Convolutional Neural Networks for blind decoding in Sparse Code Multiple Access

Imen Abidi, Moez Hizem, Iness Ahriz, Maha Cherif, Ridha Bouallegue

Convolutional Neural Networks for blind decoding in Sparse Code Multiple Access

Imen Abidi[1], Moez Hizem[1], Iness Ahriz[2], Maha Cherif[1], Ridha Bouallegue[1]
[1]Innov'COM, Sup'Com, University of Carthage, Tunis, Tunisia
[2] CEDRIC/ LAETITIA, CNAM, Paris, France
{imen.abidi, moezhizem, maha.cherif}@supcom.tn
iness.ahriz@cnam.fr
ridha.bouallegue@supcom.tn

***Abstract*—Sparse code multiple access (SCMA) has attracted growing research interests in order to meet the targets of the next generation of wireless communication networks. Since it relies on non-orthogonal multiple access (NOMA) techniques, it is considered as a promising candidate for future systems that can improve the spectral efficiency and solve the problem of massive user connections. In this paper, the basic concept of SCMA is introduced, including SCMA encoding, codebook mapping, and SCMA decoding. The major challenge of SCMA is the very high detection complexity. Then, a novel strategy for blind decoding based on convolutional neural networks is proposed. Through simulations, we showed that our proposed scheme outperforms conventional schemes in terms of both BER and computational complexity, where 0.9 dB improvements can be achieved.**

***Index Terms*—SCMA; deep learning; convolutional neural networks; MPA**

I. Introduction

Due to the significant problem of spectrum resources scarcity and dramatic increase of number of users, wireless communication is evolving towards new generations [1]. Future wireless generations should be very flexible to support much higher capacity and more services for more connected mobile users. The next generation is needed to provide maximum throughput, high spectral efficiency, and high data bit rate in large coverage area [2]. These requirements are rather challenging and it is difficult to satisfy it by using orthogonal multiple access (OMA) schemes [3]. Future systems need to put forward a new type of multiple access technology to provide higher spectral efficiency and increase the access capability and capacity of the system. To enhance the multiple access gain, non-orthogonal multiple access schemes (NOMA) have been proposed and emerged as promising candidates solution for future systems [4] [5]. SCMA has been considered as a promising technique for future systems and has attracted growing research interests in order to meet the targets of the next generation of wireless communication networks [6]. SCMA is a waveform in which sparse codewords of multiple layers of devices are overlaid in code and power domains and carried over shared time-frequency resources [7]. SCMA is a technique, in which data streams, to be transmitted, are directly mapped to code words of different multi-dimensional codebooks, where each codeword represents a spread transmission layer [8].

The major challenge of SCMA is the very high detection complexity. In order to reduce the computational complexity [9], some methods have been proposed in recent researches, including the low-density parity check (LDPC) message passing algorithm (LDPC-MPA) detector in [10], the Jacobi approximation message passing algorithm (Max-log MPA) detector in [11], and the low complexity Maximum-Likelihood Approximation detector in [12]. The proposed methods have exhibited a good tradeoff between the bit error rate (BER) performance and the computational complexity [13]. These methods have high computational overheads restricting the real-time operation of SCMA.

In recent years, deep neural networks technologies have been applied to successfully resolve the problems of image recognition and classification [14]. Convolutional Neural Networks are a category of Neural Networks that have proven very effective in such areas [15]. We extend the application of deep learning to SCMA detection and propose a low complexity deep neural networks SCMA blind detector. Recently, parallel works were presented in [16] and [17] investigating SCMA enabled by deep neural network. [16] proposes a SCMA scheme in which the generation of codewords and the detection of symbols are based on deep learning whilst [17] utilizes a fixed constellation mapping and studies only a detector based on neural network. Different from [16], we are only interested on the detection problem and different from [17], a novel convolutional neural networks (CNNs) based decoding for SCMA is proposed, where a blind decoding strategy for SCMA is learned using deep convolutional neural networks, in which the architecture of the network is different. Thanks to its structure, CNNs overcome the limitations of traditional deep neural networks. In fact, it explicitly focus on local structure of the signal and it is able to gain invariance to local deformations due to the noise, which minimize bit error rate (BER). The number of free parameters in CNNs is limited, which make the complexity lower compared to traditional deep learning. The training can be done in a single shot to generate the decoding model, while a MPA based decoder requires several iterations to achieve a high performance. A set of simulations is performed to confirm that the blind CNNs based decoder provides a lower BER and less computational complexity than a conventional MPA based decoder for SCMA. The remainder

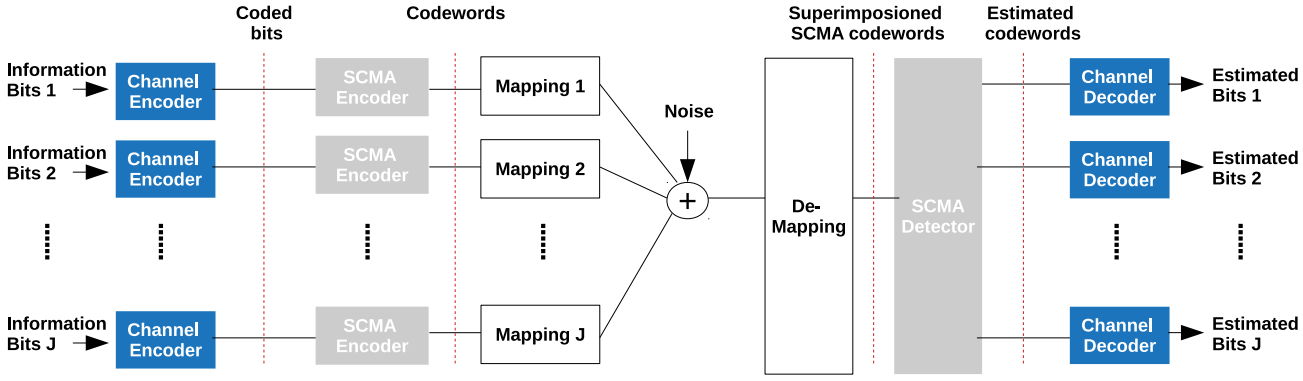


Fig. 1: Block diagram of SCMA.

of this paper is organized as follows. In section I, we introduce the system model of SCMA. The conventional MPA based decoder and the proposed convolutional neural networks based decoder are discussed in Section II. The simulation results and discussion are provided in Section III. We conclude and propose perspectives in Section IV.

II. System Model

We consider a basic system model of SCMA with J users spreading over K resources through additive white Gaussian noise (AWGN) channel. J is greater than K and the ratio J/K is defined to be the overloading factor as shown in Fig. 1. Bit streams to be transmitted are directly mapped to codewords of different multi-dimensional codebooks, where each codeword represents a spread transmission layer. The transmitted signal from each user is multiplexed to some orthogonal resources. An SCMA decoder processes the superimposed signal.

A. SCMA Encoder

An SCMA encoder is defined as a map from $log2(M)$ bits to a K-dimensional complex codebook of size M. The SCMA codebooks are designed in order to reduce the detection complexity. The codewords are composed of multi-dimensional complex symbols, and the codewords in the same codebook have the same sparse pattern.

Fig. 2 shows an example of a codebook set containing 6 codebooks. Each codebook transmits only one layer, which is composed of 4 multi-dimensional complex codewords that corresponds to 4 points of constellation. The length of each codeword is $M = 4$, which is the same as the spreading length. For mapping, the selection of each codeword from codebook is based on the input bit sequence.

The SCMA encoder can be described as a mapping of coded bits b_j to a multidimensional complex codeword x_j. Then, the data are modulated to the subcarriers. The received signal can be expressed as bellow:

$$\mathrm{y} = \sum_{j=1}^{J} diag(h_j)\mathrm{x}_j + n \tag{1}$$

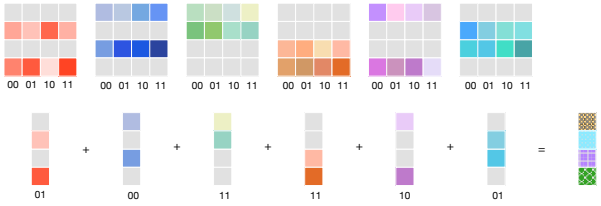


Fig. 2: SCMA Mapping.

where $\mathrm{x}_j = (x_{1j}, .., x_{Kj})^T$ is the SCMA codeword of layer j, and $\mathrm{h}_j = (h_{1j}, .., h_{Kj})$ is the channel vector of layer j, and n is the ambient noise.

B. SCMA Decoder

This section proposes a new low-complexity blind-detection method based on CNNs for SCMA which outperforms MPA based decoding and reduces the complexity. Subsection $B.1)$ briefly describes the original MPA decoder. The proposed detection method is discussed in subsection $B.2)$.

1) Message Passing Algorithm: At the receiver, maximum a posterior probability (MAP) detection is optimal to decode the overlaid symbols over each resource but with very large complexity [18]. There are already some optimizations on the detection [19], but the complexity is still exponential. The estimation of transmitted codewords $X = (x_1, x_1, ..., x_J)$ is resolved according to:

$$\widetilde{X} = \underset{X \in (X_{j=1}^{J})\{x_j\}}{\arg\max} P(X|Y) \tag{2}$$

where x_j is the codebook of user j.
By utilizing SCMA codewords sparsity, an iterative multi-user detection based on MPA is employed to achieve a near optimal performance with low complexity. The task of MPA detector is to separate the interfering symbols in an iterative fashion. It is based on the message exchange between layer nodes VN and resource nodes FN in the factor graph. MPA algorithm updates the extrinsic information of FN and VN nodes along

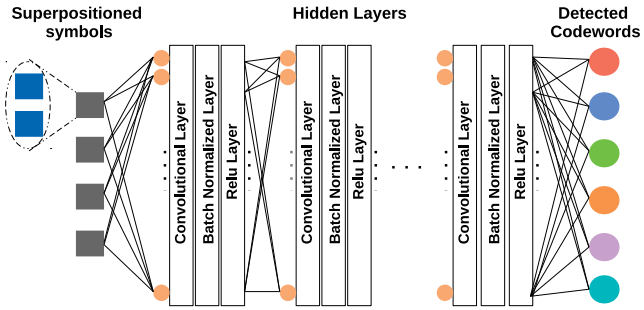


Fig. 3: Structure of CNNs based decoder for SCMA.

the edges in the factor graph, which is defined as low-density matrix F. F is defined as $F = (f_1, f_2, ..., f_J)$, in which $f_j = (f_{j,1}, f_{j,2}, ..., f_{j,K})^T$ is the indicator vector of the j^{th} layer and $f_{j,k} = 1$ means that the layer j is connected to the resource k, otherwise, $f_{j,k} = 0$. The following matrix illustrates an example of factor graph, which is presented as 4 x 6 matrix given by:

$$F = \begin{bmatrix} 0 & 1 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 & 1 & 0 \end{bmatrix} \quad (3)$$

The codewords of each codebook has a probability to be the transmitted symbol. Firstly, the probability distribution of codewords are initialized to be equal. Then, they are calculated as the exchanged messages in MPA according to the received signal y, the knowledge of the channel h_j, the noise power N_0 and the factor graph.
In each iteration, the probability distribution of each sent codeword on resource k is deduced by MPA detector according the rest of the connected user nodes. Considering the equation (2), the value of the probability of x_{12} is computed according to x_{13}, x_{15}, the noise power N_0 and the value of y_1. The procedure is detailed in Algorithm 1.

2) Convolutional Neural Networks: In this section, we briefly introduce a blind detector for SCMA based on convolutional neural networks [20]. CNNs use a variation of multilayer perceptron designed to require minimal preprocessing. These deep-learning networks perform automatic feature extraction without human intervention, unlike most traditional machine-learning algorithms [21]. Deep learning's ability to process and learn from huge quantities of unlabelled data, give it a distinct advantage over previous algorithms. In deep-learning networks, each layer of nodes trains on a distinct set of features based on the previous layer's output [14]. Thanks to its structure, CNNs can perfectly overcome the limitations of traditional neural networks. A CNN consists of number of layers such as convolutional, non linearity and fully connected layers. Convolutional layers take several feature maps as input and produce n feature maps as output through the calculation

Algorithm 1: MPA detection for SCMA

Input: $y, F, N_0, h_{j,k}, j = 1..J, k = 1..K, X_j$;
Step1: Initialization
for *k = 1..K* **do**
  $ind = find(F(k,:) == 1)$ //find users connected to k;
  for $m_1 = 1:M \& m_2 = 1:M \& m_3 = 1:M$ **do**
    $f(m_1, m_2, m_3, k, N_{0,k}, h_k) =$
    $-(1/N_0)|y_k - (X(k, m_1, ind(1))h_{k,ind(1)} +$
    $X(k, m_2, ind(2))h_{k,ind(2)}) +$
    $X(k, m_3, ind(3))h_{k,ind(3)})|^2$;
    $I^0_{j\to k} = 1/M, j = 1..J, k = 1..K$;
  end
end
Step2: Iterative message passing along edges
for *it = 1..Niter* **do**
  {Resource node k passes updates obtained from extrinsic information to its neighboring layer nodes }
  for *k = 1..K & F(j,k) = 1* **do**
    $ind = find(F(k,:) == 1)$;
    $v_1 = ind(1)$,$v_2 = ind(2)$, $v_3 = ind(3)$;
    $I^{it}_{k\to v_1}(m_1) =$
    $\sum_{m_2=1}^{M}\sum_{m_3=1}^{M} f(m_1, m_2, m_3, k, N_{0,k}, h_k)$;
    $(I^{it-1}_{v_2\to k}(m_2)(I^{it-1}_{v_3\to k}(m_3), m_1 = 1..M$;
    $I^{it}_{k\to v_2}(m_2) =$
    $\sum_{m_1=1}^{M}\sum_{m_3=1}^{M} f(m_1, m_2, m_3, k, N_{0,k}, h_k)$;
    $(I^{it-1}_{v_1\to k}(m_1)(I^{it-1}_{v_3\to k}(m_3), m_2 = 1..M$;
    $I^{it}_{k\to v_3}(m_3) =$
    $\sum_{m_1=1}^{M}\sum_{m_3=1}^{M} f(m_1, m_2, m_3, k, N_{0,k}, h_k)$;
    $(I^{it-1}_{v_1\to k}(m_1)(I^{it-1}_{v_2\to k}(m_2), m_3 = 1..M$;
  end
end
{**Update messages from layer nodes**}
{ Layer node j passes updates obtained from extrinsic information to its neighboring resource nodes }
for *j = 1..J* **do**
  for *m = 1..M* **do**
    $I^{it}_{j\to k_1} = normalize(ap_j(m)I_{k_2\to j}(m))$;
    $I^{it}_{j\to k_2} = normalize(ap_j(m)I_{k_1\to j}(m))$;
    Step3: Output the final guess of the sent codeword by each layer
    The guess at layer node j for codeword m is a chain product of all guesses from all its connected resource nodes and the a prior probability
    $Q_j(m) = ap_j \prod_{l=1}^{d_v} I^{Niter}_{l\to j}, m = 1..M$;
    d_v is the number of connected resources to a user node.
  end
end
Finally, the codeword, which maximizes $Q_j(.)$ is considered as the transmitted codeword in the j^{th} user.

of the scalar product between their weights and the region connected to the input volume, where n is the number of filters in the convolution layer. The output of layer l consists of $m_1^{(l)}$ feature maps of size $m_2^{(l)}$ x $m_3^{(l)}$. The i^{th}feature map in layer l, denoted $Y_i^{(l)}$, is computed as

$$Y_i^{(l)} = B_i^{(l)} + \sum_{j=1}^{m_1^{(l-1)}} K_{i,j}^{(l)} * Y_j^{(l-1)} \quad (4)$$

where $B_i^{(l)}$ is a bias matrix and $K_{i,j}^{(l)}$ is the filter connecting the j^{th} feature map in layer $(l-1)$ with the i^{th} feature map in layer l.

If layer l is a non-linearity layer, its input is given by $m_1^{(l)}$ feature maps and its output comprises again $m_1^{(l)} = m_1^{(l-1)}$ feature maps, each of size $m_2^{(l-1)} \times m_3^{(l-1)}$, given by

$$Y_i^{(l)} = f(Y_i^{(l-1)}) \tag{5}$$

where f is the activation function used in layer l and operates point wise. A fully connected (FC) layer with d inputs and n nodes have connections to all activations in the previous layer. It can be interpreted as a convolutional layer with an input of shape $1 \times 1 \times d$ and n filters of size 1×1. This will produce an output shape $1 \times 1 \times n$. FC acts as classifier.

Our goal in using CNNs in investigating SCMA is to construct a blind decoding strategy with high performance and low complexity. The act of transmitting data streams through noisy channels make the decoding technique more difficult, and the job of the convolutional neural network is to recognise the codewords sent by each user within the noise, without prior knowledge of channel information or the relationships between resource and layer nodes presented by the factor graph. When a network is being trained, it generates a model, and measures the distance between that model and the benchmark through a loss function. The loss function can be defined in many different ways but a common one is mean squared error (MSE), which can be defined as:

$$L = \sum_i (T_i - O_i)^2 \tag{6}$$

where T_i is the actual data and O_i is the predicted results. The attempts of CNNs to minimize the loss function involve resampling the shuffled inputs and re-reconstructing the data, until it finds those inputs, which bring its model closest to achieve the designed targets. A backprpagation method is used to calculate the gradients of the loss with respect to all weights in the network. Thus, in order to minimize the output error, the weights, biases of CNNs and filter values are updated according to channel samples using a stochastic gradient descent (SGD) method. The weight update is performed based on the following equation:

$$w = w_i - \alpha \frac{dL}{dw} \tag{7}$$

where α is the learning rate.

In order to construct the model, it is required to define the number of layers, the learning weights, number and size of filters, along with other tunable parameters. These parameters and the architecture of the network are fixed initially and do not change during the training process. Only the values of the filter matrix and connection weights get updated during training.

We have proposed a blind SCMA decoder based on deep neural networks. Initially, We have structured the data for training and testing sinuously. Then, we have architected the layers and the filters in order to generate the adequate model, which will be used to detect blindly the codewords sent by each user, without prior knowledge of signal characteristics or channel information.

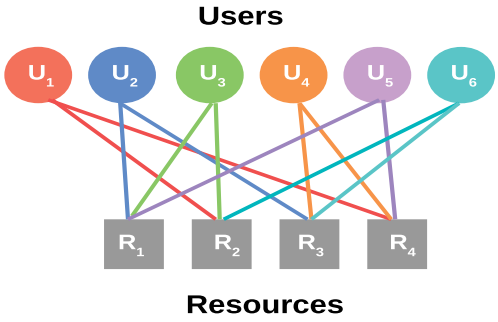


Fig. 4: Factor graph representation of an SCMA with $K = 4$ and $J = 6$.

In our proposed decoding scheme, the superimposed input data streams on each resource are treated as real and imaginary data in order to construct the adequate inputs for the convolutional layer as shown in Fig. 3. The input data streams used for training are generated randomly. Then, they are propagated through the batch-normalized layers and Rectified layers to reach the final layer, which can be a logistic, or softmax, classifier that assigns a likelihood to a particular outcome or label. We utilized the fully connected (FC) layer for the final layer in order to combine all information that could spread over all the resources, given that multiple data streams are multiplexed over multiple resources.

During training, all the weights and parameters of the network have been optimized to correctly classify the input data from the training set. Once the training process is finished, our model is generated and our network is able to learn to recognize the sent codewords. When a new data is input into the CNN, the network goes through the forward propagation step and outputs a probability for each class. If our training set is large enough, the network will generalize well to new inputs and classify them into correct categories.

III. SIMULATION RESULTS

A. Data set

The SCMA design follows the factor graph of Fig. 4 with the basic parameters $K = 4$ resources, $J = 6$ users and $d_f = 3$, where d_f is the number of branches arriving to a resource node. Hence, the overloading factor is up to 150%. A high ratio of user's stream to resource degrades BER performance because more bits are super-positioned to single resource. The simulations are performed over a fixed constellation mapping as depicted in Fig. 5. The data are trained over E_b/N_0 of values: $2, 4, 6, 8, 10, 12$ and 14. Then, we conclude that $E_b/N_0 = 6$ is the best training environment for a high SNR. For training, $300,000$ values of input data streams are generated randomly. Thus, our network is large enough to generalize well. In addition, we utilized 4 sets of convolutional layer, batch normalization layer and ReLU layer. For the final layer, a full-connected layer is utilized. The weights and biases are initialized randomly. Yet, the

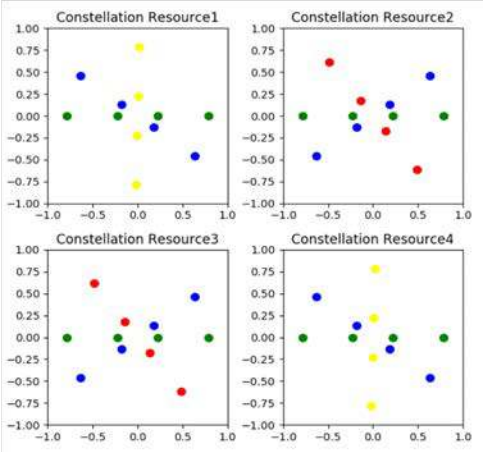


Fig. 5: Constellations of 6 users on 4 resources

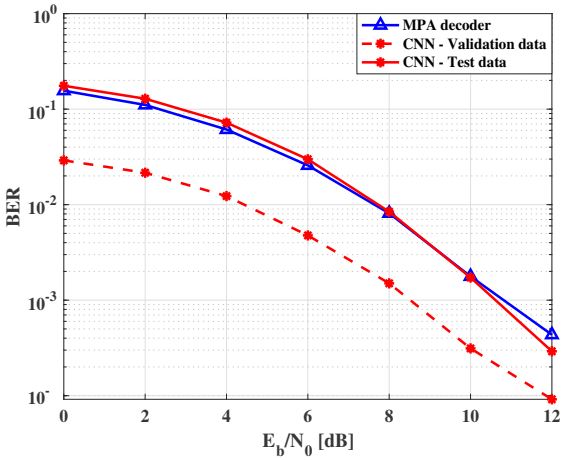


Fig. 6: BER performance of MPA based decoder vs blind CNN detector for SCMA

learning rate of SGD method, α, is set to 0.001, which is sufficiently small for convergence of loss function. It should be noted that the convolutional neural network parameters, e.g., learning rate, number of hidden layers and the number of nodes, are not arbitrary chosen but determined through exhaustive searches.

Through training, we generate our model and in order to investigate its efficiency, data set input streams are utilized for validation.

B. Results

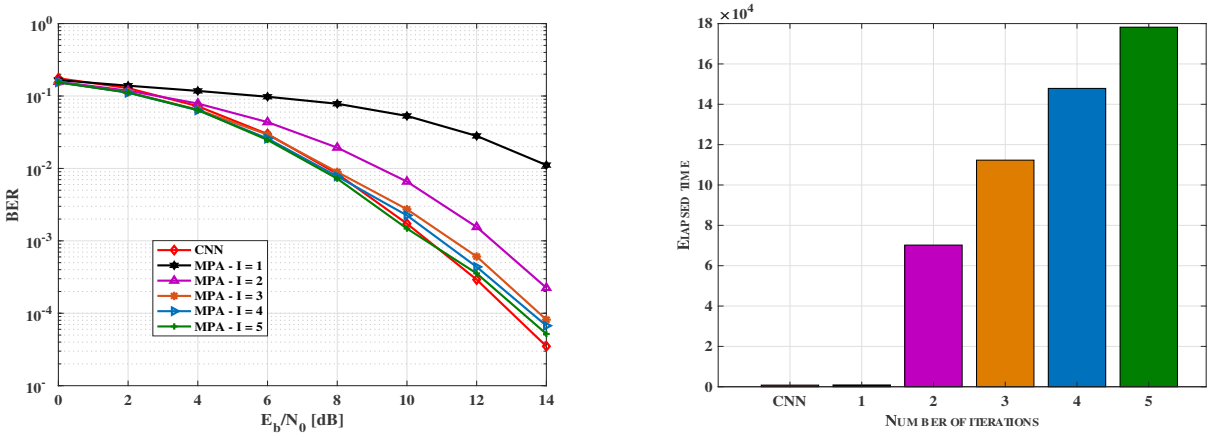
Numerical results show that the proposed CNNs for blind decoding in SCMA can achieve an excellent performance gain as depicted in Fig. 6. It can be seen that the BER performance of blind CNNs-SCMA are better than MPA based decoder for SCMA in additive white Gaussian noise channels, where $0.9dB$ improvement can be achieved. It outperforms MPA based decoder for SCMA using validation data in terms of BER. The deep learning enables an efficient blind decoding strategy for a sparse and multidimensional superimposed signal.

In Fig. 6, we evaluate the BER for CNNs based decoding for SCMA using data set values for testing. The simulation results show that blind CNNs-SCMA exhibits the same BER performance as MPA under the value of $E_b/N_0 = 6$. Then it outperforms conventional MPA decoder without prior knowledge about channel information and user-resource connections.

C. Performance vs Complexity

Through simulations, we find that the computational complexity of CNNs for blind decoding in SCMA is much less than MPA based decoder scheme. The main contribution of deep learning is to train the network once, and for a new input data, the output probabilities are calculated using the weights which have been optimized to correctly classify all the previous examples treated during training. Yet, by using MPA, much iteration is needed to achieve a good performance, as shown in Fig. 7a. In CNNs for blind detection in SCMA, all the steps of optimizing the weights and tuning the network parameters are performed during the training process. Once the training is finished and the model is generated, our network is able to detect quickly the sent codewords in low as well high SNR environment. While, by using MPA based SCMA detector, the detection of sent codewords vary according to the value of SNR. Thus, the steps of decoding of new data streams get repeated at each value of E_b/N_0 and during the iterative process.

We denote the number of iterations by I. The complexity of MPA receiver is to the order of $I.(d_f^2.M^{d_f} + d_f^2.JM/K)$, where M is the modulation order and d_f is the maximum degree of resource node. Thus, it is clear that the complexity of MPA based decoder increases linearly with the system parameters, namely the number of iterations and the codebook dimension. Whereas, our proposed scheme is a blind detection, which does not take into consideration the degree of resource node and user-user connection. For both methods, our proposed scheme and MPA based detector, the simulations are performed under the same conditions. The simulations are implemented by utilizing $2.5GHz$ Intel Core i5 CPU with $8Go$, $1600MHz$ double data rate three (DDR3) memory. The obtained results showed that the computation time of our proposed scheme is lower than the MPA scheme even with one iteration as shown in Fig. 7b. However, the BER performance is much better as confirmed in Fig. 7a. In consequence, CNNs for blind decoding technique outperforms conventional schemes in terms of both BER and computational complexity. For a new data set of symbols, the output probabilities are calculated using the weights which have been optimized to correctly classify all the previous training examples.



(a) BER performance

(b) Runtime

Fig. 7: BER Performance of MPA based decoder vs CNNs based decoder for SCMA.

IV. Conclusion

In this paper, we have proposed a novel strategy of decoding for SCMA based on convolutional neural networks. The simulation results provided illustrate the performance advantage of decoding strategy based on convolution neural networks over the one based on the iterative approach MPA. Through simulations, we showed that our proposed scheme outperforms conventional schemes in terms of both BER and computational complexity. Coupled with the rapid advancements in distributed and GPU (graphics processing units) computation, it is now possible to train much larger and more powerful CNNs that achieve better performance in lower time. Future research will be directed towards further improvements on both transceiver and receiver. We aim to implement a flexible SCMA encoder an decoder that accept a codebook allocation and a mother constellation as input and utilize the deep learning advantage in designing the SCMA codebooks.

References

[1] Jaap Van De Beek and Branislav M Popovic. Multiple access with low-density signatures. In *Global Telecommunications Conference, 2009. GLOBECOM 2009. IEEE*, pages 1–6. IEEE, 2009.

[2] Zhiguo Ding, Yuanwei Liu, Jinho Choi, Qi Sun, Maged Elkashlan, I Chih-Lin, and H Vincent Poor. Application of non-orthogonal multiple access in lte and 5g networks. *IEEE Communications Magazine*, 55(2):185–191, 2017.

[3] Linglong Dai, Bichai Wang, Yifei Yuan, Shuangfeng Han, I Chih-Lin, and Zhaocheng Wang. Non-orthogonal multiple access for 5g: solutions, challenges, opportunities, and future research trends. *IEEE Communications Magazine*, 53(9):74–81, 2015.

[4] Bichai Wang, Kun Wang, Zhaohua Lu, Tian Xie, and Jinguo Quan. Comparison study of non-orthogonal multiple access schemes for 5g. In *Broadband Multimedia Systems and Broadcasting (BMSB), 2015 IEEE International Symposium on*, pages 1–5. IEEE, 2015.

[5] Yuya Saito, Yoshihisa Kishiyama, Anass Benjebbour, Takehiro Nakamura, Anxin Li, and Kenichi Higuchi. Non-orthogonal multiple access (noma) for cellular future radio access. In *Vehicular Technology Conference (VTC Spring), 2013 IEEE 77th*, pages 1–5. IEEE, 2013.

[6] Hosein Nikopour and Hadi Baligh. Sparse code multiple access. In *Personal Indoor and Mobile Radio Communications (PIMRC), 2013 IEEE 24th International Symposium on*, pages 332–336. IEEE, 2013.

[7] Hosein Nikopour, Eric Yi, Alireza Bayesteh, Kelvin Au, Mark Hawryluck, Hadi Baligh, and Jianglei Ma. Scma for downlink multiple access of 5g wireless networks. In *Global Communications Conference (GLOBECOM), 2014 IEEE*, pages 3940–3945. IEEE, 2014.

[8] Yiqun Wu, Chao Wang, Yan Chen, and Alireza Bayesteh. Sparse code multiple access for 5g radio transmission. In *Vehicular Technology Conference (VTC-Fall), 2017 IEEE 86th*, pages 1–6. IEEE, 2017.

[9] Hosein Nikopour and Mohammadhadi Baligh. Systems and methods for sparse code multiple access, January 19 2016. US Patent 9,240,853.

[10] Baicen Xiao, Kexin Xiao, Shutian Zhang, Zhiyong Chen, Bin Xia, and Hui Liu. Iterative detection and decoding for scma systems with ldpc codes. In *Wireless Communications & Signal Processing (WCSP), 2015 International Conference on*, pages 1–5. IEEE, 2015.

[11] Jiaqi Liu, Gang Wu, Shaoqian Li, and Olav Tirkkonen. On fixed-point implementation of log-mpa for scma signals. *IEEE Wireless Communications Letters*, 5(3):324–327, 2016.

[12] Roya Alizadeh and Yvon Savaria. Performance analysis of a reduced complexity scma decoder exploiting a low-complexity maximum-likelihood approximation. In *Electronics, Circuits and Systems (ICECS), 2016 IEEE International Conference on*, pages 253–256. IEEE, 2016.

[13] Mehmood Alam and Qi Zhang. Performance study of scma codebook design. In *Wireless Communications and Networking Conference (WCNC), 2017 IEEE*, pages 1–5. IEEE, 2017.

[14] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[15] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *Proceedings of the IEEE international conference on computer vision*, pages 1026–1034, 2015.

[16] Minhoe Kim, Nam-I Kim, Woongsup Lee, and Dong-Ho Cho. Deep learning-aided scma. *IEEE Communications Letters*, 22(4):720–723, 2018.

[17] Chao Lu, Wei Xu, Hong Shen, Hua Zhang, and Xiaohu You. An enhanced scma detector enabled by deep neural network. *arXiv preprint arXiv:1808.08015*, 2018.

[18] Frank R Kschischang, Brendan J Frey, and H-A Loeliger. Factor graphs and the sum-product algorithm. *IEEE Transactions on information theory*, 47(2):498–519, 2001.

[19] Kexin Xiao, Baicen Xiao, Shutian Zhang, Zhiyong Chen, and Bin Xia. Simplified multiuser detection for scma with sum-product algorithm. In *Wireless Communications & Signal Processing (WCSP), 2015 International Conference on*, pages 1–5. IEEE, 2015.

[20] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

[21] Steve Lawrence, C Lee Giles, Ah Chung Tsoi, and Andrew D Back. Face recognition: A convolutional neural-network approach. *IEEE transactions on neural networks*, 8(1):98–113, 1997.