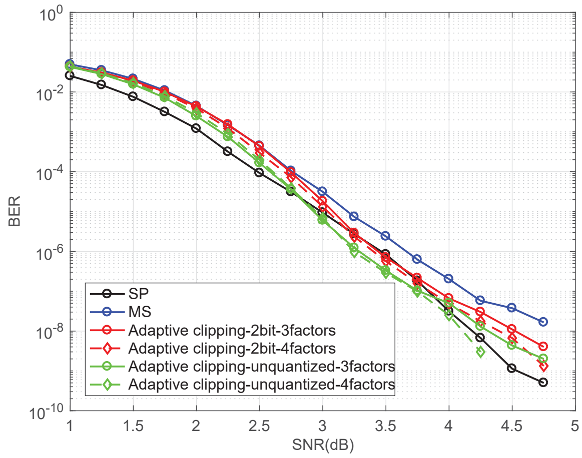

Fig. 5: The BER performance for $(576, 432)$ LDPC code under different numbers of optional clipping thresholds.

V. Conclusions

In this paper, we have investigated that clipping can improve the performance of the MSA but causes an early error floor. By simply adding an extra adaptive step that calculating the number of unsatisfied check nodes for the selection of the clipping thresholds, the early error floor has been effectively lowered. The proposed algorithm with low complexity has been tested under different conditions. Numerical results have shown that the proposed algorithm has an overall performance improvement compared with the conventional MSA under finite and infinite precision. And it can achieve the performance near the unquantized SPA within few iterations under two-bit precision, indicating that the proposed decoding algorithm is a promising technique for the 5G systems with low latency and low power consumption as demand.

In the future work, we will focus on the optimization for the clipping factors. In addition, the performance of the proposed algorithm will be verified in MIMO systems [20].

References

[1] R. Maunder, "The 5G channel code contenders," White Paper, AccelerComm Ltd., August 2016.

[2] R. Qiu and M. Wicks, Cognitive Networked Sensing and Big Data. New York, NY, USA: Springer, 2013.

[3] L. Chu, R. Qiu, and F. Wen, "Eigen-Inference Precoding for Coarsely Quantized Massive MU-MIMO System with Imperfect CSI," To appear in IEEE Trans. Veh. Technol., 2019.

[4] R. G. Gallager, "Low-density parity-check codes," IRE Trans. Inf. Theory, vol. IT-8, pp. 21-28, Jan. 1962.

[5] D. J. C. MacKay and R. M. Neal, "Near Shannon limit performance of low-density parity-check codes," Electron. Lett., vol. 32, pp. 1645-1646, Aug. 1996.

[6] T. Richardson and R. Urbanke, "The capacity of low-density parity-check codes under message-passing decoding," IEEE Trans. Inform. Theory, vol. 47, no. 2, pp. 599-618, Feb. 2001.

[7] J. Chen, A. Dholakia, E. Eleftheriou, M. P. C. Fossorier and X.-Y. Hu, "Reduced-complexity decoding of LDPC codes," IEEE Trans. Commun., vol. 53, no. 8, pp. 1288-1299, Aug. 2005.

[8] J. Chen and M. Fossorier, "Density evolution for two improved BP Based decoding algorithms of LDPC codes," IEEE Commun. Lett., vol. 6, no. 5, pp. 208-210, May. 2002.

[9] G. Lechner and J. Sayir, "Improved sum-min decoding of LDPC codes," in Proc. IEEE Int. Symp. Information Theory and its Applications, 2004.

[10] G. Lechner and J. Sayir, "Improved sum-min decoding for irregualr LDPC codes," in Proc. International Symposium on Turbo Codes and Related Topics, Apr. 2006.

[11] J. Zhang, M. Fossorier, and D. Gu, "Two-dimensional correction for min-sum decoding of irregular ldpc codes," IEEE Commun. Lett., vol. 10, no. 3, pp. 180-182, Mar. 2006.

[12] Z. Zhang, L. Dolecek, B. Nikolic, V. Anantharam, and M. J.Wainwright, "Design of LDPC decoders for improved low error rate performance: quantization and algorithm choices," IEEE Trans. Commun., vol. 57, no. 11, pp. 3258-3268, Nov. 2009.

[13] X. Zhang and P. H. Siegel, "Quantized iterative message passing decoders with low error floor for LDPC codes," IEEE Trans. Commun., no. 1, pp. 1-14, Jan. 2014.

[14] L. Chu, F. Wen, R. Qiu, "Robust Precoding Design for Coarsely Quantized MU-MIMO Under Channel Uncertainties," To appear in IEEE ICC, 2019.

[15] L. Chu, F. Wen, R. Qiu, "Efficient nonlinear precoding for massive MU-MIMO downlink systems with 1-bit DACs," To appear in IEEE Trans. Wireless Commun., 2019.

[16] F. Wen, L. Chu, Liu P, et al., " A survey on nonconvex regularization-based sparse and low-rank recovery in signal processing, statistics, and machine learning," IEEE Access, no. 6, pp. 69883-69906, 2019.

[17] J. Zhao, F. Zarkeshvari, and A. H. Banihashemi, "On implementation of min-sum algorithm and its modifications for decoding LDPC codes," IEEE Trans. Commun., vol. 53, no. 4, pp. 549-554, Apr. 2005.

[18] I. Adjudeanu, J.-Y. Chouinard, and P. Fortier, "On the correlation between error weights and syndrome weights for belief propagation decoding of LDPC codes," in Proc. IEEE Can. Workshop Inf. Theory, pp. 36-41, May. 2009.

[19] M. Helmling and S. Scholl, "Database of channel codes and ML simulation results," Wimax 3/4A, University of Kaiserslautern, Kaiserslautern, Germany, 2016. [Online]. Available: www.uni-kl.de/channel-codes.

[20] E. G. Larsson, O. Edfors, and T. L. Marzetta, "Massive MIMO for next generation wireless systems," IEEE Commun. Mag., vol. 52, no. 2, pp. 186-195, Feb. 2014.