M.S. THESIS

A Study on Sparse Spreading Multiple Access for Massive Internet of Things (IoT) Communications

대용량 IoT 네트워크를 위한
희소 확산 다중접속기법 연구

BY

SEO HEE-JIN

FEBRUARY 2017

DEPARTMENT OF ELECTRICAL ENGINEERING AND
COMPUTER SCIENCE
COLLEGE OF ENGINEERING
SEOUL NATIONAL UNIVERSITY

M.S. THESIS

A Study on Sparse Spreading Multiple Access for Massive Internet of Things (IoT) Communications

대용량 IoT 네트워크를 위한
희소 확산 다중접속기법 연구

BY

SEO HEE-JIN

FEBRUARY 2017

DEPARTMENT OF ELECTRICAL ENGINEERING AND
COMPUTER SCIENCE
COLLEGE OF ENGINEERING
SEOUL NATIONAL UNIVERSITY

A Study on Sparse Spreading Multiple Access for Massive Internet of Things (IoT) Communications

대용량 IoT 네트워크를 위한
희소 확산 다중접속기법 연구

지도교수 심 병 효
이 논문을 공학석사 학위논문으로 제출함

2016년 12월

서울대학교 대학원

전기 컴퓨터 공학부

서 희 진

서희진의 공학석사 학위 논문을 인준함

2016년 12월

위 원 장:	김 남 수	(인)
부위원장:	심 병 효	(인)
위 원:	최 성 현	(인)

Abstract

This paper addresses a problem of massive connectivity with millions of Internet of Things (IoT) devices. Massive connectivity is one of the most important requirements for the next generation of networks. In this paper, we propose sparse spreading multiple access which overloads a large number of users on limited sizes of resources as a solution for massive connectivity. To do this, first, a single low density signature (LDS) codebook is used for accessing the medium to transmit data and a pilot signal. The number and length of the signature is designed to maximize the access rate. Second, a compressed sensing method is utilized for estimating user activity and channel impulse response vector from overloaded pilot signals. To maximize successive detection rates, we organize a particular spreading signature by concatenating multiple codewords from the LDS codebook and utilize the active user information and estimated channel information jointly. Third, an adaptive message passing algorithm (MPA) is applied to minimize inter-code interference between users who use the same sparse code casually.

keywords: IoT, Massive connectivity, Low density signature, Active User Detection, Non orthogonal multiple access

student number: 2015-20936

Contents

List of Tables

List of Figures

Chapter 1

Introduction

Internet of things (IoT) is a new paradigm to support autonomous data transfer among machines. Applications of IoT include smart metering, healthcare, autonomous driving, factory automation, and many more. Density of machines in IoT networks is in general much higher than that of human mobile phone, and hence it is of great importance to support seamless connection and quality of service (QoS) in the IoT environment where the device density is high. In the traditional wireless systems, such as 4G OFDMA cellular systems, number of users being served is proportional to the number of (time/frequency) resources. However, in the IoT regime where the number of machine devices is at least order of magnitude higher than the number of human communication devices, conventional multiple access strategy, so called *orthogonal multiple access* (OMA), cannot be an appropriate option. Furthermore, scheduling numerous IoT devices needs much overhead and will not properly work with IoT devices with limited energy resources. In recent years, many approaches, collectively called *non-orthogonal multiple access* (NOMA) techniques, have been proposed to deal with the imbalance between the communication resources and machine-type devices. Examples of NOMA include interleave-division multiple-access (IDMA), low density signature (LDS) [4, 5], and sparse code multiple access (SCMA) [7–9].

One representative non-orthogonal access technique to support the connectionless

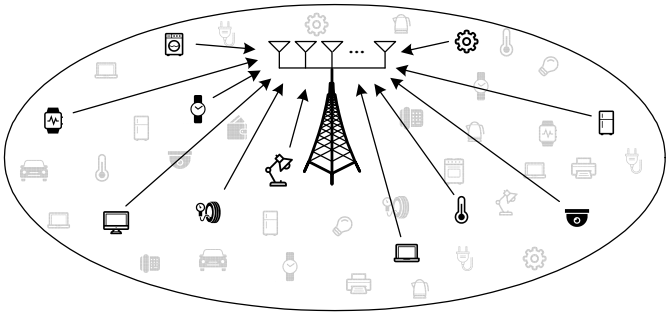

Figure 1.1: Illustration of multiple access in IoT networks.

multiple access is the low density spreading (LDS) scheme. In the LDS scheme, each data channel spreads its data signal over small number of resources [4, 5]. The interference caused by the multiple users can be controlled by using sparsity of spreading codes. In other words, by setting some elements in the codeword to zero, we can control the interference caused by other codewords. For example, if we set two elements as zeros in a 4-length walsh code (e.g. $[+1, +1, 0, 0]^T$, $[0, 0, +1, -1]^T$, $[+1, 0, -1, 0]^T$, and $[0, -1, 0, +1]^T$), the interference from other codes can be reduced by half of the original code. However, in this case, the orthogonality of the spreading code is not guaranteed.

Recently, an approach referred to as the sparse code multiple access (SCMA) have been proposed for connectionless data transmission using LDS [7–9]. In order to support uplink multiple access, the SCMA employs active user detection (AUD) and a message passing algorithm (MPA). Firstly, the active user information, estimated by AUD, is delivered to MPA. Next, MPA decodes the data signal using the active user information and generates a prior probability which is delivered to AUD, for next iteration. While SCMA is interesting, there are weaknesses, not thoroughly addressed in the previous efforts, to maintaining massive connectivity for IoT network. First, the codebook design is difficult and computational burden is high. Since SCMA relies on user-specific codebook, codebook design complexity increases exponentially as the

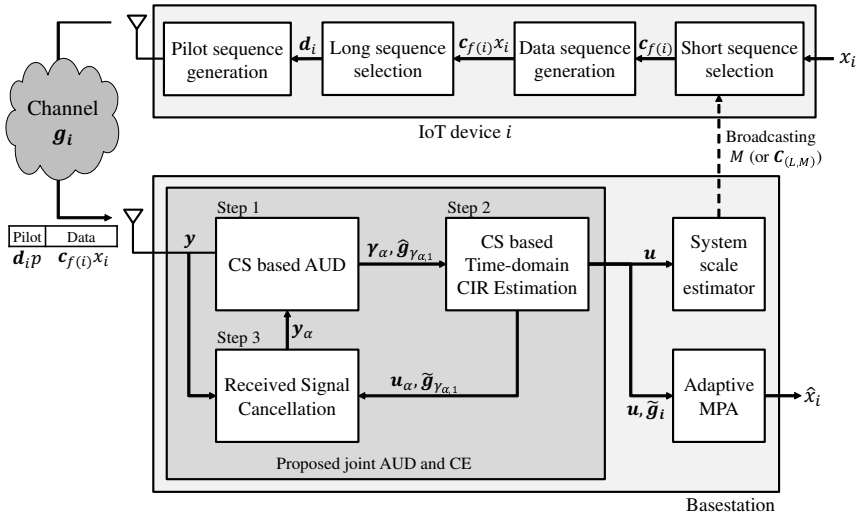

Figure 1.2: Block diagram of the SSMA technique

number of users increases. Second, the effect of code collision that multiple users choose same codebook is not considered. Since the MPA distinguishes the active users using the codebook, the code collision brings the performance degradation severely. To avoid the code collision, each user should know the codebook information of other users, which is generally impossible in the IoT network.

In this paper, we propose a novel non-orthogonal multiple access schemes, referred to as *sparse spreading multiple access* (SSMA), suitable for the grant-free IoT networks. We design an overloaded system using single LDS codebook, in which the number of codewords is much smaller than the number of devices. For example, 1000 devices share 200 codewords with size being 100, so the system can be modeled as 100 by 1000 under-determined system. In our approach, a codeword of an active user is chosen at random from the single LDS codebook and transmits pilot and data signals over spreading sequences. In order to recover the data from the overloaded (under-determined) signals, we use the compressed sensing (CS) based joint AUD and time-domain sparse channel estimation (CE). By exchanging active user information and channel impulse response between AUD and CE iteratively, the proposed SSMA tech-

nique achieves significant improvement in performance over the conventional AUD method. In order to decode the data signal, we also employ the adaptive MPA which reorganize the codebook using only actual codeword of each user. Numerical results demonstrate that the proposed technique achieves significant increasing of simultaneously supportable devices over the conventional orthogonal techniques under the same size of data packet target.

Chapter 2

Sparse Spreading Multiple Access

2.1 Uplink Multiple Access System

We briefly describe IoT uplink multiple access scenario in which the basestation with one or more antennas receives information from multiple devices with a single antenna. In the IoT system, some devices are sending information (we call these devices as active devices) and the others are in idle state (see Fig.1.1).. Suppose N devices spread their own symbols using L resources, then the received vector $\mathbf{y}$ corresponding to one symbol period over L resources at the basestation is

$$\mathbf{y} \quad = \quad \sum_{i=1}^{N} \mathrm{diag}(\mathbf{g}_i)\boldsymbol{\phi}_i x_i + \mathbf{v}, \tag{2.1}$$

where x_i is the data symbol, $\mathbf{g}_i = [g_{i,1} \dots g_{i,L}]$ is the channel vector between the i-th device and the basestation, $\boldsymbol{\phi}_i = [\phi_{i,1}, \dots, \phi_{i,L}]^T$ is the codeword vector of the i-th device, and $\mathbf{v}$ is the complex Gaussian noise vector ($\mathbf{v} \sim \mathcal{CN}(0, \sigma^2\mathbf{I})$). Note that the entries of x_i are chosen from a finite modulation set $\mathbb{Q}$ when the device i is active and zero otherwise.

2.2 Sparse Spreading Multiple Access

Fig. 1.2 depicts the block diagram of the proposed SSMA technique. The basestation broadcasts the number of LDS codewords M predicted based on the number of active users and the system requirements (such as latency or throughput). Using M, each device generates LDS codebook $\mathbf{C}_{(L,M)}$ and selects its own codeword $\mathbf{c}_{f(i)}$.

In order to overload the data, that is, to transmit more information symbols than the number of resources, the data signals are spread by a short codeword $\mathbf{c}_{f(i)}$. For the pilot transmission, on the other hand, long LDS codeword $\mathbf{d}_i$, generated from short codewords, is employed to perform 1) AUD and 2) reliable channel estimation of active users. By using short and long LDS codeword, SSMA has following advantages over existing other sparse code based method. First, the device can easily generate codebook without complicated process [5]. Second, the codebook can be easily expanded, thereby achieving user scalability. Third, the accuracy of active user detection and the quality of the channel estimation are improved at the basestation. At the basestation side, since the pilot signal of all devices in a cell can be modeled as sparse signal due to the inactive devices, the active user information $\mathbf{u}$ as well as channel $\tilde{\mathbf{g}}_i$ of active users can be estimated using the compressed sensing algorithm. After AUD and channel estimation, the basestation performs data decoding using the adaptive MPA and broadcasts the number of LDS codewords M to all devices in the cell for next transmission.

2.3 Pilot and Data Transmission Process

In this subsection, we describe the pilot and data transmission strategy in SSMA systems. In the conventional system such as OFDMA, since data transmission is performed with orthogonal code, the amount of data being transmitted is limited by the number of available resources (e.g. resource element in LTE-A system).

However, in the proposed scheme, we employ LDS code which is one of non-

orthogonal code to serve more users than orthogonal code can allow at the expense of the orthogonality.

The LDS codebook $\mathbf{C}_{(L,M)}$ is designed to have a set of codes with L-length and M codes (i.e., $\mathbf{C}_{(L,M)} = [\mathbf{c}_1, \cdots, \mathbf{c}_M] \in \mathbb{C}^{L \times M}$, $\mathbf{c}_i = [c_{i,1}, \cdots, c_{i,L}]^T \in \mathbb{C}^L$, and $M > L$). For example, a codebook $\mathbf{C}_{(6,9)}$ when 9 devices are transmitting information using 6 resources is given by

$$\mathbf{C}_{(6,9)} = \begin{bmatrix} w_0 & 0 & 0 & 0 & 0 & w_1 & w_2 & 0 & 0 \\ 0 & 0 & w_0 & 0 & 0 & w_1 & 0 & 0 & w_2 \\ w_0 & w_1 & 0 & 0 & w_2 & 0 & 0 & 0 & 0 \\ 0 & w_0 & 0 & w_1 & 0 & 0 & 0 & w_2 & 0 \\ 0 & 0 & w_0 & w_1 & 0 & 0 & w_2 & 0 & 0 \\ 0 & 0 & 0 & 0 & w_0 & 0 & 0 & w_1 & w_2 \end{bmatrix}, \tag{2.2}$$

where w_j is the non-zero element of the codeword. Note that w_j is designed to meet the *unique decodability* (i.e., $\forall \mathbf{x}_1, \mathbf{x}_2 \in \mathbb{Q}^M : \mathbf{x}_1 \neq \mathbf{x}_2 \;\; \Rightarrow \;\; \mathbf{C}_{(L,M)}\mathbf{x}_1 \neq \mathbf{C}_{(L,M)}\mathbf{x}_2$), which is determined by modulation order and the overloading factor [5]. Note also that the unique decodability guarantees that the distance spectrum of $\mathbf{C}_{(L,M)}\mathbf{x}$ does not contain zero and also the minimum Euclidean distance between received vectors must be large [6].

For the pilot transmission, a extended codebook $\mathbf{D}_{(KL,N)}$ is generated for N users using $\mathbf{C}_{(L,M)}$. We design long sparse codeword $\mathbf{d}_i$ with length KL by concatenating K codewords selected from $\mathbf{C}_{(L,M)}$ as the pilot codebook. The long sparse codebook $\mathbf{D}_{(KL,N)}$ is given by

$$\mathbf{D}_{(KL,N)} = \begin{bmatrix} \mathbf{C}^{(1)}_{(L,M)} & \cdots & \mathbf{C}^{(j)}_{(L,M)} & \cdots & \mathbf{C}^{(M^{K-1})}_{(L,M)} \\ \mathbf{C}^{(1)}_{(L,M)} & \cdots & \mathbf{C}^{(\lceil j/M \rceil)}_{(L,M)} & \cdots & \mathbf{C}^{(M^{K-2})}_{(L,M)} \\ \mathbf{C}^{(1)}_{(L,M)} & \cdots & \mathbf{C}^{(\lceil j/M^2 \rceil)}_{(L,M)} & \cdots & \mathbf{C}^{(M^{K-3})}_{(L,M)} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \mathbf{C}^{(1)}_{(L,M)} & \cdots & \mathbf{C}^{(1)}_{(L,M)} & \cdots & \mathbf{C}^{(1)}_{(L,M)} \end{bmatrix}_{(1:N)}, \tag{2.3}$$

where $\mathbf{C}_{(L,M)}^{(j)}$ is the matrix that all columns are shifted to the right for j times from $\mathbf{C}_{(L,M)}$ and $\mathbf{A}_{(1:N)}$ is the submatrix of $\mathbf{A}$ that contains columns indexed by from 1 to N. For example, $\mathbf{C}_{(6,9)}^{(2)}$ is the matrix whose columns are shifted to the right twice from (2.2) as

$$\mathbf{C}_{(6,9)}^{(2)} = \begin{bmatrix} 0 & 0 & w_0 & 0 & 0 & 0 & 0 & w_1 & w_2 \\ 0 & w_2 & 0 & 0 & w_0 & 0 & 0 & w_1 & 0 \\ 0 & 0 & w_0 & w_1 & 0 & 0 & w_2 & 0 & 0 \\ w_2 & 0 & 0 & w_0 & 0 & w_1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & w_0 & w_1 & 0 & 0 & w_2 \\ w_1 & w_2 & 0 & 0 & 0 & 0 & w_0 & 0 & 0 \end{bmatrix}. \tag{2.4}$$

As a simple codebook example, for the case that N is 20 and K is 2, $\mathbf{D}_{(12,20)}$ becomes (2.5) [see (2.5) shown at the bottom of the page].

The purpose of long spreading codeword is to distinguish the active users by pilot signal. Unlike a downlink system, the basestation should know that who transmits the data or not in uplink system. To detect active users, we allocate user-specific long codewords (KL-length) to all users by utilizing a single codebook $\mathbf{C}_{(L,M)}$. There are several advantages by using pilot codebook design as follows. First, the user scalability is achieved without undue complexity compared to the conventional multiple access scheme such as sparse code multiple access (SCMA). Even if the number of active devices are changed, the codeword for new active users can be easily generated us-

$$\mathbf{D}_{(12,20)} = \begin{bmatrix} 0 & w_0 & 0 & 0 & 0 & 0 & w_1 & w_2 & 0 & 0 & 0 & w_0 & 0 & 0 & 0 & 0 & w_1 & w_2 & w_2 & 0 \\ w_2 & 0 & 0 & w_0 & 0 & 0 & w_1 & 0 & 0 & 0 & w_2 & 0 & 0 & w_0 & 0 & 0 & w_1 & 0 & 0 & 0 \\ 0 & w_0 & w_1 & 0 & 0 & w_2 & 0 & 0 & 0 & 0 & 0 & w_0 & w_1 & 0 & 0 & w_2 & 0 & 0 & 0 & 0 \\ 0 & 0 & w_0 & 0 & w_1 & 0 & 0 & 0 & w_2 & w_2 & 0 & 0 & w_0 & 0 & w_1 & 0 & 0 & 0 & 0 & w_2 \\ 0 & 0 & 0 & w_0 & w_1 & 0 & 0 & w_2 & 0 & 0 & 0 & 0 & 0 & w_0 & w_1 & 0 & 0 & w_2 & w_2 & 0 \\ w_2 & 0 & 0 & 0 & 0 & w_0 & 0 & 0 & w_1 & 1 & w_2 & 0 & 0 & 0 & 0 & w_0 & 0 & 0 & 0 & 1 \\ 0 & w_0 & 0 & 0 & 0 & 0 & w_1 & w_2 & 0 & 0 & w_0 & 0 & 0 & 0 & 0 & w_1 & w_2 & 0 & 0 & w_0 \\ w_2 & 0 & 0 & w_0 & 0 & 0 & w_1 & 0 & 0 & w_2 & 0 & 0 & w_0 & 0 & 0 & w_1 & 0 & 0 & w_2 & 0 \\ 0 & w_0 & w_1 & 0 & 0 & w_2 & 0 & 0 & 0 & 0 & w_0 & w_1 & 0 & 0 & w_2 & 0 & 0 & 0 & 0 & w_0 \\ 0 & 0 & w_0 & 0 & w_1 & 0 & 0 & 0 & w_2 & 0 & 0 & w_0 & 0 & w_1 & 0 & 0 & 0 & w_2 & 0 & 0 \\ 0 & 0 & 0 & w_0 & w_1 & 0 & 0 & w_2 & 0 & 0 & 0 & 0 & w_0 & w_1 & 0 & 0 & w_2 & 0 & 0 & 0 \\ w_2 & 0 & 0 & 0 & 0 & w_0 & 0 & 0 & w_1 & w_2 & 0 & 0 & 0 & 0 & w_0 & 0 & 0 & w_1 & w_2 & 0 \end{bmatrix} \tag{2.5}$$

ing the predicted number of active users M broadcasted by basestation. By using the active user prediction, each device can select its own short codeword from predefined codebook $\mathbf{C}_{(L,M)}$ for the data transmission and long codeword from the codebook $\mathbf{D}_{(KL,N)}$ for the pilot transmission. Second, since the codewords for both pilot and data transmission is generated from a single codebook, which means that additional memory for the pilot codebook is unnecessary, it is easy to be implemented for mobile devices.

For codeword selection, since the probability of the code collision is minimized when the selection probabilities of codeword selection are equal, we employ the simple method based on the modulo-based mapping strategy. For the data transmission, the column vector $\mathbf{c}_{f(i)}$ is selected in $\mathbf{C}_{(L,M)}$, where $f(i) = ((i-1) \bmod M) + 1$ and $a \bmod b$ is the remainder of the Euclidean division of a by b.

For the pilot transmission, the codebook $\mathbf{D}_{(KL,N)}$ is used as a base codebook and modified versions of this codebook are used as codebooks for different pilot transmission regions. The device i selects the i-th column of the codebook which corresponds to the pilot region as a codeword. Because of the sparsity of LDS code, correlated energy between columns of the base codebook, $\|\mathbf{d}_l^H \mathbf{d}_m\|_2^2$ ($\mathbf{d}_l$ is l-th column of $\mathbf{D}_{(KL,N)}$), has irregular values depending on the index l, m (i.e., large deviation of the correlated energy). So if only one codebook is used for all the regions, the codeword for each user is same in all the regions and the performance of AUD which is based on the correlated-energy calculation can be negatively affected. In order to solve this problem, each user can use various codewords and the correlated energy between users' codeword can have average value. Any modification rule can be applied in making modified codebooks. In this paper, we simply use the matrices that all columns are shifted to the right for specific times from $\mathbf{D}_{(KL,N)}$ as codebooks. The specific shifted times are determined by index of pilot region. Detailed will be explained in the next chapter. For notation simplicity, we skip the subscript $(\cdot)_{(KL,N)}$ of $\mathbf{D}_{(KL,N)}$ in the sequel.

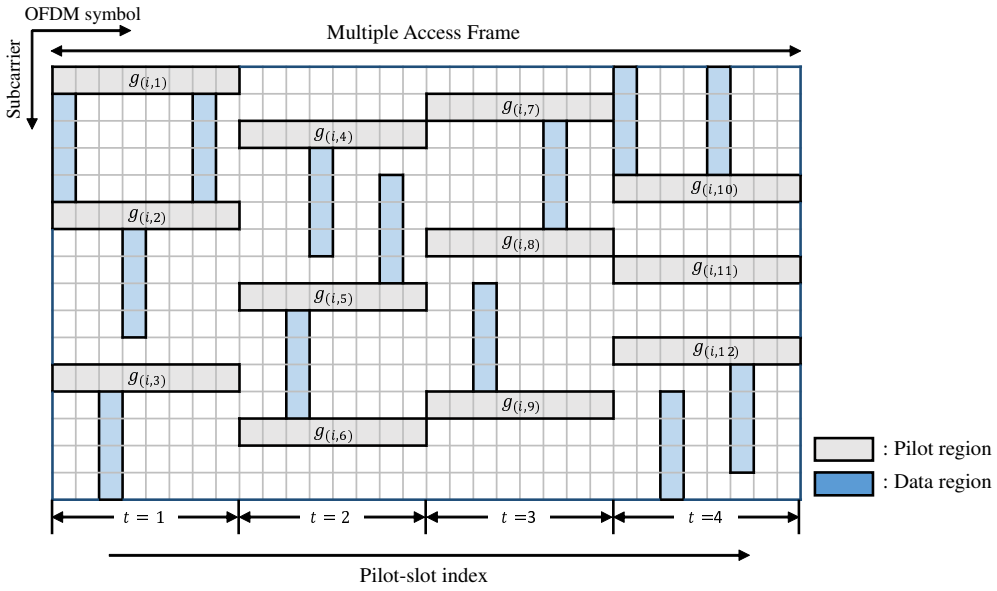

Figure 2.1: Pilot allocation example in an OFDM systems with $N_f = 3$ and $N_t = 4$.

Fig. 2.1 depicts an example of OFDM resource grid which consists of $N_{FFT} = 16$ subcarriers, and $N_{SYM} = 32$ OFDM symbols. Each grid represents one resource element and a uplink symbol is spread at the shaded region which consists of multiple resource elements. Locations and the number of regions are predetermined by basestation and all users share the regions to transmit the pilot and data. Entire transmission frame consists of $N_t = 4$ pilot-slots and each slot consists of $N_f = 3$ pilot regions. The remaining $(N_{FFT} - N_f) = 13$ subcarriers are used as data subcarriers.

Note that since most of IoT devices are settled down, it is a reasonable assumption that there is small doppler effect and not much transition of frequency response in one slot. If we set the pilot regions in the direction of time symbol (see Fig. 2.1), we can consider a block fading channel in one pilot region. For channel estimation scheme which will be explained in the next chapter, the received signal from all regions is used. In order to use received signal from multiple regions, it is needed to number the regions. Let r be the index of pilot region and $g_{(i,r)}$ be the channel for i-th user in

region r, respectively. Note that the index of region r increase from top to bottom and left to right slot. Also, as we use different codebooks for each pilot region, let $\mathbf{D}_r$ be the codebook for region r that all columns are shifted to the right for r times from $\mathbf{D}$, $\mathbf{d}_{r,i}$ be the i-th user's codeword for region r, which is i-th column of $\mathbf{D}_r$, and $\mathbf{y}_r$ be the vector forms of received pilot in region r, then we have

$$\begin{aligned} \mathbf{y}_r &= \sum_{i=1}^{N} \mathbf{d}_{r,i} g_{(i,r)} p_i + \mathbf{v}_r && (2.6) \\ &= \mathbf{D}_r \mathbf{G}_r \mathbf{p} + \mathbf{v}_r, \quad r = 1, \ldots, R && (2.7) \end{aligned}$$

where $\mathbf{G}_r = \mathrm{diag}([g_{(1,r)}, \cdots, g_{(N,r)}])$ is the channel matrix, $\mathbf{p} = [p_1, \cdots, p_N]^T$ is the vector of pilot symbols, and $R(= N_f \times N_t)$ is the total number of pilot regions, respectively. The p_i is the i-th user's pilot symbol whose element is a pilot symbol set $\mathbb{P}\,(= \{p\})$ when it is active or zero otherwise.

Unlike the pilot transmission, data symbol is spread in frequency direction in the OFDM grid (see Fig. 2.1). If we locate the data region in frequency direction, spread symbol undergo different channels in different subcarriers. This property makes overloaded pattern of multi-users' data in each subcarrier diverse and the possibility to detect users' data symbol increase. As data regions don't need to be numbered, received data signal can be expressed in a general form. Let $\mathbf{z}$ be the vector forms of received data signals in a region, then we have

$$\mathbf{z} = \sum_{i=1}^{N} \mathrm{diag}(\mathbf{g}_i^{(d)}) \mathbf{c}_{f(i)} x_i + \mathbf{v}' \qquad (2.8)$$

where $\mathbf{g}_i^{(d)} \in \mathbb{C}^{L \times 1}$ is a channel vector for user i in a data region, x_i is the data symbol, and $\mathbf{v}'$ is the complex Gaussian noise vector for data signal, respectively.

Chapter 3

Detecting Active users, Channel and Data

3.1 Overall Structure of active user detection and channel estimation

To detect the spread data, the basestation obtains information on the active users and their channels. Fig. 3.1 shows the block diagram of the proposed AUD and channel-estimation scheme. To detect the active user more precisely, the proposed AUD algorithm uses all the R received pilot vectors. By using multiple measurements, pilot signal's energy can be larger than one measurement case and the effect of noise can be averaged out.

The entire algorithm from AUD to cancellation involves three steps and is repeated until the number of detected users is equal to the number of active users expected by basestation. As a first step in the α-th iteration, the algorithm detects the index vector $\boldsymbol{\gamma}_\alpha$ which elements are the indices of active users and estimates their frequency responses $\hat{\mathbf{g}}_i$ for the pilot regions, where $i \in \boldsymbol{\gamma}_\alpha$. Note that, in the AUD process, the index of most credible active user is selected at the first element of vector $\boldsymbol{\gamma}_\alpha$ (i.e., $\gamma_{\alpha,1}$, Detailed procedure will be presented in the following sections). Thus, in step 2, the algorithm estimates the channel impulse response (CIR) only for the user whose index is $\gamma_{\alpha,1}$. After step 2, the frequency responses for the entire multiple access frame,

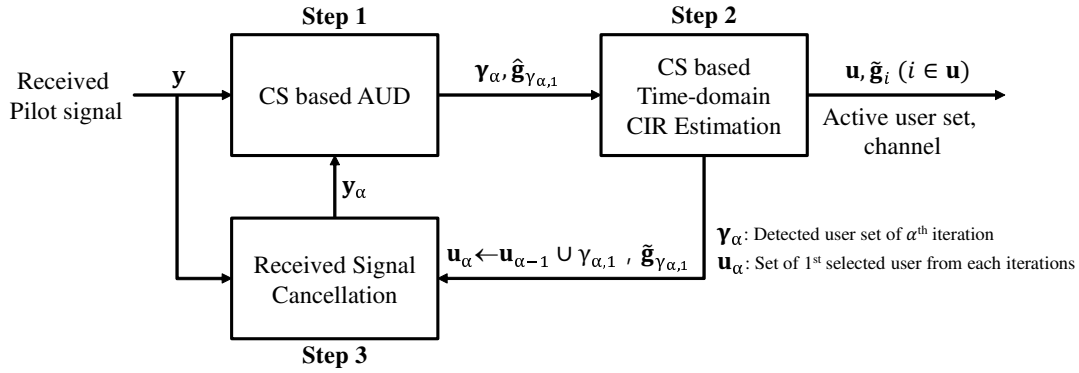

Figure 3.1: Block diagram of joint active user detection and channel estimation

$\tilde{\mathbf{g}}_{\gamma_{\alpha,1}}$, is obtained. Moreover, the user $\gamma_{\alpha,1}$ is added to a vector $\mathbf{u}_\alpha$ which is the set of detected active users' index up to α iterations. Finally, in step 3, the algorithm subtracts the signal of user $\gamma_{\alpha,1}$ from received signal using $\tilde{\mathbf{g}}_{\gamma_{\alpha,1}}$ and $\mathbf{u}_\alpha$. Let $\mathbf{y} = [\mathbf{y}_1^T \cdots \mathbf{y}_R^T]^T$ be the stacked pilot vector that comprises the received vectors from all the regions. Then, the remaining received pilot signal at αth iteration $\mathbf{y}_\alpha$ can be expressed as

$$\mathbf{y}_\alpha = \mathbf{y} - \sum_{i \in \mathbf{u}_\alpha} \left(\begin{bmatrix} \mathbf{d}_{1,i} & 0 & \dots & 0 \\ 0 & \mathbf{d}_{2,i} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \mathbf{d}_{R,i} \end{bmatrix} \begin{bmatrix} \hat{\hat{g}}_{(i,1)} p_i & 0 & \dots & 0 \\ 0 & \hat{\hat{g}}_{(i,2)} p_i & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \hat{\hat{g}}_{(i,R)} p_i \end{bmatrix} \right) \tag{3.1}$$

$$= \mathbf{y} - \sum_{i \in \mathbf{u}_\alpha} \operatorname{diag}(\mathbf{d}_{1,i}, \dots, \mathbf{d}_{R,i}) \operatorname{diag}(\hat{\hat{\mathbf{g}}}_i p_i) \tag{3.2}$$

where $p_i \in \mathbb{P}$ is a pilot symbol and $\hat{\hat{g}}_{(i,r)} (= \tilde{g}_i^k)$ is a re-estimated channel after step 2 in region r, when k is the number which corresponds to region r (e.g., $\hat{\hat{g}}_{(i,2)} = \tilde{g}_i^6$ in Fig.3.2). Then, step 1 is performed again with the subtracted pilot signal $\mathbf{y}_\alpha$. The iteration is over when the number of elements in vector $\mathbf{u}_\alpha$ satisfies the number of active users estimated by the basestation. When the iteration is over, frequency response estimation (FRE) and CIR estimation is done once again with the finally detected active user set. The results of the FRE and CIR estimation can not be accurate during the iteration, because the estimated active users, $\boldsymbol{\gamma}_\alpha$, may not be precise. Because of this

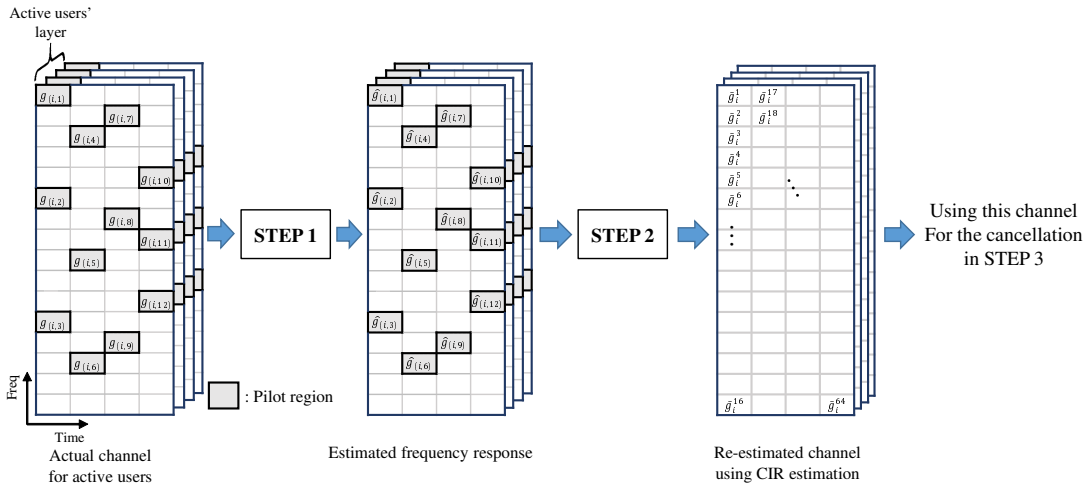

Figure 3.2: Illustration of channel estimation process

reason, the FRE and CIR estimation is done again with more authentic active user set for more accurate channel estimation.

3.2 Joint Active User Detection and Frequency Response Estimation

Since each user has its own distinctive code, the active user information can be obtained by detecting the pilot signal whose elements are the pilot symbol for active users or zero for users in idle state. For joint AUD and FRE, In this section, we employ compressed sensing (CS) algorithm. CS algorithm can recover the sparse signals from only a small number of measurements. Since the pilot signal and corresponding channels can be modeled as sparse vectors, the proposed algorithm simultaneously detects active users and estimates a frequency-domain response of channels. Fig. 3.2 depicts the illustration of the joint AUD and channel estimation method.

Using the codebook matrix for pilot transmission, the equation (2.7) can be ex-

Algorithm 1 CS-based joint active user detection and channel estimation

Input: $\mathbf{y} \in \mathbb{C}^{KLR\times 1}$ (received pilot vector), $\mathbf{D}$ (codebook for pilot transmission) N_e (estimated number of active user)

Initialization $\alpha = 0$, $\mathbf{y}_0 = \mathbf{y}$, $\mathbf{u}_0 = \emptyset$

for $\alpha = 1 : N_e$ **do**

Step 1 (Active User Detection & Frequency Response Estimation)

$N_r = N_e - \alpha + 1$

Input : $\mathbf{y}_{\alpha-1}, N_r$

Do the Algorithm 2

Output : $\boldsymbol{\gamma}, \hat{\mathbf{g}}_i$ $(i \in \boldsymbol{\gamma})$

$\boldsymbol{\gamma}_\alpha = \boldsymbol{\gamma}$

Step 2 (Channel Impulse Response Estimation)

Input : $\hat{\mathbf{g}}_{\gamma_{\alpha,1}}$

Do the Algorithm 3

Output : $\tilde{\mathbf{g}}_{\gamma_{\alpha,1}}$

$\mathbf{u}_\alpha = \mathbf{u}_{\alpha-1} \cup \gamma_{\alpha,1}$

Step 3 (Cancellation)

$\mathbf{y}_\alpha = \mathbf{y} - \sum_{i\in\mathbf{u}_\alpha} \mathrm{diag}(\mathbf{d}_{1,i}, \ldots, \mathbf{d}_{R,i})\, \mathrm{diag}(\hat{\hat{\mathbf{g}}}_i p_i)$

end for

With the indices set $\mathbf{u}_{N_e}$, do the frequency response estimation with LMMSE and CIR estimation for all users in $\mathbf{u}_{N_e}$.

Output: $\mathbf{u} = \mathbf{u}_{N_e}$ (finally estimated active users' index)

$\tilde{\mathbf{g}}_i \quad i \in \mathbf{u}$ (estimated frequency response for entire frame)

pressed as

$$\mathbf{y}_r = \mathbf{D}_r\mathbf{G}_r\mathbf{p} + \mathbf{v}_r \tag{3.3}$$

$$= \mathbf{D}_r\mathbf{q}_r + \mathbf{v}_r, \quad r = 1, \ldots, R \tag{3.4}$$

where $\mathbf{q}_r = [g_{(1,r)}p_1, \ldots, g_{(N,r)}p_N]^T \in \mathbb{C}^N$ is the vector of channels and pilot symbols.

Since the number of active users transmitting the pilot symbol is relatively smaller

than the number of total users, the vector $\mathbf{q}_r$ becomes sparse. By finding the indices of non-zero elements (a.k.a., support set) using observation $\mathbf{y}_r$ and codebook $\mathbf{D}_r$ as a sensing matrix, the AUD can distinguish between active and inactive users. This is because columns corresponding to the zero element in $\mathbf{q}_r$ can be removed from the system model so that the underdetermined system can be converted into overdetermined system. As a result, the solution for the overdetermined system generates an accurate estimate of the original sparse vector ($\hat{\mathbf{q}}_r$) and frequency response ($\hat{\mathbf{q}}_r/p$). From (3.4), the stacked pilot vector $\mathbf{y} = [\mathbf{y}_1^T \cdots \mathbf{y}_R^T]^T$ can be expressed as

$$\mathbf{y} \;=\; \mathrm{diag}(\mathbf{D}_1, \cdots, \mathbf{D}_R) \begin{bmatrix} \mathbf{q}_1 \\ \vdots \\ \mathbf{q}_R \end{bmatrix} + \begin{bmatrix} \mathbf{v}_1 \\ \vdots \\ \mathbf{v}_R \end{bmatrix}. \tag{3.5}$$

We define a vector $\mathbf{w}_i \in \mathbb{C}^{R\times 1}$ as

$$\mathbf{w}_i \;=\; [q_{1,i} \cdots q_{R,i}]^T = [p_i g_{(i,1)} \cdots p_i g_{(i,R)}]^T, \quad i = 1, \ldots, N \tag{3.6}$$

which contains the ith user's channel gain of all the R pilot regions. Eq. (3.5) can be rewritten as

$$\mathbf{y} \;=\; [\mathbf{\Lambda}_1 \ldots \mathbf{\Lambda}_N] \begin{bmatrix} \mathbf{w}_1 \\ \vdots \\ \mathbf{w}_N \end{bmatrix} + \begin{bmatrix} \mathbf{v}_1 \\ \vdots \\ \mathbf{v}_R \end{bmatrix}, \tag{3.7}$$

where $\mathbf{\Lambda}_i \in \mathbb{C}^{(KLR)\times R}$ forms a new sensing matrix that corresponds to the rearranged vector $\mathbf{w}_1, \ldots, \mathbf{w}_N$, rather than $\mathbf{q}_1, \ldots, \mathbf{q}_R$. In other words, $\mathbf{\Lambda}_i$ is the collection of the columns in $\mathrm{diag}(\mathbf{D}_1, \cdots, \mathbf{D}_R)$ which correspond to $q_{1,i} \cdots q_{R,i}$. In order to check wheter the location contains signal or not, we introduce a vector called the *support vector* $\boldsymbol{\delta} = [\delta_1 \ldots \delta_N]^T$. Note that $\delta_i = 1$ when the ith user is in the support (i.e., active user), and $\delta_i = 0$ otherwise. Using the support vector $\boldsymbol{\delta}$, (3.7) can be rewritten

Algorithm 2 CS-based joint active user detection and frequency response estimation

Input: $\mathbf{y} \in \mathbb{C}^{KLR\times 1}$ (received signal), $[\mathbf{\Lambda}_1 \dots \mathbf{\Lambda}_N] \in \mathbb{C}^{(KLR)\times(RN)}$ (sensing matrix), N_r (estimated number of active user)

Definition $\mathbf{a}^k \in \mathbb{C}^{KLR\times 1}$ (residual signal vector at the kth iteration), $\boldsymbol{\delta}^k \in \mathbb{C}^{N\times 1}$ (support vector at the kth iteration), $\hat{\mathbf{w}}_i^k$ (LMMSE estimate of $\mathbf{w}_i$ at the kth iteration).

Initialization $\mathbf{a}^0 = \mathbf{y}$, $k = 0, \boldsymbol{\delta}^0 = \mathbf{0}_N$, $\boldsymbol{\gamma} = \mathbf{0}_{N_r}$

while $\|\boldsymbol{\delta}^k\|_0 < N_r$ **do**

$k = k + 1$

$s_{max} = \underset{s=1,\dots,N}{\operatorname{argmax}} \|\mathbf{\Lambda}_s^H \mathbf{a}^{k-1}\|_2^2$ (selection of index corresponding to largest inner product)

$\gamma_k = s_{max}$

$\boldsymbol{\delta}^k = \boldsymbol{\delta}^{k-1}$ but $\delta^k_{s_{max}} = 1$ (addition of new support)

$\hat{\mathbf{w}}_i^k = E[\mathbf{w}_i \mathbf{y}^H] E[\mathbf{y}\mathbf{y}^H]^{-1}\mathbf{y}$

$= [\delta_i^k P_i \mathbf{\Lambda}_i^H][\sum_{i=1}^N \delta_i^k P_i \mathbf{\Lambda}_i \mathbf{\Lambda}_i^H + \sigma_v^2 \mathbf{I}]^{-1}\mathbf{y}, \quad i = 1, \dots, N$ (LMMSE estimation)

$\mathbf{a}^k = \mathbf{y} - \sum_{i=1}^N \mathbf{\Lambda}_i \delta_i^k \hat{\mathbf{w}}_i^k$ (residual update)

end while

$\hat{\mathbf{w}}_i = \hat{\mathbf{w}}_i^{N_r}$

Output: $\boldsymbol{\gamma}$ (set of estimated active user indices)

$\hat{\mathbf{g}}_i = [\hat{g}_{(i,1)}, \cdots, \hat{g}_{(i,R)}]^T = \hat{\mathbf{w}}_i / p, \;\; i = 1, \dots, N$, (estimated frequency response vector).

as

$$\mathbf{y} \;=\; [\mathbf{\Lambda}_1 \dots \mathbf{\Lambda}_N] \begin{bmatrix} \delta_1 \mathbf{w}_1 \\ \vdots \\ \delta_N \mathbf{w}_N \end{bmatrix} + \begin{bmatrix} \mathbf{v}_1 \\ \vdots \\ \mathbf{v}_R \end{bmatrix}, \tag{3.8}$$

where it is assumed that the non-active user have zero energies. With this system model, the AUD is performed according to Algorithm 2.

3.3 Channel Impulse Response Estimation

By using AUD and FRE, we can obtain the frequency response in all the pilot region for the estimated active users. With this information, we can apply another novel sparse channel estimation technique that exploits the common support of the consecutive channel impulse responses (CIR) over the one multiple access frame [15]. After AUD and FRE, we only know the frequency responses in pilot region. But, in order to decode the data which is transmitted in a data region, frequency response for whole bandwidth and whole transmission frame should be obtained. With the CIR estimation, we can achieve the frequency response for data regions and pilot regions as well. In the IoT devices, a small number of uplink pilot regions are places at randomly chosen locations in time and frequency resource grid. As a result of the random allocation, we can construct so called a random *sensing matrix*, which retains desirable properties in recovering the unknown signal vector from a small number of measurements. Based on the observation that the support of the CIR vector rarely changes during the time period of several OFDM symbols, this scheme finds the common support of the several measured CIR vectors. The whole algorithm is quite similar with the AUD and FRE algorithm which is suggested in previous section. By estimating the CIR vector, we can finally estimate the whole bandwidth frequency response (see Step 2 in Fig. 3.1 and 2nd and 3rd illustration in Fig. 3.2). Also, by using the side information that the CIR vector is sparse during transmission frame, more precise frequency response can be obtained after the CIR estimation. In other words, the estimated frequency response which has lower mean square error can be achieved. So, by using this estimated result in the AUD part again, the performance of AUD can be better than the performance of non-feedback cases. Unlike the conventional OFDM systems where each resource elements for pilot is monopolized by one user, each pilot region is shared by multiple users in our algorithm. As our goal is to estimate the channel of respective users, if we represent the effective pilot signal received from user i at t pilot slot, $\hat{\mathbf{w}}_i^{(t)}$, can be expressed by pilot symbol and estimated frequency response which is the output of

Algorithm 2.

As our goal is to estimate the channel of respective users, if we represent the effective pilot signal received from user i at pilot slot t, $\hat{\mathbf{w}}_i^{(t)} = [p_i\hat{g}_{(i,N_f(t-1)+1)} \cdots p_i\hat{g}_{(i,N_ft)}]^T$, can be expressed by pilot symbol and estimated frequency response which is the output of Algorithm 2. The $\hat{\mathbf{w}}_i^{(t)}$ is subset of $\hat{\mathbf{w}}_i$, which is vector of elements corresponding to pilot slot t in $\hat{\mathbf{w}}_i$ (i.e., $\hat{\mathbf{w}}_i^{(t)} = \{p_i\hat{g}_{(i,r)} | N_f(t-1)+1 < r < N_ft\}$). Also we can express the relationship between $\hat{\mathbf{w}}_i^{(t)}$ and real channel as

$$\hat{\mathbf{w}}_i^{(t)} = \mathbf{w}_i^{(t)} + \mathbf{n}_i^{(t)} \tag{3.9}$$

$$= \mathrm{diag}(\tilde{\mathbf{p}})\mathbf{g}_i^{(t)} + \mathbf{n}_i^{(t)}, \quad i = 1, \ldots, N \tag{3.10}$$

where $\mathbf{w}_i^{(t)} = [p_ig_{(i,N_f(t-1)+1)} \cdots p_ig_{(i,N_ft)}]^T$ is the vector of channels and pilot symbols, $\tilde{\mathbf{p}} \in \mathbb{P}^{N_f \times 1}$ is the transmitted pilot vector in one slot, $\mathbf{g}_i^{(t)} = [g_{(i,N_f(t-1)+1)} \cdots g_{(i,N_ft)}]^T$ is real frequency response of channels at the t-th pilot slot, and $\mathbf{n}_i^{(t)} \in \mathbb{C}^{N_f \times 1}$ is the additive noise. Note that the additive noise, $\mathbf{n}_i = [(\mathbf{n}_i^{(1)})^T \ldots (\mathbf{n}_i^{(R)})^T]^T$, can be expressed as

$$\mathbf{n}_i = \hat{\mathbf{w}}_i - \mathbf{w}_i \tag{3.11}$$

which is an error of LMMSE estimation in Algorithm 2. In [16], the error covariance matrix after LMMSE is given by

$$\mathbf{C}_n = E[\mathbf{w}_i\mathbf{w}_i^H] - E[\mathbf{w}_i\mathbf{y}^H]E[\mathbf{y}\mathbf{y}^H]^{-1}E[\mathbf{y}\mathbf{w}_i^H], \tag{3.12}$$

and the covariance matrix is used in the CIR estimation algorithm which is going to be accounted later.

Let $\eta^{(t)} = \{c_{t,1}, \ldots, c_{t,N_f}\}$ be the set of indices of N_f pilot regions belonging to the tth pilot slot ($1 \le c_{t,f} \le N_{FFT}$). Due to random pilot region allocation, $\eta^{(t)}$ changes with t. However, since an additional information on pilot location is required for random allocation at both transmitter and receiver side, we instead use 'pseudo-random' pilot location that appear to be random but actually the location is chosen by the deterministic process.

Let $\mathbf{h}_i^{(t)} = [h_{i,1}^{(t)} \dots h_{i,N_{CIR}}^{(t)}]^T$ be the time domain CIR vector in the t-th slot, where N_{CIR} represent the length of the time domain CIR vector. Note that we assumed that the support (non zero indices) of the several measured CIR vector is varying slowly, and hence their supports can be assumed to be constant over the transmission duration (e.g., from $t = 1$ to $t = 4$ in Fig. 3.1). However, due to multipath fading of wireless channel, the value of $\mathbf{h}_i^{(t)}$ changes with some temporal correlation in respective t. From the Jakes' fading model, the correlation of the CIR vector is given by

$$E[(h_{i,p}^{(k)})(h_{i,q}^{(l)})^*] = \begin{cases} P_p J_0(2\pi f_d T_s(k-l)), & \text{for } p = q \\ 0, & \text{for } p \neq q \end{cases} \tag{3.13}$$

where $1 \leq p, q \leq N_{CIR}$, $P_p = E[|h_{i,p}^{(k)}|^2]$ is the variance of p-th component, $J_0(x)$ is the zero-order Bessel function of the first kind, f_d is Doppler frequency, and T_s is the time interval between neighbouring pilot slots.

The effective received pilot vector $\hat{\mathbf{w}}_i^{(t)}$, given by (3.10), can be rewritten using CIR vector and DFT matrix as

$$\hat{\mathbf{w}}_i^{(t)} = \mathrm{diag}(\tilde{\mathbf{p}})\mathbf{\Phi}^{(t)}\mathcal{F}_{N_{FFT}} \begin{bmatrix} \mathbf{h}_i^{(t)} \\ \mathbf{0}_{N_{FFT}-N_{CIR}} \end{bmatrix} + \mathbf{n}_i^{(t)} \tag{3.14}$$

$$= \underbrace{\mathrm{diag}(\tilde{\mathbf{p}})\mathbf{\Phi}^{(t)}\mathcal{F}_{N_{FFT}}\mathbf{\Pi}}_{=\mathbf{U}^{(t)}} \mathbf{h}_i^{(t)} + \mathbf{n}_i^{(t)} \tag{3.15}$$

where $\mathcal{F}_{N_{FFT}} \in \mathbb{C}^{N_{FFT} \times N_{FFT}}$ is the DFT matrix with (k,l)th entry given by $\exp(-j2\pi kl/N_{FFT})$, and $\mathbf{\Phi}^{(t)} \in \mathbb{R}^{N_f \times N_{FFT}}$ and $\mathbf{\Pi} \in \mathbb{R}^{N_{FFT} \times N_{CIR}}$ are the matrices consisting of coordinate vectors. Denoting the kth coordinate vector of length N_{FFT} as $\mathbf{e}_k$, we have $\mathbf{\Phi}^{(t)} = [\mathbf{e}_{c_{t,1}} \dots \mathbf{e}_{c_{t,N_f}}]^T$ and $\mathbf{\Pi} = [\mathbf{e}_1 \dots \mathbf{e}_{N_{CIR}}]$. In (3.15), $\boldsymbol{\gamma}^{(t)}$ can be regarded as sensing matrix between the measurement vector $\hat{\mathbf{w}}_i^{(t)}$ and the sparse vector $\mathbf{h}_i^{(t)}$. With this model, the channel estimation can be converted into the recovery of the unknown time-domain CIR vector $\mathbf{h}_i^{(t)}$ with the knowledge of $\mathbf{U}^{(t)}$ and $\hat{\mathbf{w}}_i^{(t)}$. Note that $\hat{\mathbf{w}}_i^{(t)}$ can have different values depending on the user index i, but $\mathbf{U}^{(t)}$ should be the same matrix because the values of the matrix are only influenced by the location of pilot region.

To estimate the CIR vector $[\mathbf{h}_i^{(1)} \dots \mathbf{h}_i^{(N_t)}]^T$, the algorithm uses adjacent N_t slots' received pilot vector. For such sparse channels with common support, we utilize the received vectors in several adjacent slots in which the CIR vectors have the common support. From (3.15), the stacked pilot vector $\hat{\mathbf{w}}_i = [(\hat{\mathbf{w}}_i^{(1)})^T \dots (\hat{\mathbf{w}}_i^{(N_t)})^T]^T$ which is also output of Algorithm 2 can be expressed as

$$\hat{\mathbf{w}}_i \;=\; \mathrm{diag}(\mathbf{U}^{(1)}, \cdots, \mathbf{U}^{(N_t)}) \begin{bmatrix} \mathbf{h}_i^{(1)} \\ \vdots \\ \mathbf{h}_i^{(N_t)} \end{bmatrix} + \begin{bmatrix} \mathbf{n}_i^{(1)} \\ \vdots \\ \mathbf{n}_i^{(N_t)} \end{bmatrix}. \tag{3.16}$$

We define a rearranged vector $\mathbf{m}_j \in \mathbb{C}^{N_t \times 1}$ as

$$\mathbf{m}_j \;=\; [h_{i,j}^{(1)} \dots h_{i,j}^{(N_t)}]^T, \quad j = 1, \dots, N_{CIR}. \tag{3.17}$$

Note that $\mathbf{m}_j$ is composed of the jth components of all the CIR vectors. Using $\mathbf{m}_j$, $\hat{\mathbf{w}}_i$ becomes

$$\hat{\mathbf{w}}_i \;=\; [\mathbf{\Sigma}_1, \cdots, \mathbf{\Sigma}_{N_{CIR}}] \begin{bmatrix} \mathbf{m}_1 \\ \vdots \\ \mathbf{m}_{N_{CIR}} \end{bmatrix} + \begin{bmatrix} \mathbf{n}_i^{(1)} \\ \vdots \\ \mathbf{n}_i^{(N_t)} \end{bmatrix}. \tag{3.18}$$

when $\mathbf{\Sigma}_j \in \mathbb{C}^{N_t N_f \times N_t}$ composes a new sensing matrix that corresponds to the new signal vector $\mathbf{m}_1, \dots, \mathbf{m}_{N_{CIR}}$. From (3.13) and (3.17), we have

$$E[\mathbf{m}_p \mathbf{m}_q^H] = \begin{cases} P_p \mathbf{J}_{N_t \times N_t}, & \text{for } p = q \\ \mathbf{0}_{N_t \times N_t}, & \text{for } p \neq q \end{cases} \tag{3.19}$$

where $1 < p, q < N_{CIR}$, $(\cdot)^H$ denotes Hermitian operation, and $\mathbf{J}_{N_t \times N_t}$ is the $N_t \times N_t$ covariance matrix whose (k,l)th entry is given by $J_0(2\pi f_d T_s(k-l))$. Being similar with *support vector* in the previous section, we also suggest the *support vector* $\boldsymbol{\delta} = [\delta_1 \dots \delta_{N_{CIR}}]^T$. Using the support vector $\boldsymbol{\delta}$, (3.18) can be rewritten as

$$\hat{\mathbf{w}}_i \;=\; [\mathbf{\Sigma}_1, \cdots, \mathbf{\Sigma}_{N_{CIR}}] \begin{bmatrix} \delta_1 \mathbf{m}_1 \\ \vdots \\ \delta_{N_{CIR}} \mathbf{m}_{N_{CIR}} \end{bmatrix} + \begin{bmatrix} \mathbf{n}_i^{(1)} \\ \vdots \\ \mathbf{n}_i^{(N_t)} \end{bmatrix}. \tag{3.20}$$

Algorithm 3 CS-based time-domain CIR estimation

Input: $\hat{\mathbf{w}}_i \in \mathbb{C}^{N_f N_t \times 1}$ (received signal), $[\mathbf{\Sigma}_1 \dots \mathbf{\Sigma}_{N_{CIR}}] \in \mathbb{C}^{(N_f N_t)\times(N_{CIR} N_t)}$ (sensing matrix), N_D (the number of dominant components of the CIR vector),

Definition $\mathbf{a}^k \in \mathbb{C}^{N_f N_t \times 1}$ (residual signal vector at the kth iteration), $\boldsymbol{\delta}^k \in \mathbb{C}^{N_{CIR} \times 1}$ (support vector at the kth iteration), $\hat{\mathbf{m}}_j^k$ (LMMSE estimate of $\mathbf{m}_j$ at the kth iteration).

Initialization $\mathbf{a}^0 = \hat{\mathbf{w}}_i$, $k = 0, \boldsymbol{\delta}^0 = \mathbf{0}_{N_{CIR}}$

while $\|\boldsymbol{\delta}^k\|_0 < N_D$ **do**

$k = k + 1$

$s_{max} = \underset{s=1,\dots,N_{CIR}}{\operatorname{argmax}} \|\mathbf{\Sigma}_s^H \mathbf{a}^{k-1}\|_2^2$ (selection of index corresponding to largest inner product)

$\boldsymbol{\delta}^k = \boldsymbol{\delta}^{k-1}$ but $\delta_{S_{max}}^k = 1$ (addition of new support)

$$\hat{\mathbf{m}}_j^k = E[\mathbf{m}_j \hat{\mathbf{w}}_i^H] E[\hat{\mathbf{w}}_i \hat{\mathbf{w}}_i^H]^{-1} \hat{\mathbf{w}}_i$$
$$= [\delta_j^k P_j \mathbf{J}_{N_t \times N_t} \mathbf{\Sigma}_j^H][\textstyle\sum_{j=1}^{N_{CIR}} \delta_j^k P_j \mathbf{J}_{N_t \times N_t} \mathbf{\Sigma}_j \mathbf{\Sigma}_j^H + \mathbf{C}_n]^{-1} \hat{\mathbf{w}}_i, \quad j = 1, \dots, N_{CIR}$$

(LMMSE estimation)

$\mathbf{a}^k = \hat{\mathbf{w}}_i - \sum_{j=1}^{N_{CIR}} \mathbf{\Sigma}_j \delta_j^k \hat{\mathbf{m}}_j^k$ (residual update)

end while

$\hat{\mathbf{h}}_i^{(t)} = [\hat{m}_{1,t}^{N_D} \dots \hat{m}_{N_{CIR},t}^{N_D}]^T, \quad t = 1, \dots, N_t$

Output: $\tilde{\mathbf{g}}_i = [(\mathcal{F}_{N_{FFT}} \mathbf{\Pi} \hat{\mathbf{h}}_i^{(1)})^T \dots (\mathcal{F}_{N_{FFT}} \mathbf{\Pi} \hat{\mathbf{h}}_i^{(N_t)})^T]^T$ (Estimated frequency response in entire multiple access frame)

$\hat{\tilde{\mathbf{g}}}_i = [[\mathbf{\Phi}^{(1)} \mathcal{F}_{N_{FFT}} \mathbf{\Pi} \hat{\mathbf{h}}_i^{(1)}]^T \dots [\mathbf{\Phi}^{(N_t)} \mathcal{F}_{N_{FFT}} \mathbf{\Pi} \hat{\mathbf{h}}_i^{(N_t)}]^T]^T$ (Re-estimated frequency response for pilot regions)

With this rearranged system model, the time-domain CIR estimation is performed according to Algorithm 3.

3.4 Data Detection with MPA

From AUD process, we can estimate the active devices (i.e., $\mathbf{u}$) that are highly likely to transmit the data. Using $\mathbf{u}$, the received data signal vector $\mathbf{z}$ in Eq.(2.8) can be rewritten

as

$$\mathbf{z} = \sum_{i\in\mathbf{u}} \mathrm{diag}(\mathbf{g}_i^{(d)})\mathbf{c}_{f(i)}x_i + \mathbf{v}' \tag{3.21}$$

$$= \mathbf{G}^{(d)} \odot \mathbf{C}'\mathbf{x} + \mathbf{v}', \tag{3.22}$$

where $\mathbf{u}$ is the index set of estimated active devices from the AUD, $\mathbf{G}^{(d)} = [\mathbf{g}_{u_1}^{(d)} \dots \mathbf{g}_{u_{N_e}}^{(d)}]$ is a channel matrix, $\odot$ is element-wise multiplication, $\mathbf{C}' = [\mathbf{c}_{f(u_1)} \dots \mathbf{c}_{f(u_{N_e})}]$ is a codeword matrix of estimated active users, and $\mathbf{x} = [x_{u_1}, \cdots, x_{u_{N_e}}]^T$ is the data symbol vector, respectively.

To detect the vector $\mathbf{x}$, the optimum MAP solution is given by

$$\hat{\mathbf{x}} = \arg\max_{\mathbf{x}\in\mathbb{X}^{N_e}} p\left(\mathbf{x}|\mathbf{z}\right), \tag{3.23}$$

where $\mathbb{X} = \{0, \mathbb{Q}\}$ is set of constellation points. The symbol x_n can also be estimated by calculating the marginal of function $p\left(\mathbf{x}|\mathbf{z}\right)$ and can be expressed as

$$\hat{x}_n = \arg\max_{\alpha\in\mathbb{X}} \sum_{\mathbf{x}\in\mathbb{X}, x_n=\alpha} p\left(\mathbf{x}|\mathbf{z}\right), \forall n \tag{3.24}$$

To filter out the cases of false alarm in AUD results, 0-constellation point is additionally included. Since the MAP detection should calculate all possible combinations of $\mathbf{x}$, the computational complexity of MAP detection is unpractical with large N_e. Also, the effective channel matrix $\mathbf{G}^{(d)} \odot \mathbf{C}'$ in (3.21) is not a square matrix (i.e., $N_e \neq L$) so that the low-complexity ML and MAP detection algorithms, such as sphere decoding are infeasible [13]. For low complexity solution, the MAP problem can be expressed with Bayes' rule and the marginalized product of functions (MPF) as

$$\hat{x}_n = \arg\max_{\alpha\in\mathbb{X}} \sum_{\mathbf{x}\in\mathbb{X}, x_n=\alpha} P(\mathbf{x}) p\left(\mathbf{z}|\mathbf{x}\right) \tag{3.25}$$

$$= \arg\max_{\alpha\in\mathbb{X}} \sum_{\mathbf{x}\in\mathbb{X}, x_n=\alpha} P(\mathbf{x}) \prod_{l\in\zeta_n} p(z_l|\mathbf{x}^{[l]}), \tag{3.26}$$

where ζ_n is set of resource indices that nth data symbol is spread, and $\mathbf{x}^{[l]}$ is the vector of the data symbols transmitted on z_l. The MPF still requires brute-force searching

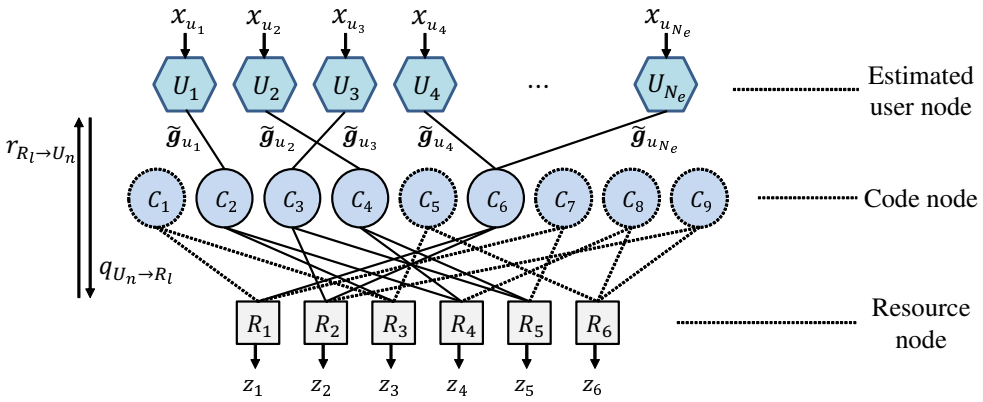

Figure 3.3: Factor graph for MPA decoding, based on active user detection results

among $|\mathbb{X}|^{N_e}$ so that an algorithm called a message passing algorithm (MPA) is well-known for approximating the solution. Since the size of $\mathbf{x}^{[l]}$ is smaller than the size of $\mathbf{x}$ with LDS structure, by factorizing MPF detection as a product of observations of the factor graph, the solution can be utilized after a few iteration stages. While the MPA is a sub-optimal algorithm of MAP, using the sparse relations between data symbol vector $\mathbf{x}$ and received signal $\mathbf{y}$, the computational complexity is much lower than that of MAP with reasonable performance degradation.

In our approach, we modified the MPA algorithm to handle the case when more than one device select the same codeword. While the LDS codebook designed to have symmetric graph between devices and codes, actual factor graph will depend on the randomness of device's traffic. In addition, since the codeword is not uniquely assigned to a device, a codeword can be used by multiples devices and factor graph can be different from what we designed. From the result of AUD, our MPA algorithm update the factor graph by placing devices at the collided code node. (see U_4 and U_{N_e} at code node 6 in Fig. 3.3)

The example of factor graph for MPA decoding is depicted in Fig. 3.3, the message values from R_l to U_n and opposite route (i.e., $r_{R_l \to U_n}$ and $q_{U_n \to R_l}$) are representing specific probabilities. The value $r_{R_l \to U_n}(x)$ is the probability that the value of re-

source element l is satisfied if symbol x_n is considered fixed at x. (i.e., $p(z_l|x_n = x)$) and the value $q_{U_n \to R_l}(x)$ is the probability that symbol x_n has the value x, given the information obtained via resource elements other than resource element l. (i.e., $p(x_n = x|\{z_{l'}\} : l' \in \zeta_n \setminus l)$).

For all $x \in (\mathbb{Q} \cup \{0\})$ the message values ($r_{R_l \to U_n}$, $q_{U_n \to R_l}$) at the i-th iteration are expressed as

$$r^i_{R_l \to U_n}(x) = \sum_{\mathbf{x}^{[l]}: x_n = x} \left(G_l(\mathbf{x}^{[l]}) \prod_{n' \in \xi_l \setminus n} q^{i-1}_{U_{n'} \to R_l}(x_{n'}) \right) \tag{3.27}$$

$$q^i_{U_n \to R_l}(x) = \gamma \prod_{l' \in \zeta_n \setminus l} r^i_{R_{l'} \to U_n}(x), \tag{3.28}$$

where

$$G_l(\mathbf{x}^{[l]}) = \exp(-\frac{1}{2\sigma^2} \| z_l - (\mathbf{g}^{[l]} \odot \mathbf{c}^{[l]})^T \mathbf{x}^{[l]} \|^2) \tag{3.29}$$

is the probability of z_l given $\mathbf{x}^{[l]}$, ξ_l is set of user index which contribute to resource element l, $\mathbf{g}^{[l]}$ and $\mathbf{c}^{[l]}$ are vector extraction of l-th row non-zero elements in matrix $\mathbf{G}^{(d)}$ and $\mathbf{C}'$, respectively. In addition, γ is normalizing factor to satisfy $\sum_x q^i_{U_n \to R_l}(x) = 1$. After a number of iterations, symbol decisions are made by

$$\hat{x}_n = \arg\max_x \{ \prod_{l \in \zeta_n} r^i_{R_l \to U_n}(x) \}. \tag{3.30}$$

The symbol that has maximum probability is finally chosen as estimated symbol.

Chapter 4

Simulation Results and Discussion

In this chapter, we compare the performance of the proposed joint AUD and channel estimation technique with the conventional approaches through numerical simulations.

4.1 Simulation Setup

The simulation setup for massive connectivity is based on the LTE-A system with a single cell. We consider the two cases that there are 100 and 400 devices in the cell for the simulation. In the cell, we randomly choose the active devices which would transmit the data signal to the basestation. For data transmission, we use 5×10 size shared codebook for 100 devices case and 10×20 size shared codebook for 400 devices case. For the pilot transmission, we generate long sequence codebook of which size is 10×100 for 100 devices case and 20×400 for 400 devices case. With the pilot codebook, each pilot symbol is spread into 10 resource element (100 devices case) and 20 resource element (400 devices case), respectively. The total number of transmitted pilot symbols is 30 for both cases. For the channel encoding, we use the one-third ($1/3$) Turbo code with feedback polynomial $1 + D + D^2$ and feedforward polynomial $1 + D^2$. Each code block spans a length of 150 symbols with QPSK modulation. As a data detection algorithm, we employ MPA receiver for LDS to handle non-orthogonal

signal detection.

As a metric to measure the performance, we use the success probability (i.e., the number of detected users divided by the number of total active users) at the output of AUD, mean square error (MSE) at the channel estimation, and block error rate (BLER) at the output of channel decoder. These metrics are measured as a function of $Eb/N0$. We tested the performances with the activity 4%, 5%, and 6% for the test 1 and 2%, 3%, and 4% for the test 2. In our simulation, we consider the conventional and proposed algorithms as follows.

1. Active User Detection

   (a) Conventional AUD : Simple active user detection method is performed based on the CS algorithm. In the conventional AUD algorithm, no iteration is conducted with channel estimator.

   (b) Proposed joint AUD and CIR estimation : Signal cancellation is performed using the active user and channel information from previous iterations.

2. Channel Estimation

   (a) Time-domain CIR estimation : Time-domain CIR estimation is performed based on the CS algorithm. The estimated result is not feed-backed to the AUD process.

   (b) LMMSE CE [17] : Pilot-based channel estimation is performed based on the MMSE criteria. The estimated result is not feed-backed to the AUD process.

   (c) Oracle-based CE : Time domain CIR estimation with perfect CIR position is performed.

   (d) Proposed joint AUD and CIR estimation : Time-domain CIR estimation is performed using the active user information.

The detailed simulation assumptions are described in Table 4.1.

Table 4.1: Details of the Simulation Setup

Parameters	**Test 1**	**Test 2**
Network layout (N-user per cell)	100	400
The number of repetition for pilot codewords (K)	2	2
The number of resource element for data (L)	5	10
Pilot codebook overloading factor ($= N/KL$)	1000%	2000%
Activity	4%,5%,6%	2%,3%,4%
The number of codewords for data(M)	10	20
Data codebook overloading factor ($= M/L$)	200%	200%
Channel	Rayleigh fading channel	Rayleigh fading channel
Antenna configuration	Single antenna per user	Single antenna per user
Modulation	QPSK	QPSK
FFT size	512	512
The number of dominant CIR taps (N_D)	3	3
N_f	10	10
N_t	3	3
Detection algorithm	MPA	MPA
Channel coding	Turbo code	Turbo code
Code rate	$1/3$	$1/3$

4.2 Simulation Results

In Fig. 4.1, we investigates AUD success probability of the active user detection algorithm in the IoT network. The success probability of AUD increases with the $Eb/N0$ due to the fact that as the received signal becomes more and more accurate by a less noise effect, the user's codeword can be detected more precisely by the calculation of correlated energy between received signal and codeword.

As shown in Fig. 4.1, the proposed AUD algorithm outperforms the conventional algorithm. Since the accurate frequency response is available from iterative process, the proposed scheme achieves higher success probability than that of conventional method even in low $Eb/N0$ regime. While the performance of conventional AUD is degraded quite much as the activity increase, the performance degradation of proposed algorithm is much smaller than that of conventional method in low $Eb/N0$ regime. As a result, the proposed method detects 100% of active user (i.e. the success probability is equal to one) at about 5 dB for all activity cases. When the activity is 6%, The proposed scheme detects 95% of active users at 0 dB and 100% at 7 dB. On the other hand, the conventional algorithm detect 95% of active users at 6 dB and can not achieve 100% of active users even in high $Eb/N0$ regime. Note that the conventional AUD has upper performance limit because of the property of non-orthogonal codewords. As the activity increases, a large number of non-orthogonal codewords are overlapped into a small number of resources. In that case, the conventional AUD cannot detect the signal of the active users and as a result the conventional AUD has limit performance bound irrespective of $Eb/N0$. However, by canceling the detected user signal from the received signal, the proposed algorithm provides the accurate active user information.

In Fig. 4.2, we investigate the MSE performance of the channel estimation algorithms in 100 total user case. As shown in Fig. 4.2, the proposed algorithm outperforms the conventional AUD and CIR channel estimator, yielding about 4dB gain at 0.001% MSE in 5% activity case. Note that the performance gain between conventional and proposed algorithm gradually grows as the activity increases. Since the conventional

AUD could not deliver the precise frequency response to CIR estimator due to the inaccuracy of active user information, the quality of channel estimation is degraded in high activity.

In Fig. 4.3, we show the BLER performance comparison between conventional and proposed algorithm. As the MPA is a joint detection algorithm between active users, the BLER performance is strongly influenced by the performance of AUD. Since the proposed algorithm provides accurate active user information, we see that the performance of proposed algorithm is superior to that of conventional AUD scheme. The conventional AUD and LMMSE channel estimation method show the worst BLER performance. Because the LMMSE channel estimation couldn't utilize the sparsity of CIR tap, the quality of estimated channel is worse than the quality of channel estimated by CIR estimation. Even the BLER performance of conventional AUD and CIR estimation is superior to LMMSE CE based algorithm, the conventional AUD algorithm could not provide the accurate active user information. The proposed algorithm could obtain more exact active user information and utilize CIR estimation, as a result the proposed algorithm outperform the conventional algorithms. Although the gain of the proposed method over the conventional scheme is 2 dB at 10^{-1} BLER point, the gain increases and becomes more than 13 dB in the high $Eb/N0$ regime.

We next consider the performance of the proposed method for 400-total user case. We observe from Fig. 4.4 that the proposed method outperforms the conventional AUD algorithm by about 6 dB at 95% AUD success probability. When compared to 100-user case, 400-user cases have more users to be detected and pilot overloading factor is double than that of 100-user case so that the proposed method using iterative AUD and CIR estimation has clear benefit over the conventional AUD. For example, while the proposed algorithm offer about 2 dB performance gain (at 4% activity and 85% AUD success probability) over conventional in 100-user case (see Fig. 4.1), the proposed algorithm offer 8 dB performance gain (at 4% activity and 85% AUD success probability) over conventional in 400-user case.

In Fig. 4.5, we plot the MSE performance for 400 massive user cases. As similar with 100-user case, the proposed methods outperform the conventional AUD-based algorithms. The proposed method offer more than 6 dB performance gain (at 0.001 % MSE) over the conventional algorithm. The gain becomes more with the activity.

In Fig. 4.6, we investigate the BLER performance for 400 massive user cases. We observe a considerable performance gain between the proposed algorithm and conventional AUD-based algorithms. As the conventional AUD and LMMSE CE cannot provide the accurate active user information and channel estimate, the BLER performance of LMMSE CE based algorithm cannot achieve the 10^{-1} BLER even in the high $Eb/N0$ regime. Though the CIR estimation based algorithm can offer the precise CE result, the conventional AUD cannot detect the active users perfectly. As a result, the CIR estimation based algorithm has performance gap with the proposed algorithm. The proposed algorithm offer 2 dB performance gain (at 10^{-1} BLER) and more than 11 dB performance gain (at 10^{-2} BLER) over the conventional algorithm.

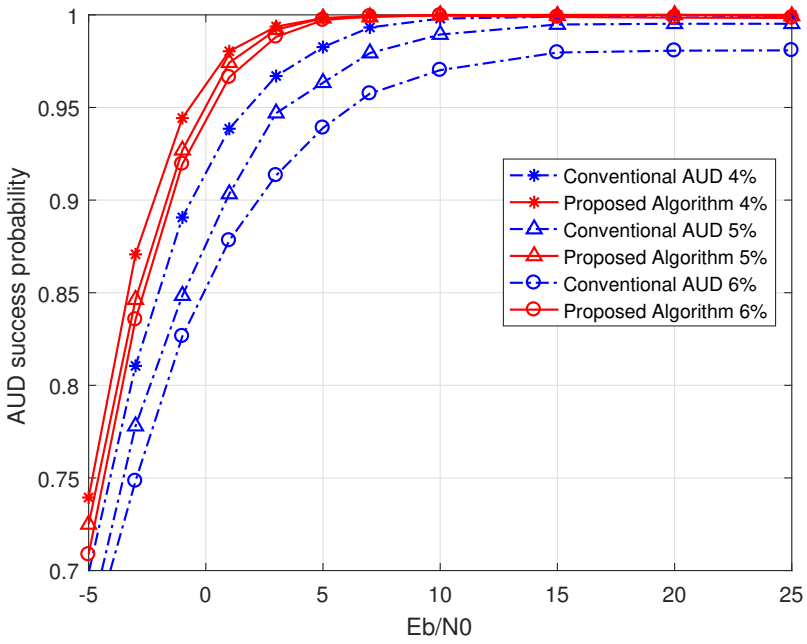

Figure 4.1: AUD success probability performance (Total user: 100)

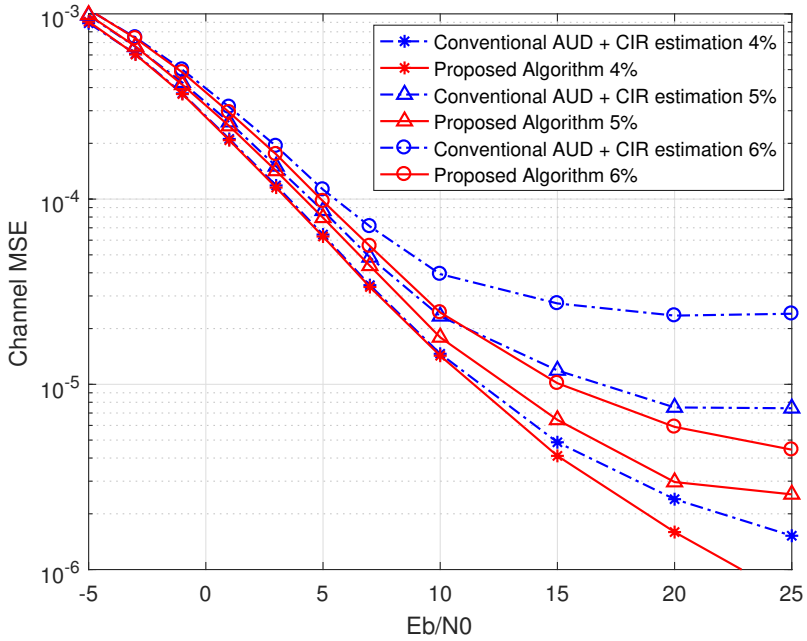

Figure 4.2: Channel MSE performances (Total user: 100)

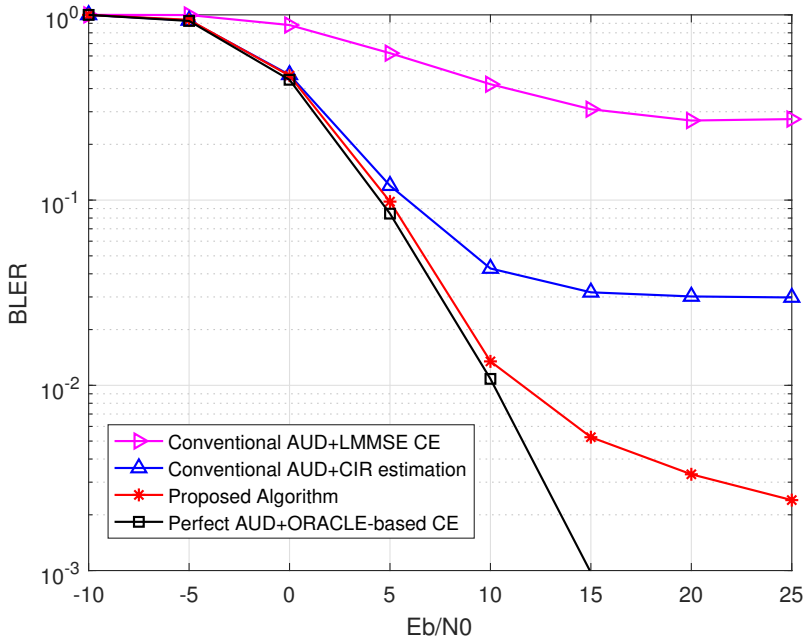

Figure 4.3: BLER performances (Total user: 100, activity : 6%)

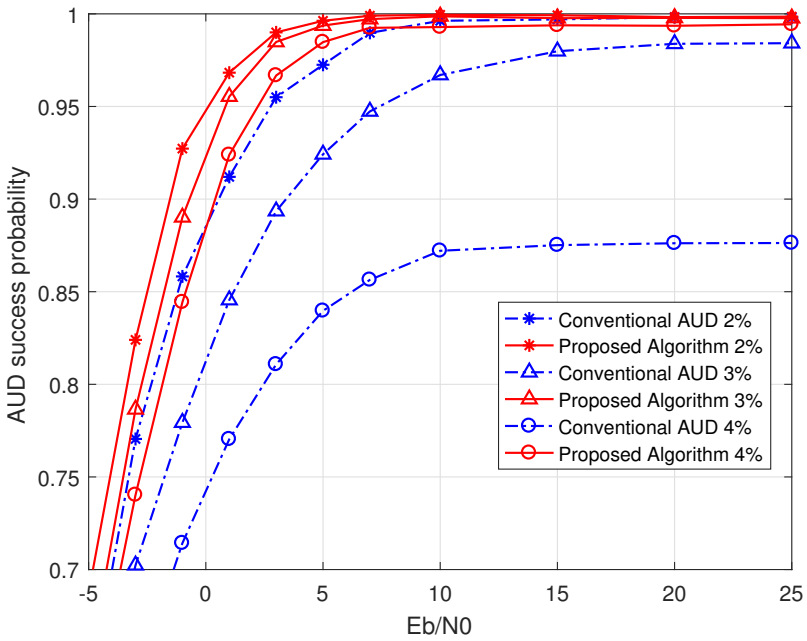

Figure 4.4: AUD success probability performance (Total user: 400)

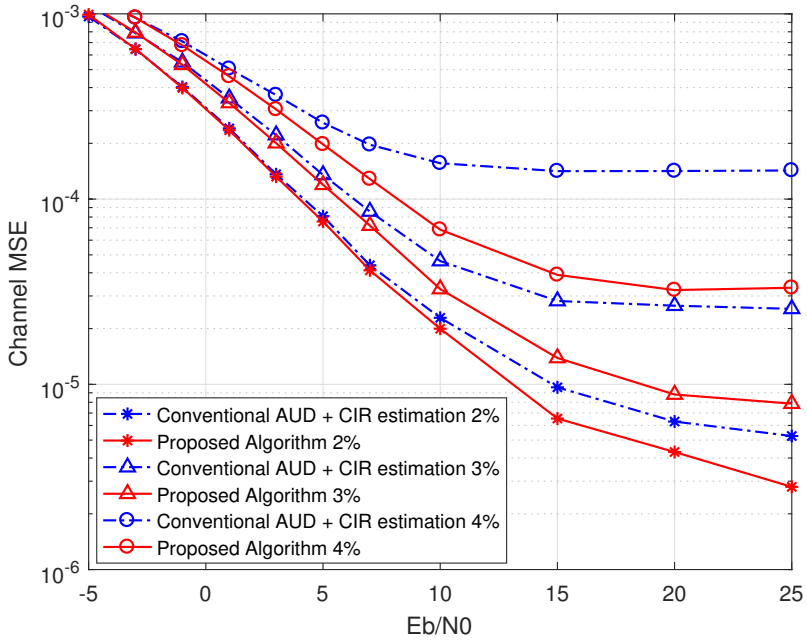

Figure 4.5: Channel MSE performances (Total user: 400)

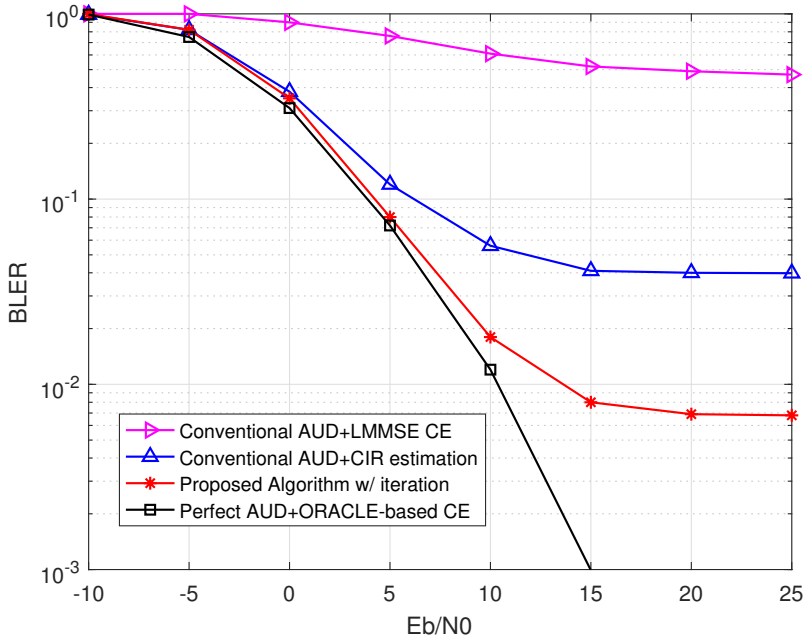

Figure 4.6: BLER performances (Total user: 400, activity : 3%)

Chapter 5

Summary and Conclusions

In this thesis, we proposed a novel grant-free based sparse spreading multiple access technique. By employing the CS based joint AUD-CIR estimation, the proposed method achieves better AUD accuracy, channel estimation quality and eventual BLER performance improvement using adaptive MPA detection. Key features of the proposed method are 1) CS-based AUD using sparsity of active user, 2) CS-based CIR estimation using sparsity of CIR taps and 3) iterative utilization of estimated CIR information in AUD process. In particular, when there are more active users, the proposed algorithm achieves more substantial gain in the AUD success probability over the conventional AUD method. We observed from the BLER results that the proposed approach uniformly outperforms conventional algorithms based on naive AUD and channel estimation, which ensures that the proposed scheme is desirable in practical massive IoT network. Future study needs to be directed towards the investigation of the performance when the multiple access is performed under the non-synchronized assumption.

Bibliography

[1] L. Coetzee and J. Eksteen, “The Internet of Things - promise for the future? An introduction,” *IST-Africa Conference Proceedings*, 2011, Gaborone, 2011, pp. 1-9.

[2] M.2083 : IMT Vision - ”Framework and overall objectives of the future development of IMT for 2020 and beyond”

[3] Evolved Universal Terrestrial Radio Access (E-UTRA); Physical Layer Procedures (Release 12), 3GPP TS 36.213 V12.3.0

[4] R. Hoshyar, F. P. Wathan and R. Tafazolli, “Novel Low-Density Signature for Synchronous CDMA Systems Over AWGN Channel,” in *IEEE Transactions on Signal Processing*, vol. 56, no. 4, pp. 1616-1626, April 2008.

[5] J. van de Beek and B. M. Popovic, “Multiple Access with Low-Density Signatures,” *Global Telecommunications Conference, 2009. GLOBECOM 2009. IEEE*, Honolulu, HI, 2009, pp. 1-6.

[6] S. Chang and J. E.J. Weldon, “Coding for T-user multiple-access channels,” *IEEE Trans. Inform. Theory*, vol. 25, no. 6, pp. 684–691, 1979.

[7] H. Nikopour and H. Baligh, “Sparse code multiple access,” *Personal Indoor and Mobile Radio Communications (PIMRC), 2013 IEEE 24th International Symposium on*, London, 2013, pp. 332-336.

[8] K. Au et al., "Uplink contention based SCMA for 5G radio access," *Globecom Workshops (GC Wkshps), 2014*, Austin, TX, 2014, pp. 900-905.

[9] A. Bayesteh, E. Yi, H. Nikopour and H. Baligh, "Blind detection of SCMA for uplink grant-free multiple-access," *Wireless Communications Systems (ISWCS), 2014 11th International Symposium on*, Barcelona, 2014, pp. 853-857.

[10] J. Wang, S. Kwon, P. Li and B. Shim, "Recovery of Sparse Signals via Generalized Orthogonal Matching Pursuit: A New Analysis," in *IEEE Transactions on Signal Processing*, vol. 64, no. 4, pp. 1076-1089, Feb.15, 2016.

[11] D. J. MacKay, "Good error-correcting codes based on very sparse matrices," *IEEE Trans. Inform. Theory*, vol. 45, no. 2, pp. 399–431, Mar. 1999.

[12] M. Taherzadeh, H. Nikopour, A. Bayesteh and H. Baligh, "SCMA Codebook Design," *Vehicular Technology Conference (VTC Fall), 2014 IEEE 80th*, Vancouver, BC, 2014, pp. 1-5.

[13] B. Shim and I. Kang, "Sphere Decoding With a Probabilistic Tree Pruning," in *IEEE Transactions on Signal Processing*, vol. 56, no. 10, pp. 4867-4878, Oct. 2008.

[14] S. Sparrer and R. F. H. Fischer, "MMSE-based version of OMP for recovery of discrete-valued sparse signals," in *Electronics Letters*, vol. 52, no. 1, pp. 75-77, Jan. 2016.

[15] J.W. Choi, B. Shim and S.H. Chang, "Downlink Pilot Reduction for Massive MIMO Systems via Compressed Sensing," in *IEEE Communications Letters*, vol. 19, pp. 1889-1892, Nov. 2015.

[16] S. M. Kay, "Fundamentals of Statistical Signal Processing: Estimation Theory," Prentice Hall, 1993.

[17] M. K. Orzdemir and H. Arslan, “Channel estimation for wireless OFDM systems,” in *IEEE Commun. Surveys & Tutorials*, vol. 9, no. 2, pp. 18-48, 2nd Quarter. 2007.

초 록

차세대 5G 네트워크에서 핵심 시나리오인 Massive machine type communication (mMTC)을 위한 대용량 다중접속 기술 (massive connectivity)기술이 최근 학계나 업계에서 많은 주목을 받고 있다.

본 논문에서는 massive connectivity를 지원하기 위해 다수의 사용자를 적은 자원을 활용하여 과적(overloading)시키는 희소확산 다중접속(sparse spreading multiple access)기법을 제안하였다. 제안하는 기법에서는 데이터와 파일럿 신호 전송을 위하여 저밀도 확산기법(low density signature)을 사용하였고, 압축센싱(compressed sensing)기반의 활성단말검출(active user detection)과 채널 추정(channel estimation) 알고리즘을 제안하였다. 최종적으로 시뮬레이션 결과를 통하여 본 논문에서 제안하는 기법이 기존의 활성단말검출 알고리즘과 채널 추정알고리즘에 비해 성능 개선이 있음을 보였다.

주요어: IoT, Massive connectivity, Low density signature, Active User Detection, Non orthogonal multiple access

학번: 2015-20936