SPARC-LDPC Coding for MIMO Massive Unsourced Random Access

Tianya Li, Yongpeng Wu, Mengfan Zheng, Dongming Wang, and Wenjun Zhang

Abstract—A joint sparse-regression-code (SPARC) and low-density-parity-check (LDPC) coding scheme for multiple-input multiple-output (MIMO) massive unsourced random access (URA) is proposed in this paper. Different from the state-of-the-art covariance based maximum likelihood (CB-ML) detection scheme, we first split users’ messages into two parts. The former part is encoded by SPARCs and tasked to recover part of the number of active users as well as the corresponding interleaving patterns by compressed sensing. The latter part is encoded by LDPC codes and then interleaved by the inner decoder adapts a non-Bayesian method (i.e., ML detection) based on the covariance of the received signal. Moreover, our scheme has higher spectral efficiency (nearly 15 times larger) than CB-ML as we only split messages into two parts.

Index Terms—unsourced random access, MIMO, compressed sensing, belief propagation, LDPC

I. INTRODUCTION

Future wireless cellular networks aim to support massive connectivity scenarios such as Internet-of-Things (IoT) and massive machine-type communications (mMTC), etc. A key feature of these scenarios is that the cellular base station (BS) needs to serve a massive number of users or devices [1]. However, the pilot-based scheme investigated in [2] will cause a waste of pilot resources because of the massive number of users and users’ sporadic traffic.

One solution to this situation is the unsourced random access (URA) scheme, which is first proposed in [3]. In URA, all the users share a common codebook and choose codewords from the codebook as their messages based on a specific strategy, which can accommodate a large number of potential users. The BS only recovers the list of transmitted messages regardless of the corresponding users’ IDs, thus leading to the so-called unsourced property. There has been some related work in additive white Gaussian multiple access channel (GMAC) and fading channel [4]–[8]. In [4], a low complexity coding scheme is proposed which requires a lower energy-per-bit than some traditional schemes such as ALOHA and treating interference as noise (TIN). A cascaded system is proposed in [5], which includes inner and outer codes. To reduce the complexity, the outer tree encoder [6] first splits the data into several slots with a large number of parity bits added. Within each slot, the messages are encoded by the sparse regression codes (SPARCs). The inner compressed sensing (CS) based AMP decoder recovers the slot-wise transmitted messages. The outer tree decoder then stitches the decoded messages across different slots according to the prearranged parity. The complexity of the tree decoder increases exponentially with the number of slots. In [7], the user’s data is divided into two parts. The former part acts as a preamble sequence for recovering the number of active users as well as the corresponding interleaving patterns. The latter part is a low-density-parity-check (LDPC) coded interleave-division multiple access (IDMA) scheme. In [8], a joint fading coefficient estimation and LDPC decoding scheme is proposed based on BP.

Besides the above works, the study of massive URA in multiple-input multiple-output (MIMO) system has also drawn increasing attention. The BS equipped with multiple antennas adds extra dimensions for the received signal, thus contributing to the signal detection and the access of a massive number of users. In [9], [10], a covariance-based maximum likelihood (CB-ML) scheme is proposed in MIMO massive URA system. The CB-ML scheme also adapts a concatenated coding scheme with the aforementioned tree code as the outer code, of which the inner decoder adapts a non-Bayesian method (i.e., ML detection) based on the covariance of the received signal. The covariance-based property of CB-ML makes it unable to acquire channel information and thus unable to combine with LDPC. The performance of the above CB-ML scheme degrades dramatically when the number of antennas is less than that of the active users. Besides, the CB-ML scheme has low spectral efficiency and high complexity because of the tree code. In [11], users’ messages are split into several slots without adding parity bits. By clustering together the slot-wise recovered channels, the decoded slot-wise messages can be...
stitched correspondingly. However, the users’ channels in [11] are assumed to remain constant over all the slots (i.e., 6798 channel uses in their simulation). This assumption is difficult to hold in practice. In this paper, we propose an improved scheme with enhanced spectral efficiency and performance as well as low complexity for MIMO massive URA.

A. Contributions

Motivated by all the aforementioned works, we propose a joint SPARC-LDPC coding scheme for MIMO massive URA, which can accommodate a large number of active users with a reasonable antenna array size at the BS. In our scheme, we first split the information into two parts. The former part is encoded by means of SPARC to pick codewords from a fixed codebook. The AMP decoder recovers information in the former part and the corresponding channel coefficients as well as interleaving patterns. The latter part is encoded by LDPC codes and then interleaved by the IDMA scheme. With the channel coefficients and interleaving patterns estimated by the former part, information in the latter part can be obtained based on BP joint with successive interference cancellation (SIC).

Compared with the state-of-the-art CB-ML scheme investigated in [9], [10], our scheme has better performance when the number of antennas at the BS is smaller than that of active users. The complexity of our scheme is with the order $O\left(2^L \cdot ML + KM\right)$ and lower than the CB-ML scheme. Moreover, our scheme has higher spectral efficiency (i.e., 15 times larger) as we only split messages into two parts.

II. SYSTEM MODEL

Consider the uplink of a single-cell cellular network consisting of $K_{\text{tot}}$ potential users, in which $K_a$ users are active in a slot. The BS is equipped with $M$ antennas. Each user is equipped with a single antenna and has $B$ bits of information to be coded and transmitted in a block-fading channel. As mentioned above, all the users share a common codebook in URA, denoted by $A \in \mathbb{C}^{L \times 2^B}$. The power of each codeword $a_k$ is constraint to $L$, i.e., $\|a_k\|_2^2 = L$. Let $i_k \in [1:2^B]$ denote the message indice of user $k$. Then $a_{i_k}$ is the coded message of user $k$. In this work, the map between messages and codewords is a SPARC scheme. The main idea of SPARC is to map the information to a sparse vector and then choose the corresponding codeword from a fixed codebook according the sparse vector. The corresponding received signal can be written as

$$Y = \sum_{k \in K_a} a_{i_k} h_{i_k}^T + Z = A \Phi H + Z$$

(1)

where $\Phi \in \{0, 1\}^{2^B \times 2^B}$ denotes the diagonal binary selection matrix. The positions of one in $\Phi$ indicate that there are users whose messages are mapped there. However, the users’ IDs are unknown to the BS, thus leading to the so-called unsourced property. Notice $\Phi$ is a sparse matrix which is a consequence of the SPARC and can be recovered by CS techniques. $H \in \mathbb{C}^{2^B \times M}$ denotes the MIMO channel coefficient matrix. $Z \in \mathbb{C}^{L \times M}$ denotes additive white Gaussian noise (AWGN) and is distributed as $CN\left(0, \sigma^2 I\right)$.

Let $L$ denote the set of recovered messages at the BS. The performance in URA is evaluated by the probability of misdetection and false alarm, denoted by $p_{md}$ and $p_{fa}$ respectively, which are given by:

$$p_{md} = \frac{1}{K_a} \sum_{k \in K_a} P(a_{i_k} \notin L)$$

(2)

$$p_{fa} = \frac{|L \setminus \{a_{i_k} : k \in K_a\}|}{|L|}$$

(3)

where $|\cdot|$ denotes the Hamming weight. In this system, the code rate $R = \frac{B}{L}$ and the spectral efficiency $\mu = \frac{B \cdot K_a}{L \cdot M}$.

III. PROPOSED SCHEME

Notice the dimension of the codebook $A$ increases exponentially with the information length $B$. To reduce the complexity, a tree code [6] is implied to split the information into several slots in the CB-ML scheme [10]. As mentioned above, the tree code greatly reduces the code rate and spectral efficiency as a large number of parity bits are added during the encoding process. Besides, the complexity of the tree decoding process increases exponentially with the number of slots. In what follows below, we propose a low-complexity joint SPARC-LDPC coding scheme which keeps the spectral efficiency in a relatively high level.

A. Encoder

Motivated by [7], we divide the $B$ bits message into two parts, where the former part is coded with SPARC and tasked to recover part of the messages, the number of active users, the interleaving patterns and most importantly, the MIMO channel based on CS. The latter part is coded with LDPC codes and can be decoded by BP with the channel estimated by the former part. For clarity, we denote the former and latter parts as CS and LDPC parts, respectively. The total $B$ bits are split into two parts of $B_p$ and $B_c$, respectively, where $B_p \ll B_c$. Correspondingly, the total $L$ channel uses are split into $L_p$ and $L_c$, respectively, where $B_p + B_c = B$ and $L_p + L_c = L$. Compared with the existing scheme of [10], to support the same amount of users, the required number of channel uses $L$ in our scheme is much shorter, resulting in higher spectral efficiency. An example is shown in Section IV.

For the CS encoding part, we use SPARCs to construct a sparse selection matrix $\Phi$ as shown in [1]. Each user chooses a codeword from the codebook $A = [a_1, a_2, ..., a_{2^B}] \in \mathbb{C}^{L_p \times 2^B_p}$. Let $i_k \in [1:2^B_p]$ denote the message indice of user $k$. Then the $B_p$ bits of user $k$ are coded to the codeword $a_{i_k}$ with power constraint $\|a_{i_k}\|_2^2 = L_p$. A user’s message indice $i_k$ and the $B_p$ bits of information is a one-to-one mapping. If $i_k$ is recovered then the corresponding $B_p$ bits of information can be recovered automatically. Users who has the same $B_p$ bits of information will choose the same codewords, in which case their corresponding channel coefficients cannot be estimated properly. This collision has been considered in our numerical results.
For the LDPC encoding part, user $k$’s $B_p$ bits of information are encoded to an $(L_c, B_c)$ LDPC code $b \in \{0, 1\}^{L_c \times 1}$ and then modulated by binary phase shift keying (BPSK), denoted by $s \in \{1, -1\}^{L_c \times 1}$. Then $s$ is interleaved by a random interleaver $\pi_{i_k}$ with interleaving pattern $i_k$. This is nearly an IDMA scheme because the interleaving patterns for most of users are different as the message indices $i_k$ are different. The above encoding scheme is illustrated in Fig. 1. After the two parts of encoding, the final coded message of user $k$ is $v = [a^T_{i_k}, \pi^T_{i_k}(s)]^T$. The received signal in this coding scheme is $Y = VH + Z$, where $V = [v_1, v_2, ..., v_{K_a}] \in \mathbb{C}^{L \times K_a}$ denotes users’ coded messages.

There are edges in the factor graph which represent the connections between nodes. The edges between check nodes and variable nodes are determined by the LDPC check matrix which cannot be clearly marked in the graph. The edges between variable nodes and observation nodes are simply determined by (7) though looking complicated. For example, the observation node $y_{1,1}$ is connected to the first variable nodes of all users (i.e., $s_{k,1}, k = 1, 2, ..., K$). Correspondingly, the variable node $s_{1,1}$ is connected to the first observation nodes from all antennas (i.e., $y_{m,1}, m = 1, 2, ..., M$). In BP algorithm, messages are passed along these edges. The types of messages are listed below:

- $R_{k,n} \rightarrow c_{k,n}$: Messages from check node $c_{k,n}$ to variable node $s_{k,l}$.
- $Q_{k,l} \rightarrow n_{k,l}$: Messages from variable node $s_{k,l}$ to check node $c_{k,n}$.
- $P_{k \rightarrow m,l}$: Messages from variable node $s_{k,l}$ to observation node $y_{m,l}$.
- $\Lambda_{m \rightarrow k,l}$: Messages from observation node $y_{m,l}$ to variable node $s_{k,l}$.

Note that interleaving and deinterleaving are needed for $P_{k \rightarrow m,l}$ and $\Lambda_{m \rightarrow k,l}$, respectively, of which the patterns can be obtained from the CS decoder. The interleaver is the connection between SPARC and LDPC. If the LDPC decoder uses the interleaver $\pi_{i_k}$ and successfully decodes the $B_c$ bits of message, the corresponding former $B_p$ bits of message can be obtained by mapping from $i_k$. Here we give details of the updating scheme of the messages for BPSK modulated system, which are listed above.

$$\Lambda_{m \rightarrow k,l} = \log \frac{P(y_{m,l} \mid H, s_{k,l} = 1)}{P(y_{m,l} \mid H, s_{k,l} = -1)} = \frac{2}{\sigma_{z_{k,l}}} \mathcal{R}(h^*_{m,k} (y_{m,l} - \mu_{z_{k,l}}))$$ (8)
where $\mathbf{H}$ is the channel matrix estimated in the CS decoding part and $\mathcal{R} (\cdot)$ denotes the real part of a complex number. In fact, $\Lambda_{m \rightarrow k,l}$ is the log-likelihood ratio (LLR) of variable node $s_{k,l}$ observed at node $y_{m,l}$. This is a TIN scheme as the received signal $y_{m,l}$ is given by

$$y_{m,l} = \sum_{j \in \mathcal{K}} h_{m,j} s_{j,l} + n_{m,l}$$

(9)

where $n_{m,l}$ is the Gaussian noise with zero mean and variance $\sigma_n^2$, and $\mathcal{K}\setminus k$ denotes all the numbers in range $[1:K]$ except $k$. In what follows, we use $[M]$ to denote the set of integers $\{1,2,\ldots,M\}$. The Gaussian noise $n_{m,l}$ and the interference from other users are all treated as noise denoted by $z_{k,l}$, which is a Gaussian variable with mean $\mu_{z_{k,l}}$ and variance $\sigma_{z_{k,l}}^2$ given below.

$$\mu_{z_{k,l}} = \sum_{j \in \mathcal{K}\setminus k} h_{m,j} (2P_{j \rightarrow m,l} - 1)$$

(10)

$$\sigma_{z_{k,l}}^2 = 4 \sum_{j \in \mathcal{K}\setminus k} |h_{m,j}|^2 (1 - P_{j \rightarrow m,l}) P_{j \rightarrow m,l} + \sigma_n^2$$

(11)

where $P_{j \rightarrow m,l}$ denotes the probability of $s_{j,l} = 1$, and is initialized to 0.5. The update of $P_{j \rightarrow m,l}$ is given by

$$P_{k \rightarrow m,l} = \frac{\exp \left( \sum_{j \in [M] \setminus m} \Lambda_{j \rightarrow k,l} + \sum_{j \in \mathcal{N}_c(k,l)} R_{k,j \rightarrow l} \right)}{1 + \exp \left( \sum_{j \in [M] \setminus m} \Lambda_{j \rightarrow k,l} + \sum_{j \in \mathcal{N}_c(k,l)} R_{k,j \rightarrow l} \right)}$$

(12)

The updates of other two messages in LDPC decoding process are given by

$$Q_{k,l \rightarrow n} = \sum_{j \in M} \Lambda_{j \rightarrow k,l} + \sum_{j \in \mathcal{N}_c(k,l) \setminus n} R_{k,j \rightarrow l}$$

(13)

$$R_{k,n \rightarrow l} = 2tanh^{-1} \left( \prod_{j \in \mathcal{N}_v(k,n) \setminus l} \tanh \left( \frac{Q_{k,j \rightarrow n}}{2} \right) \right)$$

(14)

where $\mathcal{N}_c(k,l) \setminus n$ denotes the set of check nodes connected to $s_{k,l}$ except $c_{k,n}$, and $\mathcal{N}_v(k,n) \setminus l$ denotes the set of variable nodes connected to $c_{k,n}$ except $s_{k,l}$. $R_{k,n \rightarrow l}$ is initialized to 0. The LLR of the variable node $s_{k,l}$ at the end of an iteration is given by

$$L_{k,l} = \sum_{j \in M} \Lambda_{j \rightarrow k,l} + \sum_{j \in \mathcal{N}_c(k,l)} R_{k,j \rightarrow l}.$$  

(15)

The decoded bit $\hat{s}_{k,l}$ is 1 when $L_{k,l} > 0$ and 0 otherwise. Let $\mathbf{S} \in \{0,1\}^{L_c \times K}$ and $\mathbf{H}_c \in \{0,1\}^{N \times L_c}$ denote the decoded messages and LDPC check matrix respectively. The LDPC decoding process ends when $\text{mod}(\mathbf{H}_c, \mathbf{S}, 2) = 0$ or the maximum number of iterations is reached. Note the estimated number of active users $K$ is not guaranteed to be equal to $K_a$. Therefore, not all the decoded messages in $\mathbf{S}$ satisfy the check. Let $\hat{S} = \{s_{k,l}, k \in \hat{K}\}$ denote the set of the successfully decoded messages and $\hat{K}$ is the set of the corresponding subscripts. Obviously, $|\hat{K}| \leq K_a \leq K$. Hence the LDPC part can reduce the probability of false alarm. Finally, the LDPC decoder outputs $\hat{S}$ and $\hat{K}$. To further improve the performance, we combine the LDPC decoder with SIC and we denote it by LDPC-SIC. LDPC-SIC works as follows.

Let $\hat{\mathbf{H}} \in \mathbb{C}^{K \times M}$ denote the channel matrix estimated by the CS decoder and $\mathcal{K}$ is the set of the corresponding subscripts. Let $\hat{\mathcal{S}}_0 = \emptyset$ and $\hat{K}_0 = \emptyset$ respectively denote the sets of decoded messages and the corresponding subscripts obtained by the LDPC decoder, which are initialized to empty sets. With $\mathbf{Y}_c$ and $\hat{\mathbf{H}}$, the LDPC decoder outputs the set of decoded messages $\hat{\mathcal{S}}$ and the corresponding subscripts $\hat{\mathcal{K}}$. Then we have $\hat{\mathcal{S}}_0 \leftarrow \hat{\mathcal{S}}_0 \cup \hat{\mathcal{S}}, \hat{K}_0 \leftarrow \hat{K}_0 \cup \hat{\mathcal{K}}$ and $\mathbf{H} = \hat{\mathbf{H}}_{k,l}$ for $k \in \mathcal{K}\setminus \hat{\mathcal{K}}_0$. The residual signal is updated by

$$\mathbf{Y} = \mathbf{Y}_c - \sum_{k \in \hat{\mathcal{K}}_0} (2s_k - 1) \mathbf{h}_k$$

(16)

where $s_k \in \{0,1\}^{L_c \times 1}$ is the $k$-th codeword in $\hat{\mathcal{S}}_0$. Note the sum needs to be BPSK modulated when calculating $\mathbf{Y}$. $\mathbf{h}_k$ is the $k$-th row vector in $\hat{\mathbf{H}}$. $\mathbf{Y}$ and $\hat{\mathbf{H}}$ are inputs to the LDPC decoder for next decoding. This iterative process ends when $\hat{\mathcal{K}} = \emptyset$ or $\mathcal{K}\setminus \hat{\mathcal{K}}_0 = \emptyset$. The overall decoding scheme is shown in Algorithm [1].

After the LDPC decoder outputs the subscript set of the decoded messages $\hat{\mathcal{K}}_0$, the stitching of two parts of messages is easy. Let $L = \{l_k, k \in \hat{\mathcal{K}}_0\}$ denotes the message indices.
Algorithm 1 Joint CS and LDPC-SIC decoder for URA

Input: $Y_p$, $A$
$\triangleright$ CS decoder
Output: $\hat{H}$, $L = \{i_k, k \in K\}$
Initialize: $Y = Y_c$, $H = \hat{H}$, $\hat{S}_0 \leftarrow \emptyset$, $\hat{K}_0 \leftarrow \emptyset$
repeat
Input: $Y$, $H$, noise var $\sigma_n^2$
Initialize: $P_{k \rightarrow m,l} = 0.5$, $R_{k,n,l} = 0$ for all $k \in [K]$, $m \in [M]$, $l \in [L_c]$ and $n \in [N]$.
$\triangleright$ LDPC decoder
Output: $S$, $\hat{K}$
$\hat{S}_0 \leftarrow \hat{S}_0 \cup S$, $\hat{K}_0 \leftarrow \hat{K}_0 \cup \hat{K}$
$H = \hat{H}_{k,:}$ for $k \in K \backslash \hat{K}_0$
$Y = Y_c - \sum_{k \in \hat{K}_0} (2s_k - 1) \hat{H}_{k,:}$
until $\hat{K} = \emptyset$ or $K \backslash \hat{K}_0 = \emptyset$
Return: $\hat{S}_0$, $\hat{K}_0$
corresponding to users in $\hat{K}_0$. $L$ can be directly mapped to the $B_p$ bits of information. Let $S$ denotes the set of $B_s$ bits of information by removing the redundant LDPC check bits of $S_0$. Then these two parts of information with the same subscript $k \in \hat{K}_0$ can be stitched together. Then the probability of misdetection and false alarm can be obtained by (2) and (3).

IV. NUMERICAL RESULTS

We compare by numerical results the performance of the proposed LDPC and LDPC-SIC schemes with the CB-ML scheme of [10] in various signal noise ratio (SNR) regions and with different number of antennas. The LDPC scheme refers to the LDPC decoder without SIC. The parameter settings in our simulation are shown in TABLE I. In the LDPC scheme, $(L_c, B_c)$ means the $B_c$ bits of information are coded to $L_c$ bits of LDPC code. The LDPC and LDPC-SIC decoders share the same parameter settings. Both are the $(3, 6)$-regular LDPC code and the code rate is 0.5. In the tree code scheme, the 96 bits of information are split into $S = 24$ slots with slot length $J = 16$ bits. The first slot has 16 bits of information and no parity bit. The following second to 21st slots have 4 bits of information and 12 bits of parity, and the final 3 slots have 16 bits of parity and no information bit. For tree encoding and decoding details, please refer to [6].

| TABLE I | PARAMETER SETTINGS |
|---------|-------------------|
| $E_b/N_0$ (dB) | CB-ML | LDPC, LDPC-SIC |
| M | 50 | 100 |
| $L_p$ | 300 |
| $B$ | 96 | 96, $B_p = 10$, $B_c = 80$ |
| $L_c$ | 200 |
| channel code | $S = 24$, $J = 16$ | $L_p + L_c = 500$ |
| code rate | 0.1920 |

As is mentioned above, our scheme requires a much smaller number of channel uses compared with the CB-ML scheme of [10]. As is shown in TABLE I the CB-ML scheme needs 7200 channel uses while our LDPC and LDPC-SIC schemes only need 500 channel uses. As a consequence, the code rate of the proposed LDPC scheme is nearly 15 times lager than that of the CB-ML scheme. Let $R$ denote the code rate. The energy per symbol $E_s$ of the coded messages is $E_s = E_b/R$. This is a fair comparison because both the transmitted data $B$ and the energy per bit $E_b$ are the same for CB-ML and our scheme. The large scale fading coefficient (LSFC) is set to 1 and is known to all the above schemes and the noise variance $N_0 = 1$. The empirical threshold for all algorithms except AMP is set to 0.5. AMP has its own activity detection threshold according to formula (42) in [12].

Fig. 3 shows how the error probability $P_e$ falls as a function of the energy per bit $E_b/N_0$ at $M = 50$. In Fig. 3 the proposed LDPC-SIC scheme outperforms the CB-ML scheme with a nearly 0.8 dB gap at $E_b/N_0 = 22.5$ dB, while the LDPC scheme without SIC is a little bit worse. Besides, the spectral efficiency of LDPC-SIC is 0.384 bps/Hz/RX which is nearly 15 times larger than CB-ML, the spectral efficiency of which is 0.027 bps/Hz/RX.

Fig. 4 shows how the error probability $P_e$ falls as a function of the number of antennas $M$ at $E_b/N_0 = 20$ dB. See TABLE I for other parameter settings.

As is mentioned above, our scheme requires a much smaller number of channel uses compared with the CB-ML scheme of [10]. As is shown in TABLE I the CB-ML scheme needs 7200 channel uses while our LDPC and LDPC-SIC schemes only need 500 channel uses. As a consequence, the code rate of the proposed LDPC scheme is nearly 15 times lager than that of the CB-ML scheme. Let $R$ denote the code rate. The energy per symbol $E_s$ of the coded messages is $E_s = E_b/R$. This is a fair comparison because both the transmitted data $B$ and the energy per bit $E_b$ are the same for CB-ML and our scheme. The large scale fading coefficient (LSFC) is set to 1 and is known to all the above schemes and the noise variance $N_0 = 1$. The empirical threshold for all algorithms except AMP is set to 0.5. AMP has its own activity detection threshold according to formula (42) in [12].

Fig. 3 shows how the error probability $P_e$ falls as a function of the energy per bit $E_b/N_0$ at $M = 50$. In Fig. 3 the proposed LDPC-SIC scheme outperforms the CB-ML scheme with a nearly 0.8 dB gap at $E_b/N_0 = 22.5$ dB, while the LDPC scheme without SIC is a little bit worse. Besides, the spectral efficiency of LDPC-SIC is 0.384 bps/Hz/RX which is nearly 15 times larger than CB-ML, the spectral efficiency of which is 0.027 bps/Hz/RX.

![Fig. 3](image1.png)

![Fig. 4](image2.png)
Floating Point Multiplications (FPM)

| Schemes   | Floating Point Multiplications (FPM)                      | Floating Point Additions (FPA)                      |
|-----------|-----------------------------------------------------------|----------------------------------------------------|
| CB-ML     | ML decoder: $2^J (3L^2 + 3L + 4) + ML^2 + L^2$           | $2^J (2L^2 + 1) + (M - 1) L^2$                     |
|           | Tree decoder: $K \left( \sum_{l=1}^{S} l^2 \right) + \sum_{l=1}^{S} l_{j=2}^{l_{j=2}} l_{j=1}^{l_{j=1}} \sum_{l=1}^{S} l_{j=2}^{l_{j=2}} + \sum_{l=1}^{S} l_{j=2}^{l_{j=2}} l_{j=1}^{l_{j=1}}$ | $K^{1-q} (K - 1) \prod_{l=1}^{q} a_{l=1}^{a_{l=1}}$ |
| LDPC      | AMP decoder: $2 \cdot 2^{B_p} ML + ML + 11 \cdot 2^{B_p} M + 4M + 2 \cdot 2^{B_p} + 5$ | $2^{B_p} ML + 4ML + 6 \cdot 2^{B_p} M - 2M - 2^{B_p}$ |
| LDPC-SIC  | LDPC decoder: $10KML + 2KN_{vc} + KM$                     | $(13K - 2) ML + KL (N_c - 1) + 2KN_{vc}$           |

Note: $K$ and $K$ are the number of active users estimated by the ML decoder and AMP decoder respectively. $N$ is the number of check nodes connected to a variable node and $N_c$ is the number of variable nodes connected to a check node.

V. Conclusion

In this paper, we propose a low-complexity SPARC-LDPC coding scheme for MIMO massive URA. Based on compressed sensing, belief propagation as well as successive interference cancellation, the proposed scheme outperforms the state-of-the-art CB-ML scheme when the number of active users is larger than that of the antennas at the BS. This is reasonable because the number of antennas at the BS is limited and less than the number of active users in a massive access scenario. The complexity of our scheme is with the order $O \left( 2^{B_p} ML + \bar{K}ML \right)$ and lower than the CB-ML scheme. Furthermore, our scheme is able to recover users’ information with a near 15 times higher spectral efficiency than the CB-ML scheme. Our future work is to avoid collisions through some scheduling measures and take a message passing method between SPARC and LDPC modules.

References

[1] Y. Wu, X. Gao, S. Zhou, W. Yang, Y. Polyanskiy, and G. Caire, “Massive access for future wireless communication systems,” to appear in IEEE Wireless Commun., Online Available: https://arxiv.org/abs/1910.12678v1.

[2] L. Liu and W. Yu, “Massive connectivity with massive MIMO—Part I: Device activity detection and channel estimation,” IEEE Trans. Signal Process., vol. 66, no. 11, pp. 2933–2946, 2018.

[3] Y. Polyanskiy, “A perspective on massive random-access,” in 2017 IEEE Int. Symp. Inform. Theory (ISIT), 2017, pp. 2523–2527.

[4] O. Ordentlich and Y. Polyanskiy, “Low complexity schemes for the random access Gaussian channel,” in 2017 IEEE Int. Symp. Inform. Theory (ISIT), 2017, pp. 2528–2532.

[5] A. Fengler, P. Jung, and G. Caire, “SPARCs and AMP for unsourced random access,” in 2019 IEEE Int. Symp. Inform. Theory (ISIT), 2019, pp. 2843–2847.

[6] V. K. Amalladinne, J.-F. Chamberland, and K. R. Narayanan, “A coded compressed sensing scheme for uncoordinated multiple access,” Online Available: https://arxiv.org/abs/1809.04745.

[7] A. Pradhan, V. Amalladinne, A. Vem, K. R. Narayanan, and J. Chamberland, “A joint graph based coding scheme for the unsourced random access Gaussian channel,” in 2019 IEEE Global Commun. Conf. (GLOBECOM), 2019, pp. 1–6.

[8] S. S. Kowshik, K. Andreiv, A. Frolov, and Y. Polyanskiy, “Energy efficient coded random access for the wireless uplink,” IEEE Trans. Commun., pp. 1–1, 2020.

[9] S. Haghighatshoor, P. Jung, and G. Caire, “Improved scaling law for activity detection in massive MIMO systems,” in 2018 IEEE Int. Symp. Inform. Theory (ISIT), 2018, pp. 381–385.

[10] A. Fengler, S. Haghighatshoor, P. Jung, and G. Caire, “Grant-free massive random access with a massive MIMO receiver,” in 2019 53rd Asilomar Conference on Signals, Systems, and Computers, 2019, pp. 23–30.

[11] V. Shiyianov, F. Bellili, A. Mezghani, and E. Hossain, “Massive unsourced random access based on uncoupled compressive sensing: Another blessing of massive MIMO,” Online Available: https://arxiv.org/abs/2002.03044.

[12] T. L. Narasimhan, A. Chockalingam, and B. S. Rajan, “Factor graph based joint detection/decoding for LDPC coded large-MIMO systems,” in 2012 IEEE 75th Vehicular Technology Conference (VTC Spring), 2012, pp. 1–5.

[13] D. L. Donoho, A. Maleki, and A. Montanari, “Message-passing algorithms for compressed sensing,” Proceedings of the National Academy of Sciences, vol. 106, no. 45, pp. 18 914–18 919, 2009.