Ph.D. DISSERTATION

Sparsity-Aware Multi-User Detection for Massive Machine-Type Communications

대규모 사물 통신을 위한 압축센싱 기반 다중 사용자 검출

BY

Jinyoup Ahn

FEBRUARY 2019

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

Abstract

Massive machine-type communication (mMTC) is a newly introduced service category in 5G wireless communication systems to support a variety of Internet-of-Things (IoT) applications. In the mMTC network, a large portion of devices is inactive and hence does not transmit data. Thus, the transmit vector consisting of data symbols of both active and inactive devices can be readily modeled as a sparse vector. In recovering sparsely represented multi-user vectors, compressed sensing based multi-user detection (CS-MUD) can be used. CS-MUD is a feasible solution to the grant-free uplink non-orthogonal multiple access (NOMA) environments. In this dissertation, two novel techniques regarding CS-MUD for mMTC networks are proposed.

In the first part of the dissertation, the sparsity-aware ordered successive interference cancellation (SA-OSIC) technique is proposed. In CS-MUD, multi-user vectors are detected based on a sparsity-aware maximum a posteriori probability (S-MAP) criterion. To reduce the computational complexity of S-MAP detection, sparsity-aware successive interference cancellation (SA-SIC) can be used. SA-SIC is a simple low-complexity scheme that recovers transmit symbols in a sequential manner. However, SA-SIC does not perform well without proper layer sorting due to error propagation. When multi-user vectors are sparse and each device is active with a distinct probability, the detection order determined solely by channel gains might not be optimal. In this dissertation, to reduce the error propagation and enhance the performance of SA-SIC, an activity-aware sorted QR decomposition (A-SQRD) algorithm that finds the optimal detection order is proposed. The proposed technique finds the optimal detection order based on the activity probabilities and channel gains of machine-type devices. Numerical results verify that the proposed technique greatly improves the performance of SA-SIC.

In the second part of the dissertation, the expectation propagation based joint AUD

and CE (EP-AUD/CE) technique is proposed. In several studies regarding CS-MUD, the uplink channel state information (CSI) from the MTD to the BS is assumed to be perfectly known to the BS. In practice, however, the uplink CSI from the devices to the BS should be estimated before data detection. To address this issue, various joint active user detection (AUD) and channel estimation (CE) schemes have been proposed. Since only a few devices are active at one time, an element-wise (i.e., Hadamard) product of the binary activity pattern and the channel vector is also a sparse vector and thus compressed sensing (CS)-based technique is a good fit for the problem at hand. One potential shortcoming in these studies is that a prior distribution of the sparse vector is not exploited. In fact, these studies are based on the non-Bayesian greedy algorithms such as the orthogonal matching pursuit (OMP) and approximate message passing (AMP) algorithms, which do not require a prior distribution of the sparse vector. In essence, these algorithms find out non-zero values based on the instantaneous correlation between the sensing matrix and the observation vector so that they might not be effective in the situation where the prior distribution is available. In this case, clearly, by exploiting the statistical distribution of the sparse vector, the performance of AUD and CE can be improved substantially. The proposed technique finds the best approximation of the posterior distribution of the sparse channel vector based on the expectation propagation (EP) algorithm. Using the approximate distribution, AUD and CE are jointly performed. Numerical simulations show that the proposed technique substantially enhances AUD and CE performances over competing algorithms.

keywords: Massive machine-type communication, compressed sensing, non-orthogonal multiple access, multi-user detection.

student number: 2012-20805

Contents

List of Tables

List of Figures

Chapter 1

Introduction

Recently, massive machine-type communication (mMTC) has received much attention due to its wide variety of Internet-of-Things (IoT) applications such as smart metering, factory automation, autonomous driving, surveillance, and health monitoring, to name just a few [1]. In accordance with this trend, the International Telecommunication Union (ITU) defined mMTC as one of the key service categories for 5G wireless communications [2]. As illustrated in Fig. 1.1, mMTC concerns the *massive connectivity* of a large number of machine-type devices (MTDs) to the base station (BS). mMTC is distinctive from human-centric communications in the sense that the data traffic is *uplink-dominated*, and devices are *sporadically active* only for a short period of time to transmit *short packets* with *low data rates* [3, 4].

In the mMTC network, conventional scheduling-based multiple access schemes in which the BS allocates orthogonal time/frequency resources to each device is not relevant due to the significant signaling overhead and excessive latency caused by complicated scheduling procedure. To overcome these drawbacks, grant-free *non-orthogonal multiple access* (NOMA) schemes have been proposed in recent years [5–7]. In grant-free NOMA schemes, devices transmit data symbols in a non-orthogonal manner without relying on the granting procedures.

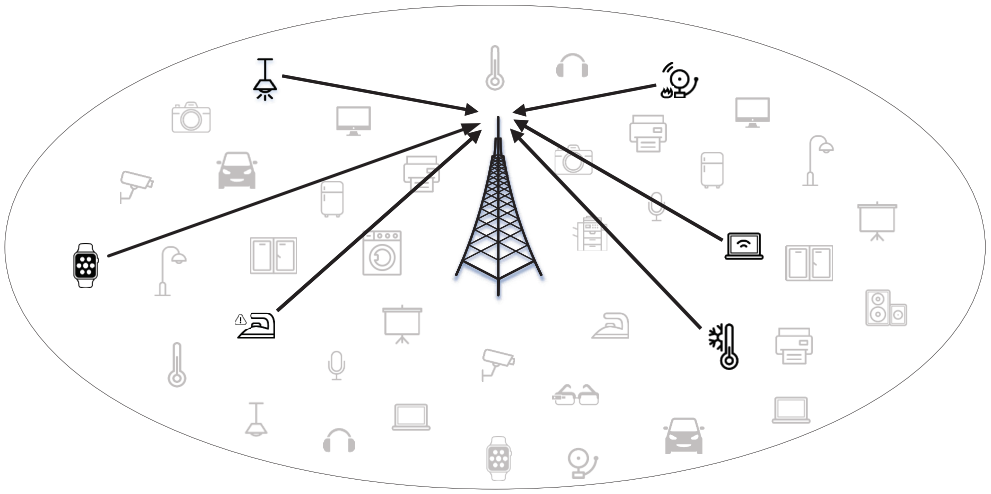


Figure 1.1: The sporadic uplink multiple access in a mMTC network.

In the mMTC network, a large portion of devices is inactive and hence does not transmit data. Thus, the transmit vector consisting of data symbols of both active and inactive devices can be readily modeled as a sparse vector. By capitalizing on the sparsity of this multi-user vector, the multi-user detection (MUD) problem can be formulated as a sparse signal recovery problem [8–11]. This type of detection scheme, called compressed sensing based multi-user detection (CS-MUD), has been a key ingredient in the grant-free uplink NOMA schemes. In this dissertation, two novel CS-MUD techniques are proposed.

1.1 Sparsity-Aware Ordered Successive Interference Cancellation

Recently, several approaches to cast the CS-MUD problem into a sparsity-aware maximum *a posteriori* probability (S-MAP) detection problem have been suggested. In a nutshell, the main goal of S-MAP detection is to perform a MAP detection of sparse multi-user symbol vectors from a zero-augmented finite alphabet. In [12], linear relaxed S-MAP detectors have been suggested. In [13, 14], sparsity-aware sphere decoding (SA-SD) has been proposed. In [15], as a variant of SA-SD, K-Best detector has been proposed.

With an aim to reduce the computational complexity, sparsity-aware successive interference cancellation (SA-SIC) has been studied in [16]. SA-SIC is a simple low-complexity scheme that recovers transmit symbols in a sequential manner. The difference between SA-SIC and the conventional successive interference cancellation (SIC) is that SA-SIC incorporates a sparsity constraint into the detection process. Although SA-SIC can find a sparse solution with very low computational effort, error propagation during the successive detection severely degrades the performance of SA-SIC. In order to minimize such error propagation and thus enhance the performance, the detection order should be sorted properly prior to the SA-SIC operation. To this end, sorted

QR decomposition (SQRD) and its variants ensuring that devices with a high channel gain are detected in the early layers have been proposed. It has been shown that SQRD determines the optimal detection order with high probability in the recovery of non-sparse multi-user vectors [17].

However, when multi-user vectors are sparse and each device is active with a distinct probability, the detection order determined solely by channel gains might not be optimal. In practice, owing to their heterogeneous traffic demands, each MTC device is active with a distinct probability [18]. In this situation, the detection ordering should incorporate the (heterogeneous) activity probabilities of devices.

An aim of this dissertation is to propose a novel sparsity-aware ordered SIC scheme for the recovery of sparse multi-user vectors in mMTC systems. Specifically, an activity-aware sorted QR decomposition (A-SQRD) algorithm that finds the optimal detection order to reduce the error propagation and enhance the performance of SA-SIC is presented. The proposed A-SQRD algorithm is distinct from existing methods in that the detection order is determined not only by the channel gains but also by the activity probabilities of devices. Numerical simulations show that the proposed technique achieves a significant enhancement in the detection performance over conventional schemes without significantly increasing computational complexity.

1.2 Expectation Propagation-based Joint Active User Detection and Channel Estimation

In several studies regarding CS-MUD, the uplink channel state information (CSI) from the MTD to the BS is assumed to be perfectly known to the BS. In practice, however, the uplink CSI from the devices to the BS should be estimated before data detection. To address this issue, various joint active user detection (AUD) and channel estimation (CE) schemes have been proposed. Since only a few devices are active at one time, an element-wise (i.e., Hadamard) product of the binary activity pattern and the channel

vector is also a sparse vector and thus compressed sensing (CS)-based technique is a good fit for the problem at hand [20–24].

One potential shortcoming in these studies is that a prior distribution of the sparse vector is not exploited. In fact, these studies are based on the non-Bayesian greedy algorithms such as the orthogonal matching pursuit (OMP) and approximate message passing (AMP) algorithms, which do not require a prior distribution of the sparse vector. In essence, these algorithms find out non-zero values based on the instantaneous correlation between the sensing matrix and the observation vector so that they might not be effective in the situation where the prior distribution is available. In this case, clearly, by exploiting the statistical distribution of the sparse vector, the performance of AUD and CE can be improved substantially.

An aim of this dissertation is to propose a novel Bayesian joint AUD and CE technique based on the expectation propagation (EP) algorithm [25–28]. The EP algorithm is a Bayesian technique to approximate a computationally intractable target probability distribution to the distribution from a tractable family. By iteratively minimizing the Kullback-Leibler divergence between the target distribution and the approximate distribution via moment matching, the EP algorithm can efficiently find a tractable approximation of the target distribution. In describing the prior distribution of activity and channel of each device, we employ the Bernoulli-Gaussian probabilistic model. The posterior distribution of user activities and channels is computationally intractable due to the discrete nature of the binary activity variables. In this work, we iteratively find the best approximation of the posterior distribution of the composite vector of user activities and channels using the EP algorithm. Using the obtained approximation, activity identification and CSI estimation of active devices are performed jointly. The data detection of active devices is then performed based on the obtained knowledge of user activities and channels. Numerical evaluations in realistic mMTC scenarios show that the proposed technique outperforms conventional non-Bayesian greedy algorithms and other Bayesian techniques. In particular, the proposed technique performs close to

the Oracle detector, an ideal detector having perfect knowledge on the user activities in the high signal-to-noise-ratio (SNR) regime.

Chapter 2

Sparsity-Aware Ordered Successive Interference Cancellation

2.1 System model

We consider the uplink of mMTC systems where N MTC devices access a single base station, as illustrated in Fig. 2.1. The symbol of each device is spread with a user-specific sequence with a length of M. Here, we assume that devices are synchronized in time, meaning that all devices switch activity and draw symbols in the same time slot. In this setup, the received signal at the base station can be described as

$$\mathbf{y} = \mathbf{H}\mathbf{x} + \mathbf{w}, \tag{2.1}$$

where $\mathbf{H}$ is a $M \times N$ complex-valued matrix capturing the spreading sequences and channel impulse responses between devices and the base station, $\mathbf{x}$ is the symbol vector of all (active and inactive) devices, and $\mathbf{w}$ is the complex Gaussian noise vector with the noise variance σ_w^2. The symbol x_n is drawn from an equi-probable finite modulation alphabet $\mathcal{A}$ when the n-th device is active, and zero otherwise. When $M < N$, the system is said to be under-determined. While it is in general not possible to recover the symbol vector in this scenario, theory of compressed sensing (CS) guarantees that $\mathbf{x}$ can be recovered accurately if $\mathbf{x}$ is a sparse vector.

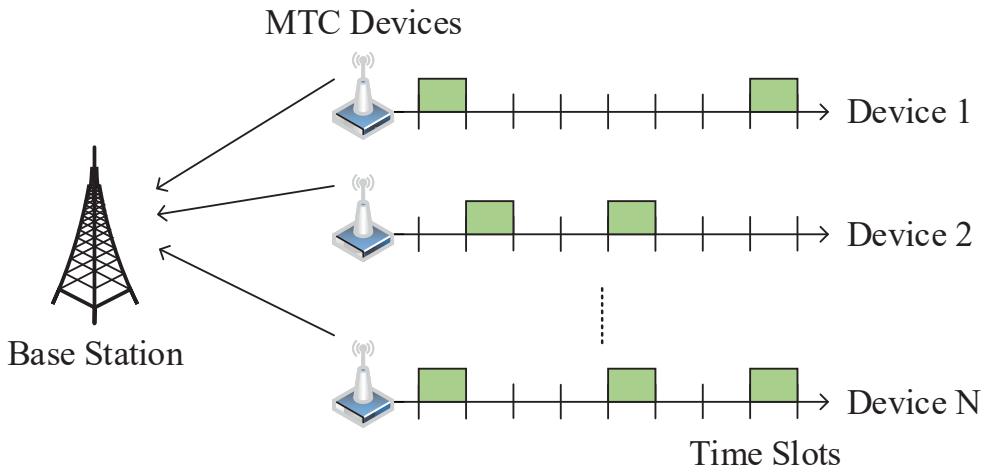


Figure 2.1: The mMTC uplink multiple access scenario with N devices sporadically transmitting data symbols to a base station.

2.2 Sparsity-Aware Successive Interference Cancellation (SA-SIC)

2.2.1 Derivation of S-MAP Detection

The output of the S-MAP detector maximizing the *a posteriori* probability $\Pr(\mathbf{x}|\mathbf{y})$ is given by Bayes' rule as

$$\begin{aligned}\hat{\mathbf{x}} &= \underset{\mathbf{x}\in\mathcal{A}_0^N}{\arg\max}\, \Pr(\mathbf{x}|\mathbf{y}) \\ &= \underset{\mathbf{x}\in\mathcal{A}_0^N}{\arg\min} - \ln \Pr(\mathbf{y}|\mathbf{x}) - \ln \Pr(\mathbf{x}),\end{aligned} \tag{2.2}$$

where $\mathcal{A}_0 = \mathcal{A} \cup \{0\}$ is the augmented modulation alphabet.

When the n-th device is active with the activity probability p_n and further p_n is very small for all n, $\mathbf{x}$ can be modeled as a sparse vector containing many zeros. Considering that the activity of each device is independent of each other, the prior distribution of $\mathbf{x}$ can then be described as

$$\Pr(\mathbf{x}) = \prod_{n=1}^{N} \Pr(x_n) = \prod_{n=1}^{N} (1-p_n)^{1-|x_n|_0} (p_n/|\mathcal{A}|)^{|x_n|_0}, \tag{2.3}$$

where $|x_n|_0$ is the element-wise l_0-norm that is equal to 1 if x_n is a non-zero value, otherwise it is zero.

From (2.2) and (2.3), we have

$$\hat{\mathbf{x}} = \underset{\mathbf{x}\in\mathcal{A}_0^N}{\arg\min} \|\mathbf{y} - \mathbf{H}\mathbf{x}\|_2^2 + \sigma_w^2 \sum_{n=1}^{N} \lambda_n |x_n|_0 , \tag{2.4}$$

where $\lambda_n = \ln[(1-p_n)/(p_n/|\mathcal{A}|)]$ is an element-wise regularization parameter.

The goal of the S-MAP detection is to find a vector in $\mathcal{A}_0^N$ that maximizes the cost function in (2.4). The optimization problem in (2.4) is in essence a regularized least squares minimization problem. The regularization term accounts for the heterogeneous activity probabilities of N MTC devices. Since λ_n is inversely proportional to p_n, the regularization term promotes the sparsity of $\mathbf{x}$.

2.2.2 Sparsity-Aware SIC (SA-SIC) Detection

In this section, we discuss the SA-SIC technique to solve the problem (2.4). SA-SIC uses the QR decomposition of matrix $\mathbf{H} = \mathbf{QR}$, where $\mathbf{Q}$ is an $M \times N$ unitary matrix and $\mathbf{R}$ is a $N \times N$ upper triangular matrix. Using QR decomposition, we have

$$\begin{aligned}
\hat{\mathbf{x}} &= \underset{\mathbf{x}\in\mathcal{A}_0^N}{\arg\min} \|\mathbf{y} - \mathbf{H}\mathbf{x}\|_2^2 + \sigma_w^2 \sum_{n=1}^{N} \lambda_n |x_n|_0 \\
&= \underset{\mathbf{x}\in\mathcal{A}_0^N}{\arg\min} \|\mathbf{y} - (\mathbf{QR})\mathbf{x}\|_2^2 + \sigma_w^2 \sum_{n=1}^{N} \lambda_n |x_n|_0 \\
&= \underset{\mathbf{x}\in\mathcal{A}_0^N}{\arg\min} \left\|\mathbf{Q}^H\mathbf{y} - \mathbf{Q}^H(\mathbf{QR})\mathbf{x}\right\|_2^2 + \sigma_w^2 \sum_{n=1}^{N} \lambda_n |x_n|_0 \\
&= \underset{\mathbf{x}\in\mathcal{A}_0^N}{\arg\min} \|\tilde{\mathbf{y}} - \mathbf{R}\mathbf{x}\|_2^2 + \sigma_w^2 \sum_{n=1}^{N} \lambda_n |x_n|_0 \\
&= \underset{\mathbf{x}\in\mathcal{A}_0^N}{\arg\min} \sum_{n=1}^{N} \left[\left| \tilde{y}_n - \sum_{l=n}^{N} R_{nl} x_l \right|^2 + \sigma_w^2 \lambda_n |x_n|_0 \right], \qquad (2.5)
\end{aligned}$$

where $\tilde{\mathbf{y}} = \mathbf{Q}^H\mathbf{y}$. Once the estimates of previous symbols $\{\hat{x}_l\}_{l=n+1}^{N}$ are available, the optimal $\hat{x}_n$ can be obtained by minimizing the n-th per-symbol cost function $d_n(x_n)$ over one scalar variable $x_n \in \mathcal{A}_0$ as

$$\hat{x}_n = \underset{x_n\in\mathcal{A}_0}{\arg\min} \underbrace{\left[\left| \tilde{y}_n - \sum_{l=n+1}^{N} R_{nl}\hat{x}_l - R_{nn}x_n \right|^2 + \sigma_w^2 \lambda_n |x_n|_0 \right]}_{\triangleq d_n(x_n)}. \qquad (2.6)$$

The detection process begins with the highest layer ($n = N$) and goes down to the lowest layer sequentially, $n = N-1, \cdots, 1$. In doing so, SA-SIC achieves an acceptable detection performance with much lower complexity compared to other optimal but complex S-MAP detectors. However, the main drawback of SA-SIC is that it is sensitive to the error propagation from the early layers. Hence, the selection of an appropriate detection order is crucial to mitigate the error propagation.

2.3 Proposed Activity-Aware Sorted-QRD (A-SQRD) Algorithm

In this section, we describe the proposed activity-aware sorted QR decomposition (A-SQRD) algorithm to enhance the detection performance of SA-SIC. The conventional SQRD algorithm sorts the columns of channel matrix $\mathbf{H}$ and find its sorted QR decomposition based on channel gains. Since the cost function of the S-MAP detection contains an additional sparsity-promoting regularization term, the regularization parameters λ_n as well as the noise variance σ_w^2 and channel matrix $\mathbf{H}$ needs to be considered in the detection ordering. To this end, we construct an augmented system and then sort the columns of the augmented system matrix such that devices with high post-detection signal-to-noise-ratio (SNR) are detected first.

In this work, we restrict ourselves to the constant modulus alphabet (i.e., $\|\mathbf{x}\|_0 = \|\mathbf{x}\|_2^2 = \|\mathbf{x}\|_p^p, p \geq 1$). Noting that the data rate of MTC devices is in general low (typically in orders of tens of kilobits per seconds), it is reasonable to assume that data symbols are modulated with constant modulus alphabets (e.g., phase shift keying constellations). Replacing the l_0-norm with the l_2-norm in (2.4), we have

$$\begin{aligned}
\hat{\mathbf{x}} &= \underset{\mathbf{x}\in\mathcal{A}_0^N}{\arg\min} \|\mathbf{y}-\mathbf{H}\mathbf{x}\|_2^2 + \sigma_w^2 \sum_{n=1}^{N} \lambda_n |x_n|^2 \\
&= \underset{\mathbf{x}\in\mathcal{A}_0^N}{\arg\min} \|\mathbf{y}-\mathbf{H}\mathbf{x}\|_2^2 + \left\|\sigma_w \mathrm{diag}(\sqrt{\boldsymbol{\lambda}})\mathbf{x}\right\|_2^2 \\
&= \underset{\mathbf{x}\in\mathcal{A}_0^N}{\arg\min} \left\| \begin{bmatrix} \mathbf{y} \\ \mathbf{0_N} \end{bmatrix} - \begin{bmatrix} \mathbf{H} \\ \sigma_w \mathrm{diag}(\sqrt{\boldsymbol{\lambda}}) \end{bmatrix} \mathbf{x} \right\|_2^2 \\
&= \underset{\mathbf{x}\in\mathcal{A}_0^N}{\arg\min} \left\|\mathbf{y}_0 - \mathbf{H}'\mathbf{x}\right\|_2^2, \qquad (2.7)
\end{aligned}$$

where $\mathbf{y}_0 \in \mathbb{C}^{M+N}$ is the zero-augmented observations, $\mathbf{H}' \in \mathbb{C}^{(M+N)\times N}$ is an augmented system matrix and $\boldsymbol{\lambda}$ is the regularization parameters of N devices.

The objective of the proposed A-SQRD algorithm is to find the optimal permutation of columns of $\mathbf{H}'$ and the corresponding QR decomposition that maximizes

the detection SNRs in the early layers. Here, we find the QR decomposition of the augmented channel matrix $\mathbf{H}'$ as $\mathbf{H}'\mathbf{P} = \mathbf{QR}$, where $\mathbf{P}$ is the binary permutation matrix. To find the optimal permutation, we employ the modified Gram-Schmidt algorithm [19] and extend it to reorder the columns of $\mathbf{H}'$ before each orthogonalization step.

Detailed steps of the proposed algorithm are as follows: First, we set $\mathbf{Q} = \mathbf{H}'$ and $\mathbf{R} = \mathbf{0}_{N\times N}$, respectively. In the next step, for each n-th iteration, the column index with the smallest l_2-norm of $\mathbf{q}_j$ ($j = n+1, \cdots, N$) is determined. Then the n-th column of each of $\mathbf{Q}$, $\mathbf{R}$ and $\mathbf{P}$ is exchanged with the one corresponding to the smallest norm. After the columns are exchanged, R_{nn} is set to $\|\mathbf{q}_n\|$ and $\mathbf{q}_n$ is normalized. In this way, the diagonal elements of $\mathbf{R}$ are ordered such that $R_{ii} < R_{nn}$ for $i < n$, and thus the post-detection SNR is higher in the early layers.

After the columns are sorted, we perform the orthogonalization of the columns such that $\mathbf{q}_{n+1}, \cdots, \mathbf{q}_N$ are orthogonalized with respect to $\mathbf{q}_n$ (i.e., $\mathbf{q}_j \perp \mathbf{q}_n$, $j = n+1, \cdots, N$). R_{nj} is computed as $R_{nj} = \mathbf{q}_n^H \mathbf{q}_j$ and $\mathbf{q}_j$ is calculated as

$$\mathbf{q}_j = \mathbf{q}_j - R_{nj}\mathbf{q}_n. \tag{2.8}$$

Using the sorted QR decomposition $\mathbf{H}'\mathbf{P} = \mathbf{QR}$ found by the A-SQRD algorithm, (2.7) can be rewritten as

$$\begin{aligned}\hat{\mathbf{x}} &= \underset{\mathbf{x}\in\mathcal{A}_0^N}{\arg\min} \left\|\mathbf{y}_0 - \mathbf{QRP}^H\mathbf{x}\right\|_2^2 \\ &= \underset{\mathbf{x}\in\mathcal{A}_0^N}{\arg\min} \left\|\tilde{\mathbf{y}}_0 - \mathbf{RP}^H\mathbf{x}\right\|_2^2,\end{aligned} \tag{2.9}$$

where $\tilde{\mathbf{y}}_0 = \mathbf{Q}^H\mathbf{y}_0$. Similarly to (2.5), we can decompose (2.9) into the sum of N per-symbol cost functions. Then, for $n = N, \cdots, 1$, the optimal solution of $\hat{x}_n$ can be found in closed form as

$$\hat{x}_n = Q_{\mathcal{A}_0}(x_n^{\text{ls}}), \tag{2.10}$$

where $x_n^{\text{ls}} = (\tilde{y}_{0,n} - \sum_{l=n+1}^{N} R_{nl}\hat{x}_l)/R_{nn}$ is the least squares solution to the n-th per-symbol cost, and $Q_{\mathcal{A}_0}(\cdot)$ is the quantization operator that maps the input to the closest

point in $\mathcal{A}_0$. Finally, by right-multiplying $\mathbf{P}^H$ by $\hat{\mathbf{x}}$ (de-sorting), the final solution is obtained. The proposed algorithm is summarized in Algorithm 2.1.

Table 2.1: SUMMARY OF THE PROPOSED SA-OSIC ALGORITHM

Input: $\mathbf{y}, \mathbf{H}, \mathcal{A}_0, \sigma_w^2, \{p_n\}_{n=1}^N$

Output: $\hat{\mathbf{x}}$

% Initialization

$\boldsymbol{\lambda}(n) \leftarrow \ln[(1-p_n)/(p_n/|\mathcal{A}|)]$

$\mathbf{y}_0 \leftarrow [\mathbf{y}; \text{zeros}(N,1)]$, $\mathbf{Q} \leftarrow [\mathbf{H}; \sigma_w \text{diag}(\sqrt{\boldsymbol{\lambda}})]$,

$\mathbf{R} \leftarrow \mathbf{0}_{N\times N}$, $\mathbf{P} \leftarrow \mathbf{I}_N$

for $n = 1, \cdots, N$ **do**

% Column Sorting

$n_{\text{min}} \leftarrow \arg\min_{j=n,\cdots,N} \|\mathbf{q}_j\|^2$

exchange columns n and n_{min} in $\mathbf{Q}, \mathbf{R}$, and $\mathbf{P}$

$R_{nn} \leftarrow \|\mathbf{q}_n\|$

$\mathbf{q}_n \leftarrow \mathbf{q}_n / R_{nn}$

% Column Orthogonalization

for $j = n+1, \cdots, N$ **do**

$R_{nj} \leftarrow \mathbf{q}_n^H \mathbf{q}_j$

$\mathbf{q}_j \leftarrow \mathbf{q}_j - R_{nj}\mathbf{q}_n$

end for

end for

% SIC Operation

$\tilde{\mathbf{y}}_0 \leftarrow \mathbf{Q}^H \mathbf{y}_0$

for $n = N, \cdots, 1$ **do**

$x_n^{\text{ls}} \leftarrow (\tilde{y}_{0,n} - \sum_{l=n+1}^N R_{nl}\hat{x}_l)/R_{nn}$

$\hat{x}_n \leftarrow Q_{\mathcal{A}_0}(x_n^{\text{ls}})$

end for

$\hat{\mathbf{x}} \leftarrow \hat{\mathbf{x}}\mathbf{P}^H$

2.4 Complexity Analysis

The computational complexity of the A-SQRD and SQRD algorithm is analyzed below by counting each required numerical operation as one complex floating point operation.

$$\mathcal{C}_{\mathrm{A-SQRD}} = 2N^3 + (2M+2)N^2 + (M-1)N, \tag{2.11}$$

$$\mathcal{C}_{\mathrm{SQRD}} = (2M+1)N^2 + (M-1)N. \tag{2.12}$$

Since the A-SQRD algorithm performs the QR decomposition of the augmented channel matrix, it causes a slight computational order increase compared to the conventional SQRD algorithm.

However, the computational burden of A-SQRD algorithm is less than that of K-Best detector and SA-SD. The K-Best detector performs a breadth-first search to find the best K paths minimizing the sum of per-symbol cost functions. The computational complexity of K-Best detector is

$$\mathcal{C}_{\mathrm{K-Best}} = K|\mathcal{A}_0|(\frac{N^3}{3} + 2N^2 + \frac{5}{3}N + \log^2(K|\mathcal{A}_0|)), \tag{2.13}$$

which scales with K. Increasing K will improve the performance of K-Best detector so that it performs close to the SA-SD. However, it causes an infeasible complexity compared to the A-SQRD algorithm.

2.5 Numerical Results

In this section, we describe the numerical experiment that demonstrates the effectiveness of the proposed algorithm. We compare the proposed A-SQRD algorithm with the linear minimum mean squared error (LMMSE) estimator[1], orthogonal matching pursuit (OMP) algorithm, least absolute shrinkage and selection operator (LASSO) detector, unsorted SA-SIC, conventional SQRD algorithm, data-dependent sorting and

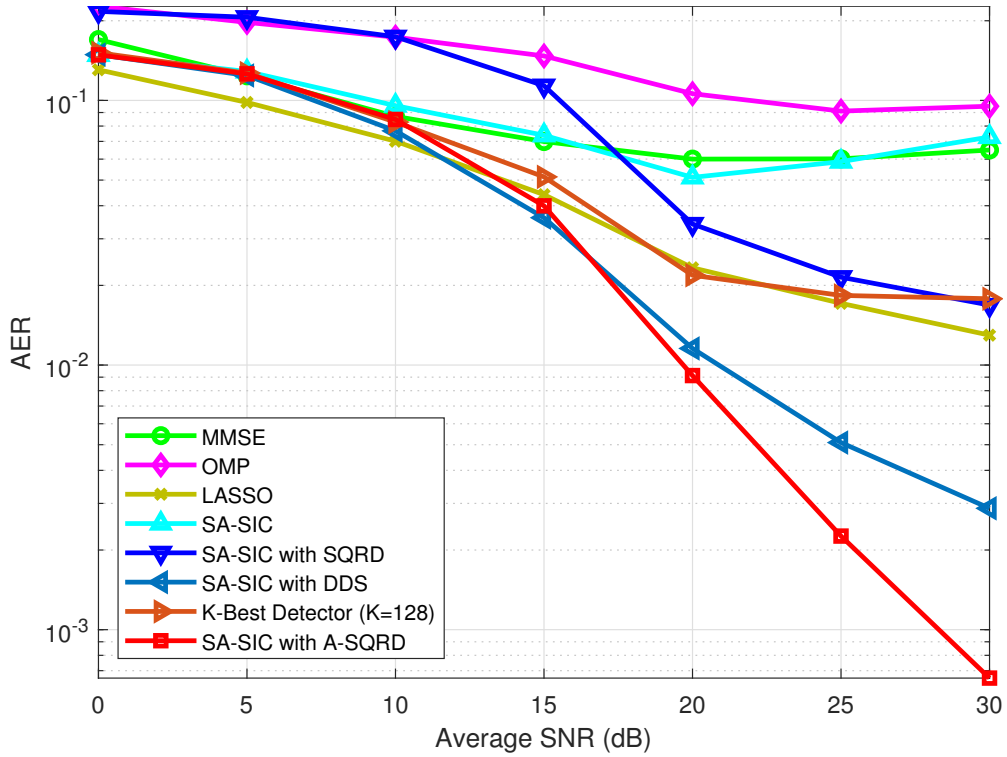
[1]We use the quantized version of the output of the LMMSE estimator.

regularization (DDS) algorithm, and K-Best detector. DDS algorithm is a heuristic approach that rescales λ_n based on the correlation between $\mathbf{y}$ and $\mathbf{H}$ to enhance the SQRD algorithm [16]. As a lower bound of the detection algorithms, we also test the Oracle LMMSE detector. Note that the Oracle detector has the support[2] (index set of nonzero entries) information of $\mathbf{x}$ and thus solves an over-determined system.

2.5.1 Simulation Setup

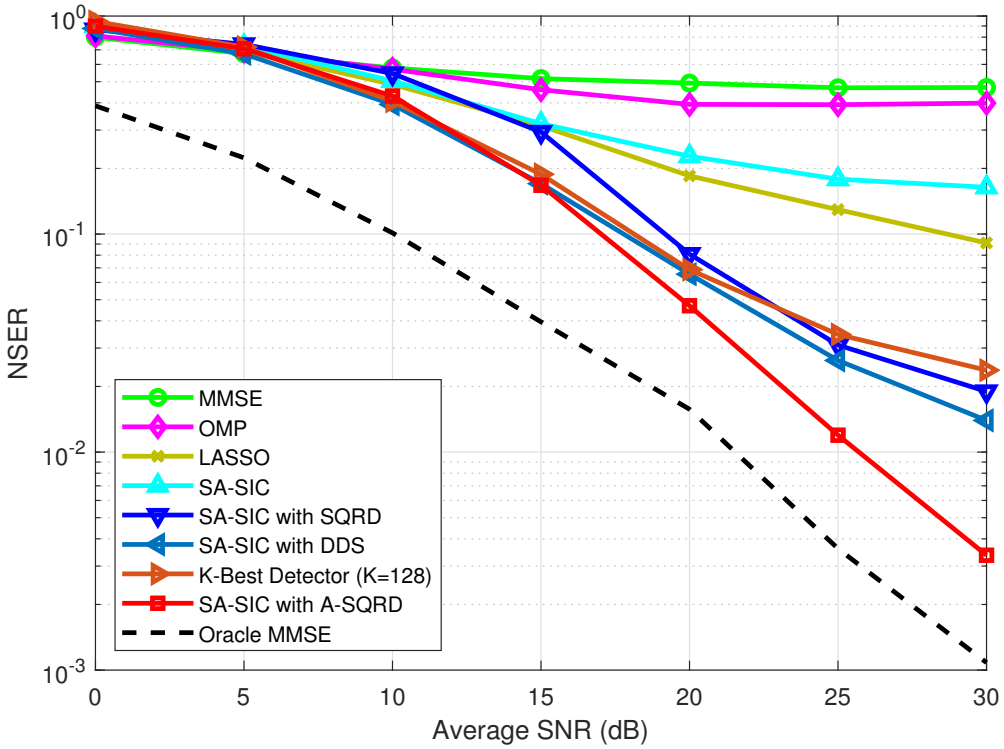
We simulate an under-determined mMTC system with N = 128 MTC devices and unit-norm random sequences with a length of M = 64 for spreading. We consider non-dispersive independent Rayleigh fading channels between devices and a base station from a complex Gaussian distribution of $\mathcal{CN}(0, 1)$. Thus, the average SNR is set to $1/\sigma_w^2$. We assume that the base station has perfect knowledge of the matrix $\mathbf{H}$. Data symbols of active devices are modulated with quadrature phase shift keying (QPSK). We consider the net symbol error rate (NSER) as a detection performance measure. NSER refers to the symbol error rate of active devices.

[2]For example, if $\mathbf{x} = [-1\ 0\ 1\ 0\ 0\ 0]^T$, then the support T is $T = \{1, 3\}$.



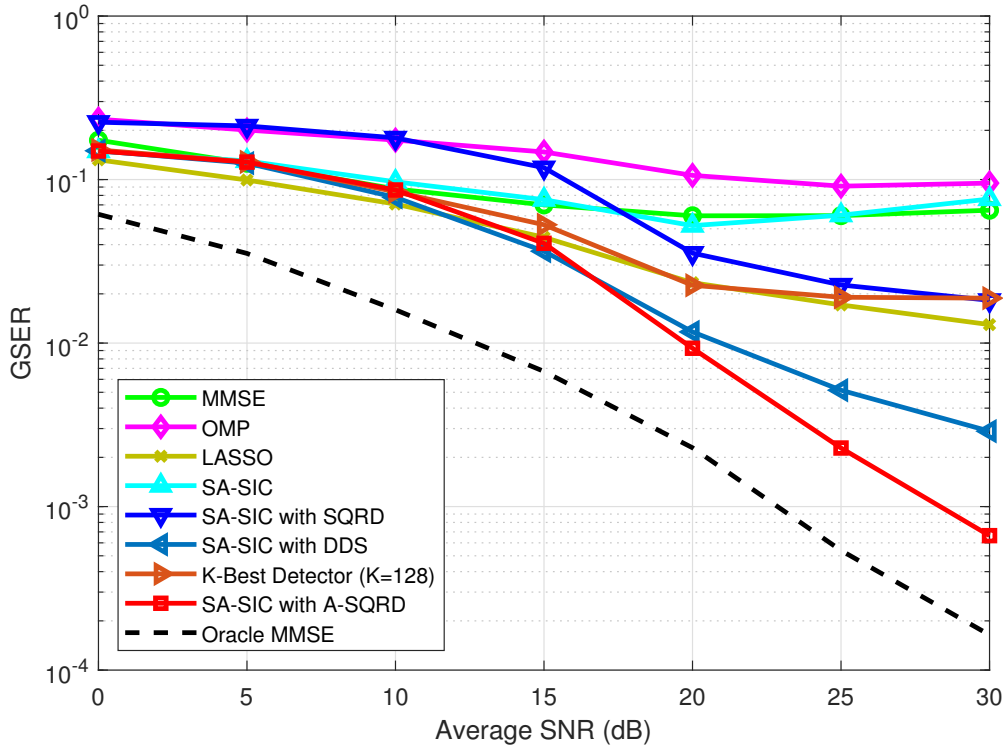
(a)

Figure 2.2: (a) AER as a function of the average SNR.



(b)

Figure 2.2: (b) NSER as a function of the average SNR.



(c)

Figure 2.2: (c) GSER as a function of the average SNR.

2.5.2 Simulation Results

Fig. 2.2 shows the AER, NSER, and GSER performance for each algorithm as a function of the average SNR. The activity probabilities $\{p_n\}_{n=1}^{N}$ are drawn uniformly at random in $[0.01, 0.3]$. We observe that SA-SIC with the proposed A-SQRD algorithm outperforms the conventional algorithms. LMMSE exhibits poor performance since the system is under-determined. Due to error propagation, the unsorted SA-SIC does not perform well. Since the SQRD and DDS algorithm do not consider the heterogeneous activity probabilities, the performance gains over the unsorted SA-SIC are marginal. In contrast, the A-SQRD algorithm is effective since it considers both the activity probabilities and channel gains to find the best detection order. The K-Best detector performs worse than the proposed algorithm with $K = 128$ and 256. In Table 2.2, we provide the number of required numerical operations of detection algorithms. When compared to the A-SQRD algorithm, the K-Best detector requires higher computational complexity to achieve the performance comparable to the A-SQRD algorithm.

Table 2.2: COMPUTATIONAL COMPLEXITY OF DETECTION ALGORITHMS

SQRD	**DDS**	**A-SQRD**	**K-Best** ($K = 128$)
2.12×10^6	2.14×10^6	6.33×10^6	4.68×10^8

Chapter 3

Expectation Propagation-based Joint Active User Detection and Channel Estimation

3.1 System model

We consider the uplink of a mMTC network where N MTDs access a single BS, as shown in Fig. 3.1. We assume that the BS and each device are equipped with one antenna. Each active device transmits a pilot symbol followed by J data symbols (which we call a frame in the sequel). In this work, we assume that devices are synchronized in time, meaning that all devices switch their activity in the same time slot.[1] Each device is either active or inactive in the whole interval of the frame. Also, we assume a flat fading channel model where channel remains unchanged in the entire frame. We denote the complex uplink channel coefficient from the n-th device to the BS by h_n, where h_n follows the zero-mean complex-Gaussian non-dispersive independent Rayleigh fading with the variance α_n, i.e., $h_n \sim \mathcal{CN}(h_n|0, \alpha_n)$. The variance α_n captures the pathloss component characterized by the each device's location. We assume that α_n is known at the BS. In the scenarios where devices are stationary, the pathloss can be estimated and stored at the BS as a prior information.

[1] Since the packet size of MTD is typically very small ($10 \sim 100$ bytes) in mMTC environment, the impact of this assumption on the proposed scheme would be marginal.

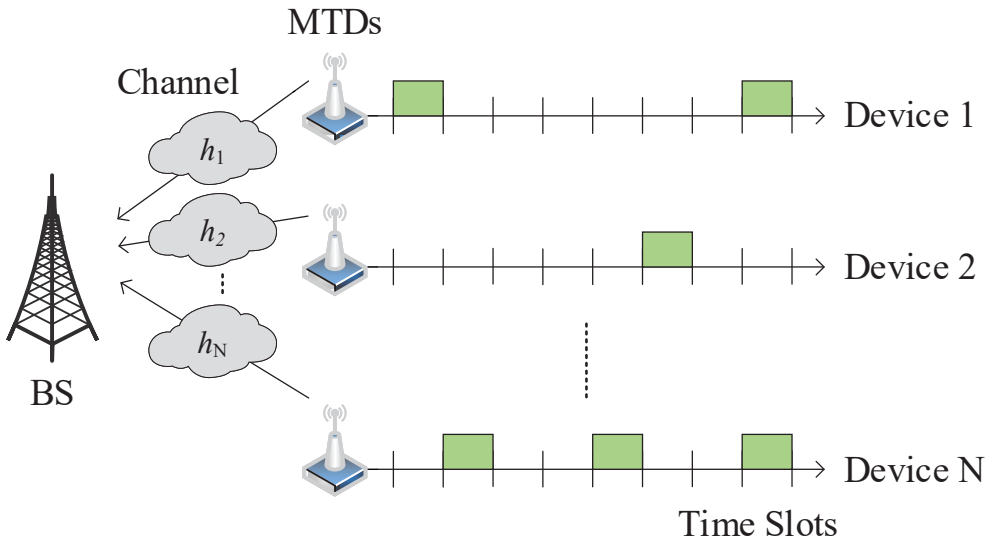


Figure 3.1: The mMTC uplink multiple access scenario with N devices sporadically becoming active.

In order to model the sporadic traffic pattern of the mMTC network, we define the binary user activity indicator a_n for the n-th device as

$$a_n = \begin{cases} 1, & \text{if the } n\text{-th device is active,} \\ 0, & \text{otherwise.} \end{cases} \tag{3.1}$$

The n-th device is active with the activity probability p_n and the activity of each device is independent of each other.

In this setup, the input-output relationship can be described as

$$\mathbf{y} = \sum_{n=1}^{N} \mathbf{s}_n h_n a_n x_n + \mathbf{w}, \tag{3.2}$$

where x_n is the transmit symbol of the n-th device, $\mathbf{s}_n \in \mathbb{C}^M$ is the spreading sequence for the n-th device, $\mathbf{y} \in \mathbb{C}^M$ is the measurement vector at the BS, and $\mathbf{w}$ is the independent zero-mean complex-Gaussian noise vector with the variance σ_w^2. We assume that the transmit powers of all devices are the same, i.e., $E[|x_n|^2] = \rho$.

In the general mMTC scenarios, the number of devices N is larger than the number of resources used for the transmission M (i.e, $M < N$). While it is in general not possible to recover the target vector in this underdetermined scenario, the theory of compressed sensing (CS) guarantees that the target vector can be recovered accurately as long as the vector is sparse and the measurement process preserves the energy of an input vector [9].

3.2 Joint Active User Detection and Channel Estimation

In this work, we adopt a grant-free multiple access protocol consisting of two operational phases in each time slot (see Fig. 3.2). In the first phase, each active device transmits the pilot symbol to the BS and the BS jointly detects the user activities and then estimate the channels of active devices. In the second phase, the active devices transmit J data symbols to the BS and the BS decodes the data symbols using the obtained user activities and channels.

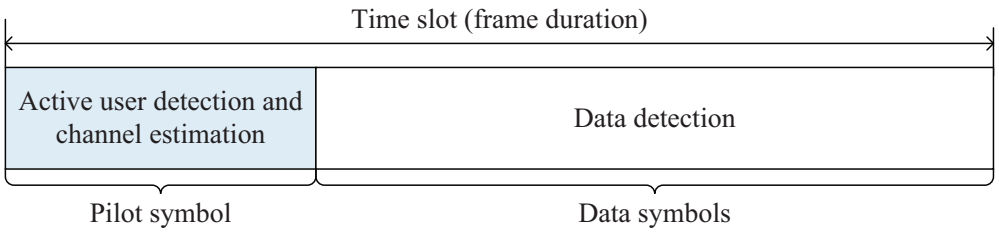


Figure 3.2: Two-phase grant-free multiple access protocol.

In this section, we describe the joint active user detection (AUD) and channel estimation (CE). Each active device transmits the pilot symbol $x_{p,n}$ to the BS before the data transmission. The pilot measurement vector $\mathbf{y}_p$ is given by

$$\mathbf{y}_p = \sum_{n=1}^{N} \mathbf{s}_n h_n a_n x_{p,n} + \mathbf{w}_p. \tag{3.3}$$

Let $\boldsymbol{\phi}_n = \mathbf{s}_n x_{p,n}$ and $\boldsymbol{\Phi} = [\boldsymbol{\phi}_1, \ldots, \boldsymbol{\phi}_N]$, then we have

$$\begin{aligned} \mathbf{y}_p &= \sum_{n=1}^{N} \boldsymbol{\phi}_n a_n h_n + \mathbf{w}_p, \\ &= \boldsymbol{\Phi}(\mathbf{a} \circ \mathbf{h}) + \mathbf{w}_p, \end{aligned} \tag{3.4}$$

where $\mathbf{a} = [a_1, \ldots, a_N]^T$ is the activity vector, $\mathbf{h} = [h_1, \ldots, h_N]^T$ is the channel vector, and $\circ$ is the Hadamard (element-wise) product operator.

Further by denoting the composite of activity vector $\mathbf{a}$ and channel vector $\mathbf{h}$ as $\mathbf{g} = \mathbf{a} \circ \mathbf{h} = [a_1 h_1, \cdots, a_N h_N]^T$, we have

$$\mathbf{y}_p = \boldsymbol{\Phi}\mathbf{g} + \mathbf{w}_p. \tag{3.5}$$

The output of the MAP estimator maximizing the *a posteriori* probability $p(\mathbf{g}|\mathbf{y}_p)$ is given by

$$\begin{aligned} \hat{\mathbf{g}} &= \arg\max_{\mathbf{g} \in \mathbb{C}^N} p(\mathbf{g}|\mathbf{y}_p) \\ &= \arg\max_{\mathbf{g} \in \mathbb{C}^N} p(\mathbf{y}_p|\mathbf{g}) p(\mathbf{g}), \end{aligned} \tag{3.6}$$

where $p(\mathbf{y}_p|\mathbf{g}) = \mathcal{CN}(\mathbf{y}_p|\boldsymbol{\Phi}\mathbf{g}, \sigma_w^2 \mathbf{I})$ is the likelihood function of $\mathbf{y}_p$ given $\mathbf{g}$.

When the n-th device is active ($a_n = 1$) with an activity probability p_n and further p_n has a very small value for all n, $\mathbf{g}$ can be readily modeled as a sparse vector. Considering that the activities of devices are independent of each other, the prior distribution of $\mathbf{g}$ can be expressed as

$$p(\mathbf{g}) = \prod_{n=1}^{N} \left[(1 - p_n)\delta(g_n) + p_n \mathcal{CN}(g_n|0, \alpha_n) \right], \tag{3.7}$$

where $\delta(\cdot)$ is the Dirac delta function.

From (3.6) and (3.7), we have

$$\begin{aligned}\hat{\mathbf{g}} &= \arg\max_{\mathbf{g}\in\mathbb{C}^N} p(\mathbf{y}_p|\mathbf{g})p(\mathbf{g}) \\ &= \arg\max_{\mathbf{g}\in\mathbb{C}^N} \mathcal{CN}(\mathbf{y}_p|\mathbf{\Phi}\mathbf{g}, \sigma_w^2\mathbf{I}) \prod_{n=1}^{N} \left[(1-p_n)\delta(g_n) + p_n\mathcal{CN}(g_n|0,\alpha_n)\right]. \end{aligned} \tag{3.8}$$

The goal of the MAP estimator is to find out a vector maximizing the cost function in (3.8). Unfortunately, when the number of devices N is very large, the optimization problem in (3.8) is computationally intractable due to the discrete nature of the binary activity vector $\mathbf{a}$. In this situation, it is desirable to construct a tractable approximation of the target posterior distribution to solve the MAP problem, i.e., $q(\mathbf{g}) \approx p(\mathbf{y}_p|\mathbf{g})p(\mathbf{g})$. In fact, the main idea of the proposed technique is to construct a multivariate Gaussian approximation of $f(\mathbf{g}) = p(\mathbf{y}_p|\mathbf{g})p(\mathbf{g})$ and then find out the mean and variance matching to $f(\mathbf{g})$ using the expectation propagation (EP) algorithm [25–28]. It has been shown that with only a few numbers of iterations, the EP algorithm can achieve an accurate approximate distribution with high probability [25, 26].

3.3 EP-Based Active User Detection and Channel Estimation

In this section, we describe the proposed EP-based joint AUD and CE method. First, we approximate the target distribution $f(\mathbf{g}) = p(\mathbf{y}_p|\mathbf{g})p(\mathbf{g})$ in (3.8) to the Gaussian distribution $q(\mathbf{g}) = \mathcal{CN}(\mathbf{g}|\tilde{\mathbf{m}}, \tilde{\mathbf{V}})$. Then, we match the mean vector $\tilde{\mathbf{m}}$ and covariance matrix $\tilde{\mathbf{V}}$ to those of the true target distribution $f(\mathbf{g})$ based on the iterative EP algorithm.

After the convergence, the approximate mean vector $\tilde{\mathbf{m}}$ becomes a reliable solution $\hat{\mathbf{g}}$ to the MAP estimation problem in (3.8) and its covariance matrix is $\tilde{\mathbf{V}}$. Then, by performing the log-likelihood test on $\hat{\mathbf{g}}$, we detect the active devices and then estimate

the CSI of the active devices. Finally, using the obtained knowledge of user activities and channels, data symbols of active devices are detected. Fig. 3.3 depicts the block diagram of the proposed technique.

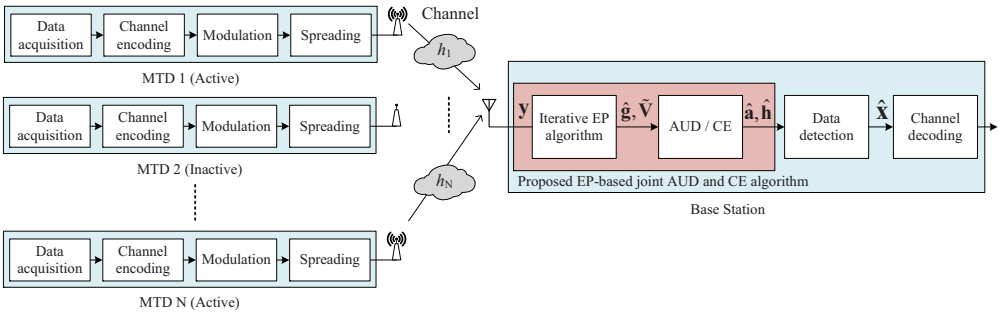


Figure 3.3: Block diagram of the proposed EP-based joint AUD and CE algorithm.

3.3.1 A Brief Review of Expectation Propagation

In this subsection, we briefly review the EP algorithm. The EP algorithm is an iterative algorithm that approximates the target probability distribution with a distribution from the exponential family $\mathcal{F}$. Suppose we have a target distribution $f(\mathbf{x})$ that can be factorized as

$$f(\mathbf{x}) = \prod_{i=1}^{I} f_i(\mathbf{x}). \tag{3.9}$$

Using the EP algorithm, we can construct a tractable approximation of $f(\mathbf{x})$ with a distribution $q(\mathbf{x}) = \prod_{i=1}^{I} q_i(\mathbf{x})$, where $q_i(\mathbf{x}) \in \mathcal{F}$.

To find the distribution $q(\mathbf{x})$ close to $f(\mathbf{x})$ from the exponential family $\mathcal{F}$, we use the Kullback-Leibler (KL) divergence criterion given by [29]

$$q(\mathbf{x}) = \underset{q'(\mathbf{x})\in\mathcal{F}}{\arg\min}\, \mathrm{D}_{\mathrm{KL}}\left[f(\mathbf{x})\|q'(\mathbf{x})\right]. \tag{3.10}$$

Because $q(\mathbf{x})$ belongs to the exponential family, the unique solution of the problem (3.10) is obtained by matching the expected sufficient statistics (moments) of $f(\mathbf{x})$ and $q(\mathbf{x})$ [30]. That is, parameters of $q(\mathbf{x})$ are chosen such that

$$E_{f(\mathbf{x})}[\mathbf{x}] = E_{q(\mathbf{x})}[\mathbf{x}], \tag{3.11}$$

$$\mathrm{Var}_{f(\mathbf{x})}[\mathbf{x}] = \mathrm{Var}_{q(\mathbf{x})}[\mathbf{x}]. \tag{3.12}$$

In the EP algorithm, the parameters of $q(\mathbf{x})$ satisfying (3.11) and (3.12) are obtained iteratively. In each iteration of the EP algorithm, $q_i(\mathbf{x})$ in $q(\mathbf{x})$ is replaced by $f_i(\mathbf{x})$. In other words, we first remove the contribution of $q_i(\mathbf{x})$ from $q(\mathbf{x})$:

$$q_{\setminus i}(\mathbf{x}) = \frac{q(\mathbf{x})}{q_i(\mathbf{x})}, \tag{3.13}$$

Then, we multiply $q_{\setminus i}(\mathbf{x})$ and $f_i(\mathbf{x})$ as

$$\hat{q}_i(\mathbf{x}) = f_i(\mathbf{x})q_{\setminus i}(\mathbf{x}) = f_i(\mathbf{x})\frac{q(\mathbf{x})}{q_i(\mathbf{x})}. \tag{3.14}$$

for all i. Finally, we update the parameters of the approximation factor $q_i(\mathbf{x})$ such that the moments of $q(\mathbf{x})$ and $\hat{q}_i(\mathbf{x})$ are matched as

$$E_{q(\mathbf{x})}[\mathbf{x}] = E_{\hat{q}_i(\mathbf{x})}[\mathbf{x}], \tag{3.15}$$

$$\mathrm{Var}_{q(\mathbf{x})}[\mathbf{x}] = \mathrm{Var}_{\hat{q}_i(\mathbf{x})}[\mathbf{x}], \tag{3.16}$$

for all i. The sequential EP algorithm is performed until a convergence criterion is satisfied or the maximum number of iterations is reached.

In this paper, we approximate the target posterior distribution $f(\mathbf{g}) = p(\mathbf{y}_p|\mathbf{g})p(\mathbf{g})$ by a Gaussian distribution $q(\mathbf{g}) = \mathcal{CN}(\mathbf{g}|\tilde{\mathbf{m}}, \tilde{\mathbf{V}})$ optimized by the EP algorithm. Then, based on the approximate distribution, AUD and CE are jointly performed and then the data is detected.

3.3.2 Form of the Approximation

In this subsection, we describe an approximate form of the target posterior distribution $f(\mathbf{g}) = f_1(\mathbf{g})f_2(\mathbf{g}) = p(\mathbf{y}_p|\mathbf{g})p(\mathbf{g})$. First, we approximate each term in $f(\mathbf{g})$ by a simple complex-Gaussian as

$$f_1(\mathbf{g}) = p(\mathbf{y}_p|\mathbf{g}) \approx q_1(\mathbf{g}) = \mathcal{CN}(\mathbf{g}|\tilde{\mathbf{m}}_1, \tilde{\mathbf{V}}_1), \tag{3.17}$$

$$f_2(\mathbf{g}) = p(\mathbf{g}) \approx q_2(\mathbf{g}) = \mathcal{CN}(\mathbf{g}|\tilde{\mathbf{m}}_2, \tilde{\mathbf{V}}_2). \tag{3.18}$$

Then, we construct an unnormalized global Gaussian approximation as

$$f(\mathbf{g}) \approx q(\mathbf{g}) = q_1(\mathbf{g})q_2(\mathbf{g}) = \mathcal{CN}(\mathbf{g}|\tilde{\mathbf{m}}, \tilde{\mathbf{V}}), \tag{3.19}$$

where the mean vector $\tilde{\mathbf{m}}$ and the covariance matrix $\tilde{\mathbf{V}}$ are given by

$$\tilde{\mathbf{V}} = \left(\tilde{\mathbf{V}}_1^{-1} + \tilde{\mathbf{V}}_2^{-1}\right)^{-1}, \tag{3.20}$$

$$\tilde{\mathbf{m}} = \tilde{\mathbf{V}}\left(\tilde{\mathbf{V}}_1^{-1}\tilde{\mathbf{m}}_1 + \tilde{\mathbf{V}}_2^{-1}\tilde{\mathbf{m}}_2\right). \tag{3.21}$$

The first approximation term $q_1(\mathbf{g})$ corresponds to the Gaussian noise likelihood $p(\mathbf{y}_p|\mathbf{g}) = \mathcal{CN}(\mathbf{y}_p|\mathbf{\Phi g}, \sigma_w^2\mathbf{I})$. In this case, $\tilde{\mathbf{m}}_1$ and $\tilde{\mathbf{V}}_1$ can be simply characterized by

the following relations

$$\tilde{\mathbf{m}}_1 = (\mathbf{\Phi}^H \mathbf{\Phi})^{-1} \mathbf{\Phi}^H \mathbf{y}_p, \tag{3.22}$$

$$\tilde{\mathbf{V}}_1 = \sigma_w^2 (\mathbf{\Phi}^H \mathbf{\Phi})^{-1}. \tag{3.23}$$

Using (3.22) and (3.23), the global approximation in (3.19) is rewritten as

$$q(\mathbf{g}) = q_1(\mathbf{g}) q_2(\mathbf{g}) = \mathcal{CN}(\mathbf{g}|\tilde{\mathbf{m}}, \tilde{\mathbf{V}}). \tag{3.24}$$

where

$$\tilde{\mathbf{V}} = \left(\sigma_w^{-2} \mathbf{\Phi}^H \mathbf{\Phi} + \tilde{\mathbf{V}}_2^{-1} \right)^{-1}, \tag{3.25}$$

$$\tilde{\mathbf{m}} = \tilde{\mathbf{V}} \left(\sigma_w^{-2} \mathbf{\Phi}^H \mathbf{y}_p + \tilde{\mathbf{V}}_2^{-1} \tilde{\mathbf{m}}_2 \right). \tag{3.26}$$

3.3.3 Iterative EP Update Rules

In this subsection, we explain the iterative EP update rules for the parameter estimations $\tilde{\mathbf{m}}_2 = [\tilde{m}_{2,1}, \ldots, \tilde{m}_{2,N}]^T$ and $\tilde{\mathbf{V}}_2 = \mathrm{diag}(\tilde{v}_{2,1}, \ldots, \tilde{v}_{2,N})$. In the first iteration, we initialize $\tilde{\mathbf{m}}_2 = \mathbf{0}$ and $\tilde{\mathbf{V}}_2 = \mathrm{diag}(p_1\alpha_1, \ldots, p_N\alpha_N)$. In the general, say l-th, iteration, the pairs $(\tilde{m}_{2,n}^{(l+1)}, \tilde{v}_{2,n}^{(l+1)})$ for all $n = 1, \ldots, N$ are updated as follows (see Fig. 3.4).

First, we compute the ratio distribution $q_{\backslash 2,n}^{(l)}(g_n)$ by removing the contribution of $q_{2,n}^{(l)}(g_n)$ from $q_n^{(l)}(g_n) = \mathcal{CN}(g_n|\tilde{m}_n^{(l)}, \tilde{V}_{nn}^{(l)})$, which corresponds to the n-th marginal of $q^{(l)}(\mathbf{g})$. That is,

$$q_{\backslash 2,n}^{(l)}(g_n) = \frac{q_n^{(l)}(g_n)}{q_{2,n}^{(l)}(g_n)} = \frac{\mathcal{CN}(g_n|\tilde{m}_n^{(l)}, \tilde{V}_{nn}^{(l)})}{\mathcal{CN}(g_n|\tilde{m}_{2,n}^{(l)}, \tilde{v}_{2,n}^{(l)})} = \mathcal{CN}(g_n|\tilde{m}_{\backslash 2,n}^{(l)}, \tilde{v}_{\backslash 2,n}^{(l)}), \tag{3.27}$$

where the mean and variance of $q_{\backslash 2,n}^{(l)}(g_n)$ are given by

$$\tilde{v}_{\backslash 2,n}^{(l)} = \left[(\tilde{V}_{nn}^{(l)})^{-1} - (\tilde{v}_{2,n}^{(l)})^{-1} \right]^{-1}, \tag{3.28}$$

$$\tilde{m}_{\backslash 2,n}^{(l)} = \tilde{v}_{\backslash 2,n}^{(l)} \left[(\tilde{V}_{nn}^{(l)})^{-1} \tilde{m}_n^{(l)} - (\tilde{v}_{2,n}^{(l)})^{-1} \tilde{m}_{2,n}^{(l)} \right]. \tag{3.29}$$

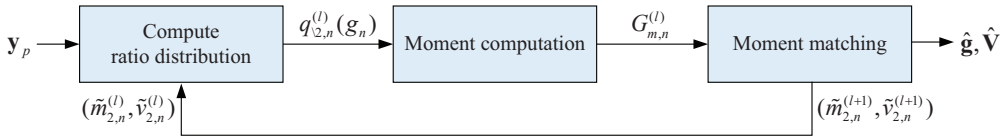


Figure 3.4: Flow chart of an EP iteration.

Second, we update $(\tilde{m}_{2,n}^{(l+1)}, \tilde{v}_{2,n}^{(l+1)})$ to match the mean and variance of $q_n^{(l+1)}(g_n)$ to those of the following distribution

$$\hat{q}_n^{(l)}(g_n) = f_{2,n}(g_n) q_{\backslash 2,n}^{(l)}(g_n). \tag{3.30}$$

To compute the mean $E_q^{(l)}[g_n]$ and variance $V_q^{(l)}[g_n]$ of the distribution $\hat{q}_n^{(l)}(g_n)$, we firstly compute $G_{m,n}^{(l)}$, the m-th ($m = 0, 1, 2$) moments of $\hat{q}_n^{(l)}(g_n)$ with respect to g_n.

First, the zeroth moment $G_{0,n}^{(l)}$ is computed as

$$\begin{aligned}
G_{0,n}^{(l)} &= \int_{-\infty}^{\infty} f_{2,n}(g_n) q_{\backslash 2,n}^{(l)}(g_n) \mathrm{d}g_n \\
&= \int_{-\infty}^{\infty} [(1-p_n)\delta(g_n) + p_n \mathcal{CN}(g_n|0,\alpha_n)] \cdot \mathcal{CN}(g_n|\tilde{m}_{\backslash 2,n}^{(l)}, \tilde{v}_{\backslash 2,n}^{(l)}) \mathrm{d}g_n \\
&= (1-p_n) \int_{-\infty}^{\infty} \delta(g_n) \mathcal{CN}(g_n|\tilde{m}_{\backslash 2,n}^{(l)}, \tilde{v}_{\backslash 2,n}^{(l)}) \mathrm{d}g_n \\
&\quad + p_n \int_{-\infty}^{\infty} \mathcal{CN}(g_n|0,\alpha_n) \mathcal{CN}(g_n|\tilde{m}_{\backslash 2,n}^{(l)}, \tilde{v}_{\backslash 2,n}^{(l)}) \mathrm{d}g_n \\
&= (1-p_n) \mathcal{CN}(0|\tilde{m}_{\backslash 2,n}^{(l)}, \tilde{v}_{\backslash 2,n}^{(l)}) \mathrm{d}g_n \\
&\quad + p_n \int_{-\infty}^{\infty} \mathcal{CN}(g_n|0,\alpha_n) \mathcal{CN}(g_n|\tilde{m}_{\backslash 2,n}^{(l)}, \tilde{v}_{\backslash 2,n}^{(l)}) \mathrm{d}g_n.
\end{aligned} \tag{3.31}$$

The product of two complex Gaussian PDFs, $p_1(g_n) = \mathcal{CN}(g_n|\mu_1, \sigma_1^2)$ and $p_2(g_n) = \mathcal{CN}(g_n|\mu_2, \sigma_2^2)$, is given by [31]

$$p_1(g_n) p_2(g_n) = \mathcal{CN}(\mu_2|\mu_1, \sigma_1^2 + \sigma_2^2) \cdot \mathcal{CN}(g_n|\frac{\mu_1\sigma_2^2 + \mu_2\sigma_1^2}{\sigma_1^2 + \sigma_2^2}, \frac{\sigma_1^2\sigma_2^2}{\sigma_1^2 + \sigma_2^2}). \tag{3.32}$$

Using the above formula, (3.31) is rewritten as

$$\begin{aligned}
G_{0,n}^{(l)} &= (1-p_n) \mathcal{CN}(0|\tilde{m}_{\backslash 2,n}^{(l)}, \tilde{v}_{\backslash 2,n}^{(l)}) \\
&\quad + p_n \int_{-\infty}^{\infty} \mathcal{CN}(0|\tilde{m}_{\backslash 2,n}^{(l)}, \alpha_n + \tilde{v}_{\backslash 2,n}^{(l)}) \cdot \mathcal{CN}(g_n|\frac{\tilde{m}_{\backslash 2,n}^{(l)}\alpha_n}{\alpha_n + \tilde{v}_{\backslash 2,n}^{(l)}}, \frac{\alpha_n \tilde{v}_{\backslash 2,n}^{(l)}}{\alpha_n + \tilde{v}_{\backslash 2,n}^{(l)}}) \mathrm{d}g_n
\end{aligned} \tag{3.33}$$

We next compute the first moment $G_{1,n}^{(l)}$ as

$$\begin{aligned}
G_{1,n}^{(l)} &= \int_{-\infty}^{\infty} g_n f_{2,n}(g_n) q_{\backslash 2,n}^{(l)}(g_n) \mathrm{d}g_n \\
&= \int_{-\infty}^{\infty} g_n \left[(1-p_n)\delta(g_n) + p_n \mathcal{CN}(g_n|0,\alpha_n)\right] \cdot \mathcal{CN}(g_n|\tilde{m}_{\backslash 2,n}^{(l)}, \tilde{v}_{\backslash 2,n}^{(l)}) \mathrm{d}g_n \\
&= (1-p_n) \int_{-\infty}^{\infty} g_n \delta(g_n) \mathcal{CN}(g_n|\tilde{m}_{\backslash 2,n}^{(l)}, \tilde{v}_{\backslash 2,n}^{(l)}) \mathrm{d}g_n \\
&\quad + p_n \int_{-\infty}^{\infty} g_n \mathcal{CN}(g_n|0,\alpha_n) \mathcal{CN}(g_n|\tilde{m}_{\backslash 2,n}^{(l)}, \tilde{v}_{\backslash 2,n}^{(l)}) \mathrm{d}g_n \\
&= p_n \int_{-\infty}^{\infty} g_n \mathcal{CN}(g_n|0,\alpha_n) \mathcal{CN}(g_n|\tilde{m}_{\backslash 2,n}^{(l)}, \tilde{v}_{\backslash 2,n}^{(l)}) \mathrm{d}g_n \\
&= p_n \mathcal{CN}(0|\tilde{m}_{\backslash 2,n}^{(l)}, \alpha_n + \tilde{v}_{\backslash 2,n}^{(l)}) \cdot \int_{-\infty}^{\infty} g_n \mathcal{CN}(g_n|\frac{\tilde{m}_{\backslash 2,n}^{(l)}\alpha_n}{\alpha_n + \tilde{v}_{\backslash 2,n}^{(l)}}, \frac{\alpha_n \tilde{v}_{\backslash 2,n}^{(l)}}{\alpha_n + \tilde{v}_{\backslash 2,n}^{(l)}}) \mathrm{d}g_n \\
&= p_n \mathcal{CN}(0|\tilde{m}_{\backslash 2,n}^{(l)}, \alpha_n + \tilde{v}_{\backslash 2,n}^{(l)}) \frac{\tilde{m}_{\backslash 2,n}^{(l)}\alpha_n}{\alpha_n + \tilde{v}_{\backslash 2,n}^{(l)}}.
\end{aligned} \tag{3.34}$$

Similarly, the second moment $G_{2,n}^{(l)}$ is computed as

$$\begin{aligned}
G_{2,n}^{(l)} &= \int_{-\infty}^{\infty} |g_n|^2 f_{2,n}(g_n) q_{\backslash 2,n}^{(l)}(g_n) \mathrm{d}g_n \\
&= \int_{-\infty}^{\infty} |g_n|^2 \left[(1-p_n)\delta(g_n) + p_n \mathcal{CN}(g_n|0,\alpha_n)\right] \cdot \mathcal{CN}(g_n|\tilde{m}_{\backslash 2,n}^{(l)}, \tilde{v}_{\backslash 2,n}^{(l)}) \mathrm{d}g_n \\
&= (1-p_n) \int_{-\infty}^{\infty} |g_n|^2 \delta(g_n) \mathcal{CN}(g_n|\tilde{m}_{\backslash 2,n}^{(l)}, \tilde{v}_{\backslash 2,n}^{(l)}) \mathrm{d}g_n \\
&\quad + p_n \int_{-\infty}^{\infty} |g_n|^2 \mathcal{CN}(g_n|0,\alpha_n) \mathcal{CN}(g_n|\tilde{m}_{\backslash 2,n}^{(l)}, \tilde{v}_{\backslash 2,n}^{(l)}) \mathrm{d}g_n \\
&= p_n \int_{-\infty}^{\infty} |g_n|^2 \mathcal{CN}(g_n|0,\alpha_n) \mathcal{CN}(g_n|\tilde{m}_{\backslash 2,n}^{(l)}, \tilde{v}_{\backslash 2,n}^{(l)}) \mathrm{d}g_n \\
&= p_n \mathcal{CN}(0|\tilde{m}_{\backslash 2,n}^{(l)}, \alpha_n + \tilde{v}_{\backslash 2,n}^{(l)}) \cdot \int_{-\infty}^{\infty} |g_n|^2 \mathcal{CN}(g_n|\frac{\tilde{m}_{\backslash 2,n}^{(l)}\alpha_n}{\alpha_n + \tilde{v}_{\backslash 2,n}^{(l)}}, \frac{\alpha_n \tilde{v}_{\backslash 2,n}^{(l)}}{\alpha_n + \tilde{v}_{\backslash 2,n}^{(l)}}) \mathrm{d}g_n \\
&= p_n \mathcal{CN}(0|\tilde{m}_{\backslash 2,n}^{(l)}, \alpha_n + \tilde{v}_{\backslash 2,n}^{(l)}) \cdot \left(\left| \frac{\tilde{m}_{\backslash 2,n}^{(l)}\alpha_n}{\alpha_n + \tilde{v}_{\backslash 2,n}^{(l)}} \right|^2 + \frac{\alpha_n \tilde{v}_{\backslash 2,n}^{(l)}}{\alpha_n + \tilde{v}_{\backslash 2,n}^{(l)}} \right).
\end{aligned} \tag{3.35}$$

Then, $E_q^{(l)}[g_n]$ and $V_q^{(l)}[g_n]$ are calculated as

$$E_q^{(l)}[g_n] = \frac{G_{1,n}^{(l)}}{G_{0,n}^{(l)}}, \tag{3.36}$$

$$V_q^{(l)}[g_n] = \frac{G_{2,n}^{(l)}}{G_{0,n}^{(l)}} - \left| E_q^{(l)}[g_n] \right|^2. \tag{3.37}$$

Finally, we update the parameter pair $(\tilde{m}_{2,n}^{(l+1)}, \tilde{v}_{2,n}^{(l+1)})$ such that the unnormalized distribution $q_n^{(l+1)}(g_n) = q_{2,n}^{(l+1)}(g_n) q_{\backslash 2,n}^{(l)}(g_n)$ has the mean $E_q^{(l)}[g_n]$ and variance $V_q^{(l)}[g_n]$ obtained from (3.36) and (3.37), respectively. The corresponding solution is

$$\tilde{v}_{2,n}^{(l+1)} = \left[V_q^{(l)}[g_n]^{-1} - (\tilde{v}_{\backslash 2,n}^{(l)})^{-1} \right]^{-1}, \tag{3.38}$$

$$\tilde{m}_{2,n}^{(l+1)} = \tilde{v}_{2,n}^{(l+1)} \left[V_q^{(l)}[g_n]^{-1} E_q^{(l)}[g_n] - (\tilde{v}_{\backslash 2,n}^{(l)})^{-1} \tilde{m}_{\backslash 2,n}^{(l)} \right]. \tag{3.39}$$

We stop the algorithm when the variation of the mean vector $\tilde{\mathbf{m}}$ is smaller than the given threshold ϵ (e.g., $\epsilon = 10^{-4}$) or the maximum number of iterations has been reached. For the robust convergence of the proposed algorithm, we can smooth the parameter update by taking a combination of the previous and new parameter values as

$$\tilde{m}_{2,n}^{(l+1)} = \beta \tilde{m}_{2,n}^{\text{new}} + (1-\beta) \tilde{m}_{2,n}^{(l)}, \tag{3.40}$$

$$\tilde{v}_{2,n}^{(l+1)} = \beta \tilde{v}_{2,n}^{\text{new}} + (1-\beta) \tilde{v}_{2,n}^{(l)}, \tag{3.41}$$

where $\beta \in [0, 1]$ is the smoothing parameter and $(\tilde{m}_{2,n}^{\text{new}}, \tilde{v}_{2,n}^{\text{new}})$ is the new parameter pair computed in (3.38) and (3.39).

After the convergence of the EP algorithm, an approximate distribution of $f(\mathbf{g}) = p(\mathbf{y}_p|\mathbf{g}) p(\mathbf{g})$ is obtained as

$$q(\mathbf{g}) = \mathcal{CN}(\mathbf{g}|\tilde{\mathbf{m}}, \tilde{\mathbf{V}}), \tag{3.42}$$

where $\tilde{\mathbf{m}}$ and $\tilde{\mathbf{V}}$ are obtained from (3.25) and (3.26). The final solution of the iterative EP estimator $\hat{\mathbf{g}}$ is the mean vector of $q(\mathbf{g})$ given by

$$\hat{\mathbf{g}} = E_{q(\mathbf{g})}[\mathbf{g}] = \tilde{\mathbf{m}}. \tag{3.43}$$

and the covariance matrix of $\hat{\mathbf{g}}$ is $\tilde{\mathbf{V}}$. Once $\hat{\mathbf{g}}$ is obtained, by performing the log-likelihood test on $\hat{\mathbf{g}}$, we detect the active devices. Then, we find the corresponding channel of each active device.

3.3.4 Active User Detection and Channel Estimation

To perform AUD, we employ the likelihood ratio test on the vector $\hat{\mathbf{g}}$. The hypothesis testing to find out the active device is given by

$$\begin{cases} H_1 & : a_n = 1, \quad \text{active device}, \\ H_0 & : a_n = 0, \quad \text{inactive device}, \end{cases} \tag{3.44}$$

and the corresponding log-likelihood ratio test is

$$\text{LLR}(\hat{g}_n) = \log\left(\frac{p_{\hat{g}_n|a_n}(\hat{g}_n|a_n \neq 0)}{p_{\hat{g}_n|a_n}(\hat{g}_n|a_n = 0)}\right) \underset{H_0}{\overset{H_1}{\gtrless}} 0. \tag{3.45}$$

Since the n-th component $\hat{g}_n$ of $\hat{\mathbf{g}}$ has the variance $\tilde{V}_{nn}$ (the n-th diagonal of $\tilde{\mathbf{V}}$), the likelihood probabilities of $\hat{g}_n$ given $a_n \neq 0$ or $a_n = 0$ are given, respectively, by

$$p_{\hat{g}_n|a_n}(\hat{g}_n|a_n \neq 0) = \frac{1}{\pi(\alpha_n + \tilde{V}_{nn})} \exp\left(\frac{-|\hat{g}_n|^2}{\alpha_n + \tilde{V}_{nn}}\right), \tag{3.46}$$

$$p_{\hat{g}_n|a_n}(\hat{g}_n|a_n = 0) = \frac{1}{\pi\tilde{V}_{nn}} \exp\left(\frac{-|\hat{g}_n|^2}{\tilde{V}_{nn}}\right). \tag{3.47}$$

and the corresponding log-likelihood ratio is

$$\begin{aligned} \text{LLR}(\hat{g}_n) &= \log\left(\frac{p_{\hat{g}_n|a_n}(\hat{g}_n|a_n \neq 0)}{p_{\hat{g}_n|a_n}(\hat{g}_n|a_n = 0)}\right), \\ &= \log\left(\frac{\tilde{V}_{nn}}{\alpha_n + \tilde{V}_{nn}} \exp\left(|\hat{g}_n|^2 \left(\frac{1}{\tilde{V}_{nn}} - \frac{1}{\alpha_n + \tilde{V}_{nn}}\right)\right)\right) \underset{H_0}{\overset{H_1}{\gtrless}} 0. \end{aligned} \tag{3.48}$$

The log-likelihood ratio test in (3.48) can be simplified to

$$|\hat{g}_n|^2 \underset{H_0}{\overset{H_1}{\gtrless}} \theta_n, \tag{3.49}$$

where

$$\theta_n = \log(1 + \frac{\alpha_n}{\tilde{V}_{nn}})/(\frac{1}{\tilde{V}_{nn}} - \frac{1}{\alpha_n + \tilde{V}_{nn}}). \tag{3.50}$$

The estimate of activity vector $\hat{\mathbf{a}}$ is obtained after the thresholding of each element of $\hat{\mathbf{g}}$ as

$$\hat{a}_n = \begin{cases} 1 & \text{if } |\hat{g}_n|^2 \geq \theta_n, \\ 0 & \text{if } |\hat{g}_n|^2 < \theta_n. \end{cases} \tag{3.51}$$

If the n-th device is detected to be active, we use $\hat{g}_n$ as the CSI estimate of the device.

3.3.5 Data Detection

Based on the knowledge of user activities and channels obtained in the AUD/CE phase, data symbols of active devices are detected. The measurement signal $\mathbf{y}_d^{[i]}$ of the i-th data symbol vector is given by

$$\mathbf{y}_d^{[i]} = \sum_{n \in \mathcal{N}} \mathbf{s}_n h_n x_{d,n}^{[i]} + \mathbf{w}_d^{[i]}, \quad i = 1, \ldots, J, \tag{3.52}$$

where $\mathcal{N} = \{n_1, \ldots, n_{|\mathcal{N}|}\}$ is the true support[2] (index set of active devices) of active devices and $x_{d,n}^{[i]}$ is the i-th data symbol of the n-th device drawn from the finite alphabet $\mathcal{A}$, respectively.

Let $\mathbf{S}_{\mathcal{N}} = [\mathbf{s}_{n_1}, \ldots, \mathbf{s}_{n_{|\mathcal{N}|}}]$ be the matrix containing the spreading sequences of active devices and $\mathbf{h}_{\mathcal{N}} = [h_{n_1}, \ldots, h_{n_{|\mathcal{N}|}}]^T$ be the vector containing the channels of active devices. Then (3.52) can be rewritten as

$$\mathbf{y}_d^{[i]} = \mathbf{S}_{\mathcal{N}} \text{diag}(\mathbf{h}_{\mathcal{N}}) \mathbf{x}_{d,\mathcal{N}}^{[i]} + \mathbf{w}_d^{[i]} \tag{3.53}$$

where $\mathbf{x}_{d,\mathcal{N}}^{[i]} \in \mathcal{A}^{|\mathcal{N}|}$ is the i-th data symbols of all active devices.

Let $\hat{\mathcal{N}} = \{\hat{n}_1, \ldots, \hat{n}_{|\hat{\mathcal{N}}|}\}$ be the estimated support. Then let $\mathbf{S}_{\hat{\mathcal{N}}}$ be the spreading sequences of devices in $\hat{\mathcal{N}}$ and $\hat{\mathbf{h}}_{\hat{\mathcal{N}}}$ be the estimated channels of devices in $\hat{\mathcal{N}}$. Further, let $\tilde{\mathbf{L}}_{\hat{\mathcal{N}}}$ be

$$\tilde{\mathbf{L}}_{\hat{\mathcal{N}}} = \mathbf{S}_{\hat{\mathcal{N}}} \text{diag}(\hat{\mathbf{h}}_{\hat{\mathcal{N}}}) = (\mathbf{s}_{\hat{n}_1} \hat{h}_{\hat{n}_1}, \ldots, \mathbf{s}_{\hat{n}_{|\hat{\mathcal{N}}|}} \hat{h}_{\hat{n}_{|\hat{\mathcal{N}}|}}), \tag{3.54}$$

[2]For example, if $\mathbf{a} = [1\ 0\ 1\ 0\ 0\ 0]^T$, then the support T is $T = \{1, 3\}$.

then the BS employs the linear MMSE detector followed by the quantization to detect the data symbols as

$$\hat{\mathbf{x}}_{d,\hat{\mathcal{N}}}^{[i]} = Q_{\mathcal{A}}\left(\left(\tilde{\mathbf{L}}_{\hat{\mathcal{N}}}^{H}\tilde{\mathbf{L}}_{\hat{\mathcal{N}}} + \sigma_w^2\mathbf{I}_{|\mathcal{N}|}\right)^{-1}\tilde{\mathbf{L}}_{\hat{\mathcal{N}}}^{H}\mathbf{y}_d^{[i]}\right), \quad i = 1,\ldots,J, \tag{3.55}$$

where $Q_{\mathcal{A}}(\cdot)$ is the quantization operator on the finite alphabet $\mathcal{A}$. In Table 3.1, we summarize the proposed algorithm.

3.3.6 Comments on Complexity

In this subsection, we briefly discuss the computational complexity of the proposed algorithm. In each EP iteration, computations of marginals and moments, and updating the parameter pairs $(\tilde{m}_{2,n}^{(l)}, \tilde{v}_{2,n}^{(l)})$ for all $n = 1,\ldots,N$ has a small linear complexity of $\mathcal{O}(N)$ because they are composed only of arithmetic operations such as addition, subtraction, multiplication, and division. The complexity of the EP algorithm is dominated by the computation of the covariance matrix $\tilde{\mathbf{V}}$ in (3.25) and the mean vector $\tilde{\mathbf{m}}$ in (3.26). When N is large, direct computation of $\tilde{\mathbf{V}}$ and $\tilde{\mathbf{m}}$ is computationally burdensome due to the computationally expensive matrix inversion. When $M < N$ and $\tilde{\mathbf{V}}_2$ is diagonal, the Woodbury matrix identity [32] offers an efficient way to compute $\tilde{\mathbf{V}}$. To be specific,

$$\begin{aligned}\tilde{\mathbf{V}} &= \left(\sigma_w^{-2}\mathbf{\Phi}^H\mathbf{\Phi} + \tilde{\mathbf{V}}_2^{-1}\right)^{-1} \\ &= \tilde{\mathbf{V}}_2 - \tilde{\mathbf{V}}_2\mathbf{\Phi}^H\left(\sigma_w^2\mathbf{I}_M + \mathbf{\Phi}\tilde{\mathbf{V}}_2\mathbf{\Phi}^H\right)^{-1}\mathbf{\Phi}\tilde{\mathbf{V}}_2.\end{aligned} \tag{3.56}$$

Note that $\left(\sigma_w^2\mathbf{I}_M + \mathbf{\Phi}\tilde{\mathbf{V}}_2\mathbf{\Phi}^H\right)^{-1}$ can be computed in the order of $\mathcal{O}(M^3)$ by a Cholesky decomposition. Thus, the computational complexity order to compute (3.56) is reduced from $\mathcal{O}(N^3)$ to $\mathcal{O}(MN^2)$.

Similarly, the posterior mean $\tilde{\mathbf{m}}$ can be computed in the order of $\mathcal{O}(MN^2)$ as

$$\begin{aligned}\tilde{\mathbf{m}} &= \tilde{\mathbf{V}}\left(\sigma_w^{-2}\mathbf{\Phi}^H\mathbf{y} + \tilde{\mathbf{V}}_2^{-1}\tilde{\mathbf{m}}_2\right) \\ &= \left[\tilde{\mathbf{V}}_2 - \tilde{\mathbf{V}}_2\mathbf{\Phi}^H\left(\sigma_w^2\mathbf{I}_M + \mathbf{\Phi}\tilde{\mathbf{V}}_2\mathbf{\Phi}^H\right)^{-1}\mathbf{\Phi}\tilde{\mathbf{V}}_2\right]\left(\sigma_w^{-2}\mathbf{\Phi}^H\mathbf{y} + \tilde{\mathbf{V}}_2^{-1}\tilde{\mathbf{m}}_2\right) \\ &= \tilde{\mathbf{V}}_2\boldsymbol{\eta} - \tilde{\mathbf{V}}_2\mathbf{\Phi}^H\left(\sigma_w^2\mathbf{I}_M + \mathbf{\Phi}\tilde{\mathbf{V}}_2\mathbf{\Phi}^H\right)^{-1}\mathbf{\Phi}\tilde{\mathbf{V}}_2\boldsymbol{\eta}, \end{aligned} \tag{3.57}$$

where $\boldsymbol{\eta} = \sigma_w^{-2}\mathbf{\Phi}^H\mathbf{y} + \tilde{\mathbf{V}}_2^{-1}\tilde{\mathbf{m}}_2$. When the EP algorithm runs L iterations ($L = 2 \sim 3$), the total computational complexity order of the EP approximation would be $\mathcal{O}(LMN^2)$.

3.4 Simulation Results and Discussions

3.4.1 Simulation Setup

In our simulations, we simulate underdetermined mMTC systems with $N = 128$ MTDs and M-dimensional unit-norm random spreading sequences ($M < N$). Devices are randomly located in a cell with a radius 200m. The pathloss component of the wireless channel between the n-th device and the BS is modeled as $\alpha_n = -128.1 - 36.7\log_{10}(d_n)$ in dB scale where d_n is the distance (in km) between the n-th device and the BS. The noise spectral density and transmission bandwidth are set to -170 dBm/Hz and 1 MHz, respectively. The activity probabilities of all devices are set to p_a (i.e., $p_n = p_a$ for all n). Note that the proposed technique can also be applied to scenarios with heterogeneous activity probabilities. The number of data symbols in a frame is set to $J = 9$. Data symbols of active devices are modulated with quadrature phase shift keying (QPSK) and the pilot symbol $x_{p,n}$ is set to 1 for simplicity.

As performance measures, we consider the activity error rate (AER), the net normalized mean squared error (NNMSE), and the net symbol error rate (NSER). The AER refers to the percentage of errors (both missed detections and false alarms) in AUD. The NNMSE, defined as the NMSE of the estimated channels of active devices,

is computed as $\frac{\|\hat{\mathbf{h}}_{\mathcal{N}}-\mathbf{h}_{\mathcal{N}}\|_2^2}{\|\mathbf{h}_{\mathcal{N}}\|_2^2}$ where $\mathcal{N}$ is the support of active devices. The NSER refers to the data symbol error rate of active devices.

As a reference, we compare the proposed EP estimator with the orthogonal matching pursuit (OMP), and approximate message passing (AMP). As Bayesian sparse recovery algorithms exploiting the prior distribution of user activities and channels, we also test the sparse Bayesian learning (SBL) [33,34] and Bayesian compressive sensing (BCS) [35,36]. SBL implicitly estimates the prior distribution from the received signal and computes a MAP estimate of the target signal by expectation maximization (EM) iterations. BCS is an extension of SBL accounting for the contribution of the noise variance. Lastly, as the best achievable bound of the estimation techniques, we use the Oracle minimum mean squared error (MMSE) detector. Since support information is given in the Oracle MMSE detector, it can solve the problem in the overdetermined setup. Each point of the performance figure represents an average of at least 100,000 realizations of the activities and channels.

Table 3.1: SUMMARY OF THE PROPOSED EP-AUD/CE ALGORITHM

Input: The pilot measurements $\mathbf{y}_p$, the data measurements $\mathbf{y}_d^{[i]}$, the spreading sequence $\{\mathbf{s}_n\}$, the pilot symbol $\{x_{p,n}\}$, the activity probability $\{p_n\}$, the channel variances $\{\alpha_n\}$, the noise variance σ_w^2, the finite alphabet $\mathcal{A}$

Output: The estimated support of active devices $\hat{\mathcal{N}}$, the estimated channel vector $\hat{\mathbf{h}}_{\hat{\mathcal{N}}}$, the detected data symbol vectors $\hat{\mathbf{x}}_{d,\hat{\mathcal{N}}}^{[i]}$

Subscript and superscript: The device index n, the iteration index l, the data symbol index i

Step 1: (Initialization)

$\mathbf{\Phi} = [\mathbf{s}_1 x_{p,1}, \ldots, \mathbf{s}_N x_{p,N}]$

$\tilde{\mathbf{m}}_2^{(1)} = \mathbf{0}_N$, $\tilde{\mathbf{V}}_2^{(1)} = \mathrm{diag}(p_1\alpha_1, \ldots, p_N\alpha_N)$

$\tilde{\mathbf{V}}^{(1)} = \left(\sigma_w^{-2}\mathbf{\Phi}^H\mathbf{\Phi} + (\tilde{\mathbf{V}}_2^{(1)})^{-1}\right)^{-1}$, $\tilde{\mathbf{m}}^{(1)} = \tilde{\mathbf{V}}\left(\sigma_w^{-2}\mathbf{\Phi}^H\mathbf{y}_p + (\tilde{\mathbf{V}}_2^{(1)})^{-1}\tilde{\mathbf{m}}_2\right)$

Step 2: (Compute ratio distribution) Compute the parameters of $q_{\backslash 2,n}^{(l)}$.

$\tilde{v}_{\backslash 2,n}^{(l)} = \left[(\tilde{V}_{nn}^{(l)})^{-1} - (\tilde{v}_{2,n}^{(l)})^{-1}\right]^{-1}$

$\tilde{m}_{\backslash 2,n}^{(l)} = \tilde{v}_{\backslash 2,n}^{(l)}\left[(\tilde{V}_{nn}^{(l)})^{-1}\tilde{m}_n^{(l)} - (\tilde{v}_{2,n}^{(l)})^{-1}\tilde{m}_{2,n}^{(l)}\right]$

Step 3: (Moment computation) Compute the mean $E_q^{(l)}[g_n]$ and variance $V_q^{(l)}[g_n]$ of $f_{2,n}q_{\backslash 2,n}^{(l)}$.

$G_{0,n}^{(l)} = (1-p_n)\mathcal{CN}(0|\tilde{m}_{\backslash 2,n}^{(l)}, \tilde{v}_{\backslash 2,n}^{(l)}) + p_n\mathcal{CN}(0|\tilde{m}_{\backslash 2,n}^{(l)}, \alpha_n + \tilde{v}_{\backslash 2,n}^{(l)})$

$G_{1,n}^{(l)} = p_n\mathcal{CN}(0|\tilde{m}_{\backslash 2,n}^{(l)}, \alpha_n + \tilde{v}_{\backslash 2,n}^{(l)})\frac{\tilde{m}_{\backslash 2,n}^{(l)}\alpha_n}{\alpha_n + \tilde{v}_{\backslash 2,n}^{(l)}}$

$G_{2,n}^{(l)} = p_n\mathcal{CN}(0|\tilde{m}_{\backslash 2,n}^{(l)}, \alpha_n + \tilde{v}_{\backslash 2,n}^{(l)}) \cdot \left(\left|\frac{\tilde{m}_{\backslash 2,n}^{(l)}\alpha_n}{\alpha_n + \tilde{v}_{\backslash 2,n}^{(l)}}\right|^2 + \frac{\alpha_n\tilde{v}_{\backslash 2,n}^{(l)}}{\alpha_n + \tilde{v}_{\backslash 2,n}^{(l)}}\right)$

$E_q^{(l)}[g_n] = \frac{G_{1,n}^{(l)}}{G_{0,n}^{(l)}}$

$V_q^{(l)}[g_n] = \frac{G_{2,n}^{(l)}}{G_{0,n}^{(l)}} - \left|E_q^{(l)}[g_n]\right|^2$

Step 4: (Moment matching) Update the parameters $\tilde{v}_{2,n}^{(l+1)}$ and $\tilde{m}_{2,n}^{(l+1)}$.

$\tilde{v}_{2,n}^{(l+1)} = \left[V_q^{(l)}[g_n]^{-1} - (\tilde{v}_{\backslash 2,n}^{(l)})^{-1}\right]^{-1}$

$\tilde{m}_{2,n}^{(l+1)} = \tilde{v}_{2,n}^{(l+1)}\left[V_q^{(l)}[g_n]^{-1}E_q^{(l)}[g_n] - (\tilde{v}_{\backslash 2,n}^{(l)})^{-1}\tilde{m}_{\backslash 2,n}^{(l)}\right]$

Step 5: (Iteration) Repeat **Step 2 - 4** until stopping criteria, $\forall n$. After the convergence of $\tilde{\mathbf{V}}_2$ and $\tilde{\mathbf{m}}_2$,

$\tilde{\mathbf{V}} = \left(\sigma_w^{-2}\mathbf{\Phi}^H\mathbf{\Phi} + \tilde{\mathbf{V}}_2^{-1}\right)^{-1}$

$\hat{\mathbf{g}} = \tilde{\mathbf{m}} = \tilde{\mathbf{V}}\left(\sigma_w^{-2}\mathbf{\Phi}^H\mathbf{y}_p + \tilde{\mathbf{V}}_2^{-1}\tilde{\mathbf{m}}_2\right)$

Step 6: (Active user detection channel estimation) Threshold $\hat{\mathbf{g}}$ using the threshold θ_n in (3.50).

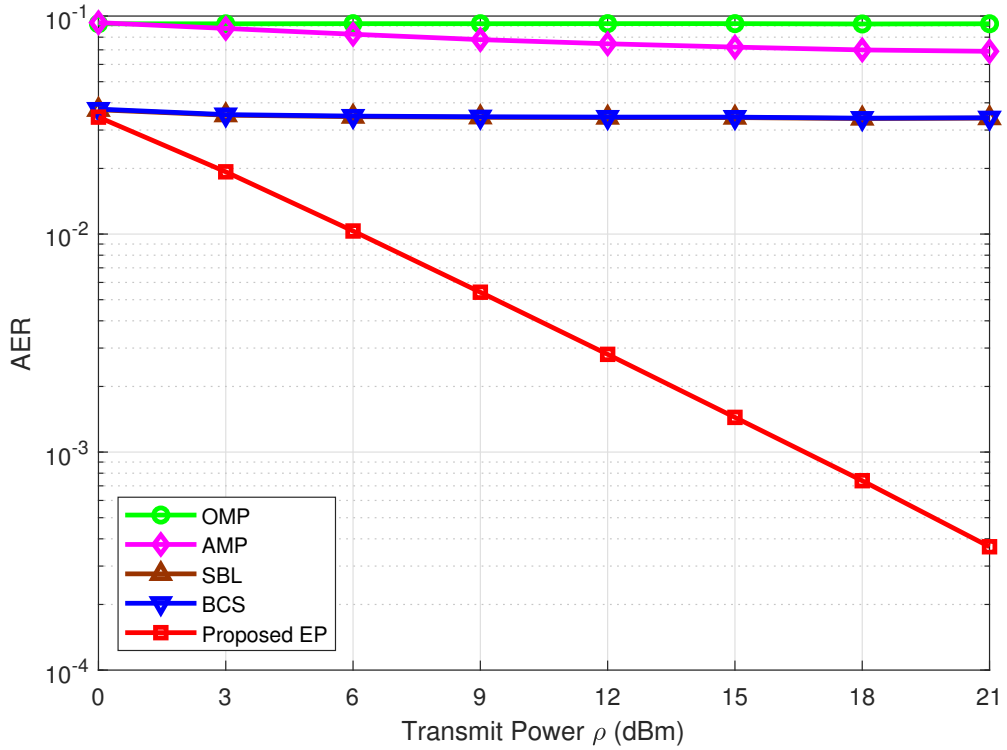
$\hat{\mathcal{N}} = \{n||\hat{g}_n|^2 \geq \theta_n\}$

$\hat{h}_n = \hat{g}_n, \forall n \in \hat{\mathcal{N}}$

Step 7: (Data detection) Use the linear MMSE detector to detect the data symbols.

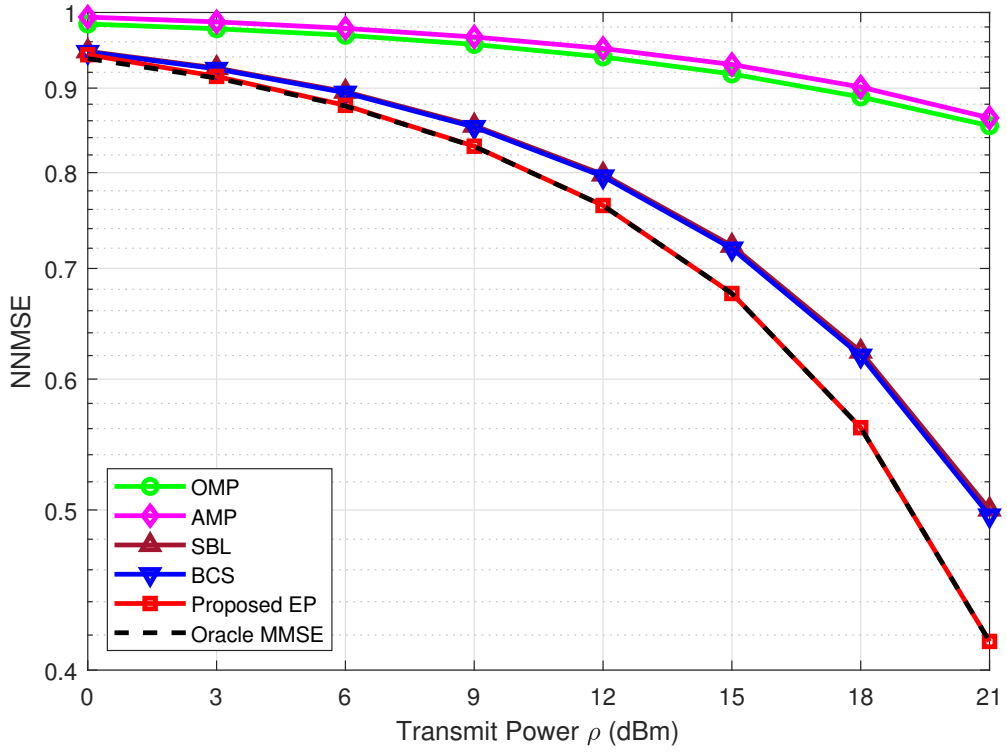
$\tilde{\mathbf{L}}_{\hat{\mathcal{N}}} = \left[\mathbf{s}_1\hat{h}_1, \ldots, \mathbf{s}_n\hat{h}_n, \ldots, \mathbf{s}_{|\hat{\mathcal{N}}|}\hat{h}_{|\hat{\mathcal{N}}|}\right]$ for $n \in \hat{\mathcal{N}}$

$\hat{\mathbf{x}}_{d,\hat{\mathcal{N}}}^{[i]} = Q_{\mathcal{A}}\left(\left(\tilde{\mathbf{L}}_{\hat{\mathcal{N}}}^H\tilde{\mathbf{L}}_{\hat{\mathcal{N}}} + \sigma_w^2\mathbf{I}_{|\mathcal{N}|}\right)^{-1}\tilde{\mathbf{L}}_{\hat{\mathcal{N}}}^H\mathbf{y}_d^{[i]}\right)$ for $i \in [1:J]$



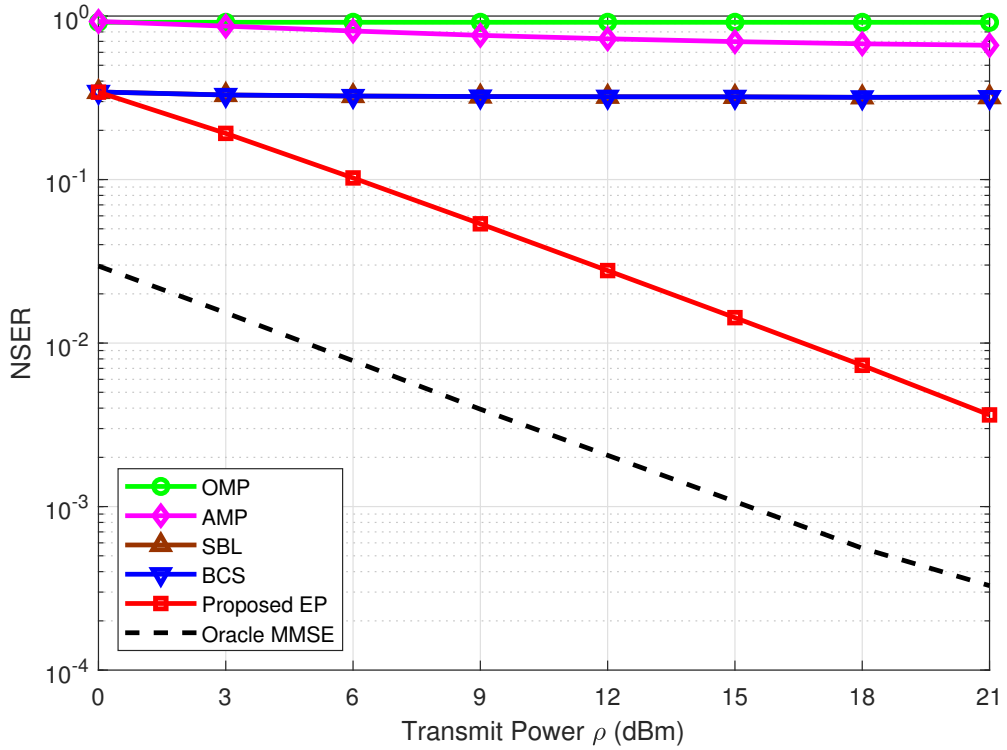
(a)

Figure 3.5: (a) AER as a function of the transmit power ρ.



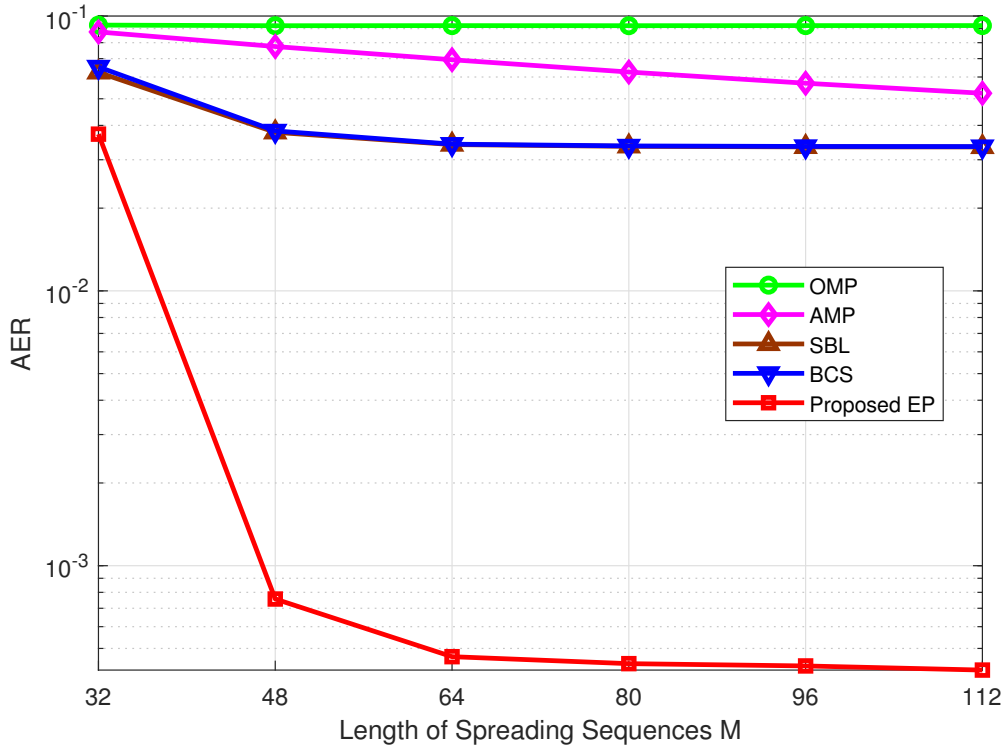
(b)

Figure 3.5: (b) NNMSE as a function of the transmit power ρ.



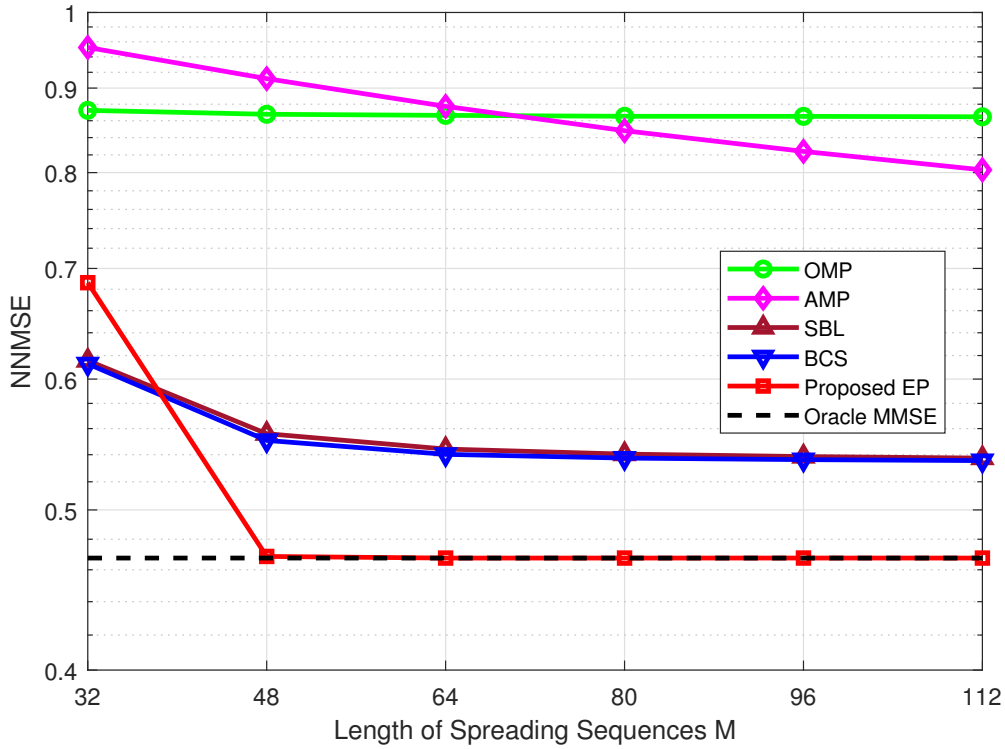
(c)

Figure 3.5: (c) NSER as a function of the transmit power ρ.



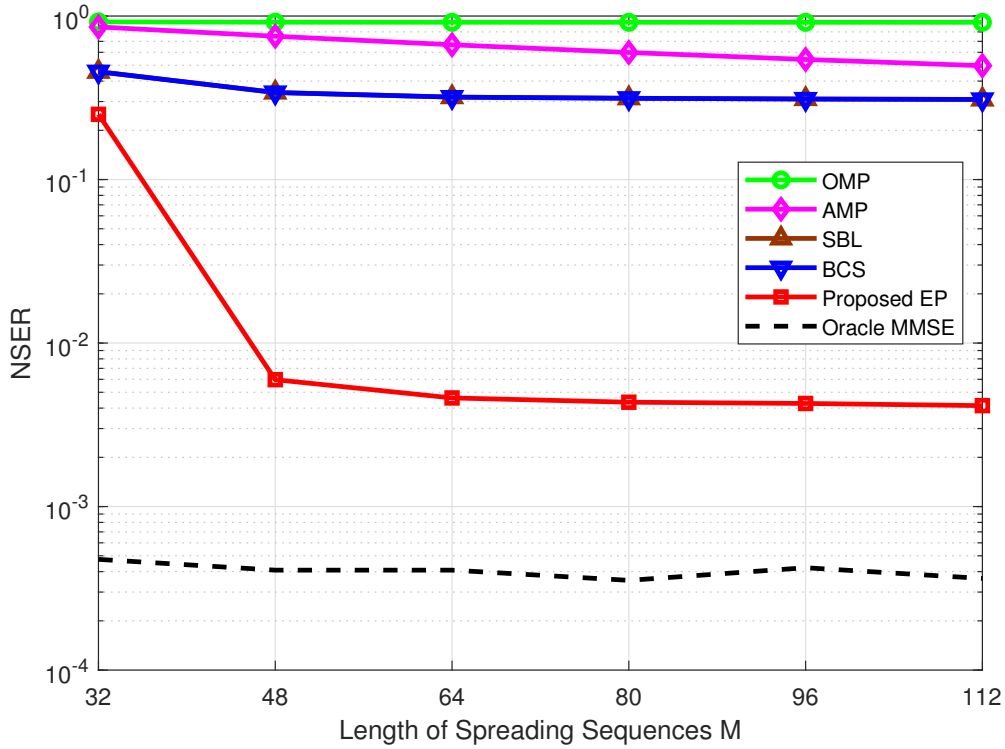
(a)

Figure 3.6: (a) AER as a function of the length of spreading sequences M.



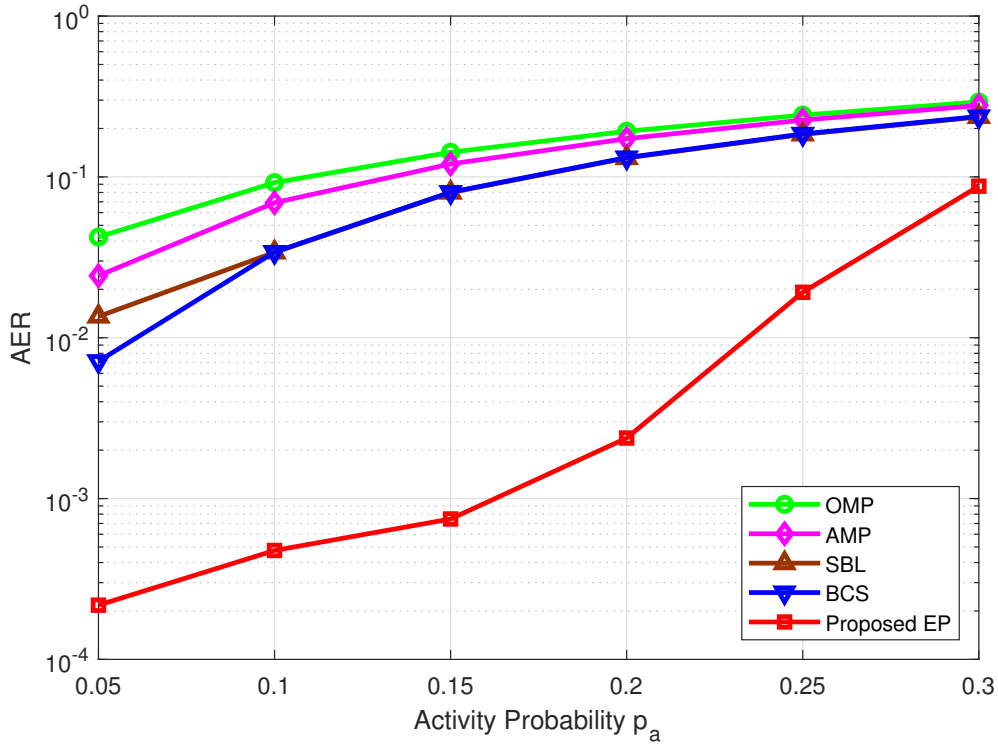
(b)

Figure 3.6: (b) NNMSE as a function of the length of spreading sequences M.



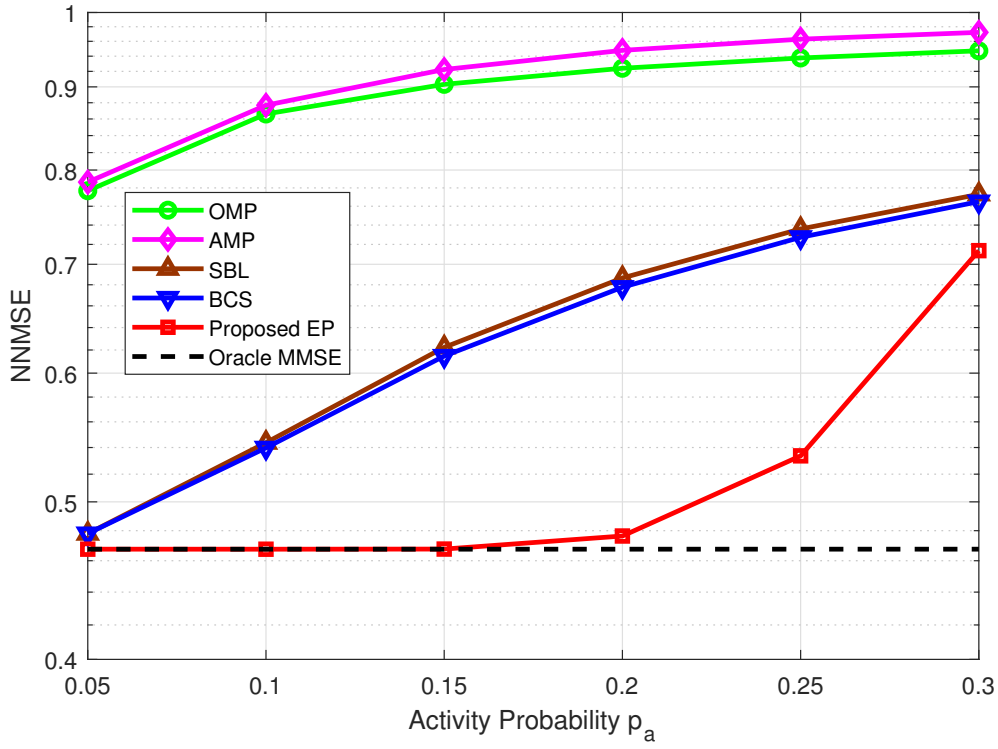
(c)

Figure 3.6: (c) NSER as a function of the length of spreading sequences M.



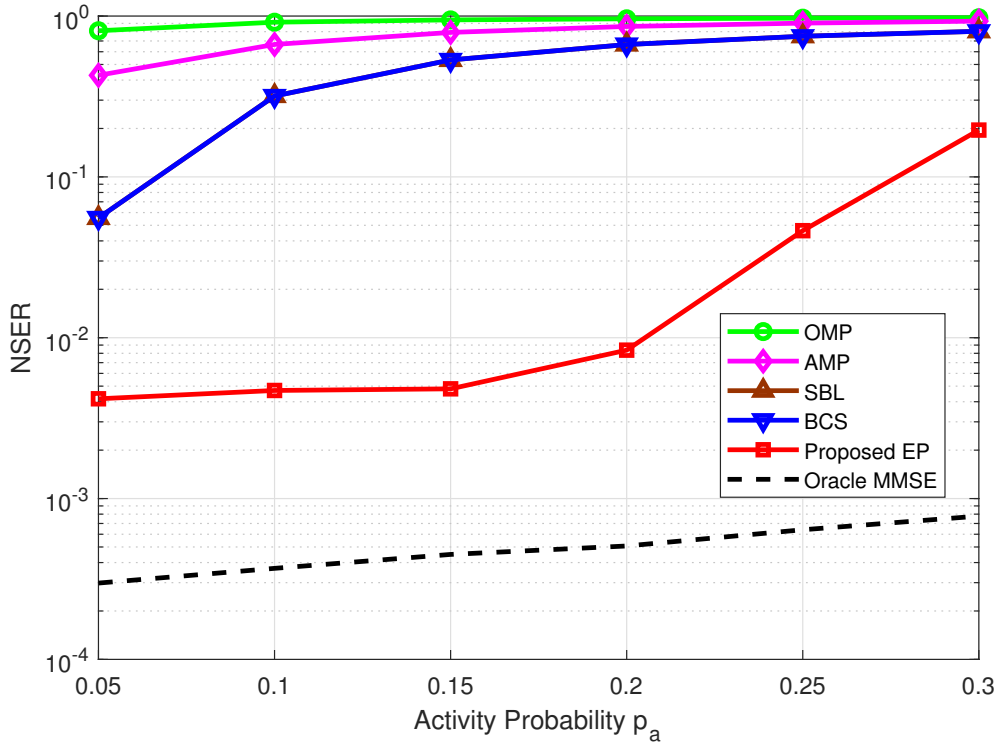
(a)

Figure 3.7: (a) AER as a function of the activity probability p_a.



(b)

Figure 3.7: (b) NNMSE as a function of the activity probability p_a.



(c)

Figure 3.7: (c) NSER as a function of the activity probability p_a.

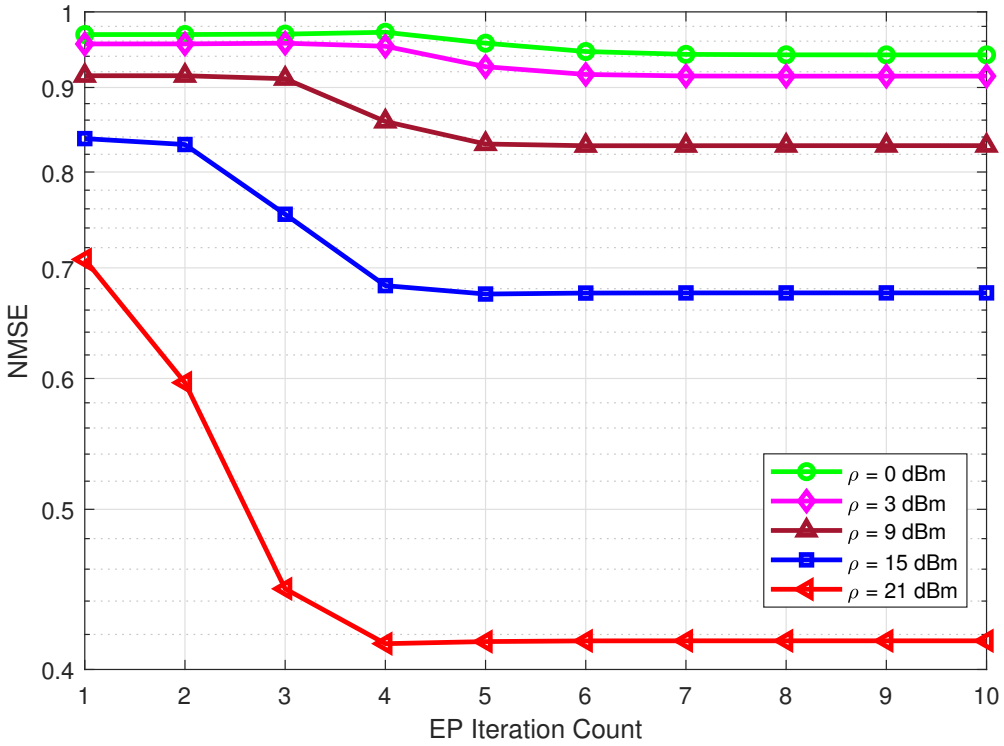


Figure 3.8: NMSE of the target vector $\mathbf{g}$ as a function of the EP iteration index l.

3.4.2 Simulation Results

Fig. 3.5 shows the AER, NNMSE, and NSER performance when $p_a = 0.1$ and $M = 64$. The non-Bayesian greedy algorithms such as OMP and AMP perform worse than the Bayesian algorithms because they do not rely on the statistical distributions of user activities and channels. SBL and BCS outperform the non-Bayesian greedy algorithms. However, since SBL and BCS exploit the distribution derived from the received signal (which is not necessarily correct), there is a substantial performance gap from the proposed technique. We observe that the proposed EP estimator outperforms the conventional algorithms. In particular, the proposed technique performs close to the Oracle MMSE estimator when the transmit power is larger than 12 dBm.

Fig. 3.6 shows the performance when the length of spreading sequences M varies from 32 to 96. In this simulation, we set the transmit power to $\rho = 20$ dBm. We can clearly observe that the performance degrades as M decreases because the ratio M/N of the system decreases (i.e., the system becomes more underdetermined). Note that the performance in the small M/N regime is important for the massive connectivity scenario. Even when $M/N = 0.25$, the proposed technique achieves acceptable performance with the NSER of about 10^{-1}.

Fig. 3.7 shows the performance when the activity probability p_a varies from 0.05 to 0.3. The transmit power and the length of spreading sequences are set to $\rho = 20$ dBm and $M = 64$, respectively. We observe that all algorithms under test including the proposed algorithm are degraded when p_a increases. This is because the interference among devices increases as more devices are active. The proposed algorithm outperforms OMP, AMP, SBL, and BCS by a large margin even if p_a is higher than 0.2.

In Fig. 3.8, we investigate the convergence behavior of the proposed EP estimator. We plot the normalized mean squared error (NMSE) of the estimate of the target vector $\mathbf{g}$, i.e., $\frac{\|\hat{\mathbf{g}}-\mathbf{g}\|_2^2}{\|\mathbf{g}\|_2^2}$, as a function of the EP iteration index l. The smoothing parameter β is set to 0.9. As shown in Fig. 3.8, we observe that the proposed EP estimator converges to the true solution within 4 to 6 iterations for all transmission power regimes. The

iterative EP algorithm provides a reliable approximate distribution $q(\mathbf{g})$ whose mean is close to the mean of the target distribution $p(\mathbf{g})$ only with a few iterations.

Chapter 4

Conclusion

In this dissertation, two novel sparsity-aware multi-user detection techniques are proposed. First proposed is a novel sparsity-aware ordered SIC scheme for mMTC systems. Based on activity probabilities and channel gains of devices, the proposed scheme constructs an augmented system and finds the optimal detection order. Numerical simulations have demonstrated that the proposed algorithm greatly improves the performance of SA-SIC. Next, an EP-based Bayesian joint active user detection (AUD) and channel estimation (CE) technique for mMTC systems is proposed. The work is motivated by the observation that most of conventional CS-MUD schemes are based on non-Bayesian approaches, and they cannot make the most of the statistical prior distribution of the user activities and channels. By exploiting the prior distribution, the proposed technique iteratively finds the best approximate Gaussian distribution that is close to the posterior distribution of the target composite vector of the user activities and channels. Then, by solving the MAP estimation problem with the approximate distribution, AUD and CE are performed jointly. The data detection (DD) of active devices are then performed based on the obtained knowledge of user activities and channels. From numerical simulations, the proposed technique has been shown to achieve significant performance gains in terms of AUD, CE, and DD over the conventional sparse recovery algorithms.

Bibliography

[1] C. Bockelmann, N. K. Pratas, G. Wunder, S. Saur, M. Navarro, D. Gregoratti, G. Vivier, E. D. Carvalho, Y. Ji, Č. Stefanović, P. Popovski, Q. Wang, M. Schellmann, E. Kosmatos, P. Demestichas, M. Raceala-Motoc, P. Jung, S. Stanczak, and A. Dekorsy, "Towards massive connectivity support for scalable mMTC communications in 5G networks," *IEEE Access*, vol. 6, pp. 28 969–28 992, May 2018.

[2] Rec. ITU-R M.2083-0, "IMT Vision - Framework and overall objectives of the future development of IMT for 2020 and beyond," Sep. 2015.

[3] Z. Dawy, W. Saad, A. Ghosh, J. G. Andrews, and E. Yaacoub, "Toward massive machine type cellular communications," *IEEE Wireless Commun.*, vol. 24, no. 1, pp. 120–128, Feb. 2017.

[4] H. Shariatmadari, R. Ratasuk, S. Iraji, A. Laya, T. Taleb, R. Jäntti, and A. Ghosh, "Machine-type communications: current status and future perspectives toward 5G systems," *IEEE Commun. Mag.*, vol. 53, no. 9, pp. 10–17, Sep. 2015.

[5] L. Liu, E. G. Larsson, W. Yu, P. Popovski, C. Stefanovic, and E. D. Carvalho, "Sparse signal processing for grant-free massive iot connectivity," *IEEE Signal Process. Mag.*, 2018, to appear.

[6] Z. Ding, Y. Liu, J. Choi, Q. Sun, M. Elkashlan, C. L. I, and H. V. Poor, "Application of non-orthogonal multiple access in LTE and 5G networks," *IEEE Commun. Mag.*, vol. 55, no. 2, pp. 185–191, Feb. 2017.

[7] M. Shirvanimoghaddam, M. Dohler, and S. J. Johnson, "Massive non-orthogonal multiple access for cellular IoT: Potentials and limitations," *IEEE Commun. Mag.*, vol. 55, no. 9, pp. 55–61, Sep. 2017.

[8] B. K. Jeong, B. Shim, and K. B. Lee, "MAP-based active user and data detection for massive machine-type communications," *IEEE Trans. Veh. Technol.*, vol. 67, no. 9, pp. 8481–8494, Sep. 2018.

[9] J. W. Choi, B. Shim, Y. Ding, B. Rao, and D. I. Kim, "Compressed sensing for wireless communications: Useful tips and tricks," *IEEE Commun. Surveys Tuts.*, vol. 19, no. 3, pp. 1527–1550, 3rd Quart. 2017.

[10] B. Shim, S. Kwon, and B. Song, "Sparse detection with integer constraint using multipath matching pursuit," *IEEE Commun. Lett.*, vol. 18, no. 10, pp. 1851–1854, Oct. 2014.

[11] B. Shim and B. Song, "Multiuser detection via compressive sensing," *IEEE Commun. Lett.*, vol. 16, no. 7, pp. 972–974, Jul. 2012.

[12] H. Zhu and G. B. Giannakis, "Exploiting sparse user activity in multiuser detection," *IEEE Trans. Commun.*, vol. 59, no. 2, pp. 454–465, Feb. 2011.

[13] B. Knoop, T. Wiegand, and S. Paul, "Low-complexity and approximative sphere decoding of sparse signals," in *Proc. 46th Asilomar Conf. Signals, Syst. and Comput.*, Nov. 2012, pp. 1241–1244.

[14] S. Barik and H. Vikalo, "Sparsity-aware sphere decoding: Algorithms and complexity analysis," *IEEE Trans. Signal Process.*, vol. 62, no. 9, pp. 2212–2225, May 2014.

[15] B. Knoop, F. Monsees, C. Bockelmann, D. Peters-Drolshagen, S. Paul, and A. Dekorsy, "Compressed sensing K-best detection for sparse multi-user communications," in *Proc. 22nd Eur. Signal Process. Conf.*, Lisbon, Portugal, 2014, pp. 1726–1730.

[16] B. Knoop, F. Monsees, C. Bockelmann, D. Wuebben, S. Paul, and A. Dekorsy, "Sparsity-aware successive interference cancellation with practical constraints," in *Proc. 17th Int. ITG Workshop Smart Antennas*, Stuttgart, Germany, Mar. 2013, pp. 1–8.

[17] D. Wubben, R. Bohnke, V. Kuhn, and K. D. Kammeyer, "MMSE extension of V-BLAST based on sorted QR decomposition," in *Proc. IEEE 58th Veh. Technol. Conf.*, vol. 1, Orlando, FL, USA, Oct. 2003, pp. 508–512.

[18] Y. Liu, C. Yuen, X. Cao, N. U. Hassan, and J. Chen, "Design of a Scalable Hybrid MAC Protocol for Heterogeneous M2M Networks," *IEEE Internet Things J.*, vol. 1, no. 1, pp. 99–111, Feb. 2014.

[19] Å. Björck, "Numerics of gram-schmidt orthogonalization," *Linear Algebra Appl.*, vol. 197-198, pp. 297–316, Jan./Feb. 1994.

[20] S. Park, H. Seo, H. Ji, and B. Shim, "Joint active user detection and channel estimation for massive machine-type communications," in *Proc. IEEE 18th Workshop Signal Process. Advances in Wireless Commun.*, Hokkaido, Japan, Jul. 2017.

[21] B. Knoop, S. Schmale, D. Peters-Drolshagen, and S. Paul, "Activity and channel estimation in multi-user wireless sensor networks," in *Proc. 20th Int. ITG Workshop Smart Antennas*, Munich, Germany, Mar. 2016, pp. 1–5.

[22] Z. Chen, F. Sohrabi, and W. Yu, "Sparse activity detection for massive connectivity," *IEEE Trans. Signal Process.*, vol. 66, no. 7, pp. 1890–1904, Apr. 2018.

[23] G. Hannak, M. Mayer, A. Jung, G. Matz, and N. Goertz, "Joint channel estimation and activity detection for multiuser communication systems," in *Proc. IEEE Int. Conf. Commun. Workshop*, London, England, Jun. 2015, pp. 2086–2091.

[24] X. Xu, X. Rao, and V. K. N. Lau, "Active user detection and channel estimation in uplink CRAN systems," in *Proc. IEEE 51th Int. Conf. Commun. Workshop*, London, England, Jun. 2015, pp. 2727–2732.

[25] M. Opper and O. Winther, "Gaussian processes for classification: Mean-field algorithms," *Neural computation*, vol. 12, no. 11, pp. 2655–2684, Nov. 2000.

[26] T. P. Minka, "Expectation propagation for approximate bayesian inference," in *Proc. 17th Conf. Uncertain. Artif. Intell.*, Seattle, WA, USA, Aug. 2001, pp. 362–369.

[27] M. R. Andersen, O. Winther, and L. K. Hansen, "Bayesian inference for structured spike and slab priors," in *Proc. 27th Adv. in Neural Inform. Process. Syst.*, Montreal, QC, Canada, Dec. 2014, pp. 1745–1753.

[28] J. M. Hernández-Lobato, D. Hernández-Lobato, and A. Suárez, "Expectation propagation in linear regression models with spike-and-slab priors," *Machine Learning*, vol. 99, no. 3, pp. 437–487, Jun. 2015.

[29] S. Kullback, *Information theory and statistics*. Courier Corporation, 1997.

[30] C. M. Bishop, *Pattern Recognition and Machine Learning*. Secaucus, NJ, USA: Springer, 2006.

[31] P. Bromiley, "Products and convolutions of Gaussian probability density functions," *Tina-Vision Memo*, vol. 3, no. 4, p. 1, Aug. 2003.

[32] W. W. Hager, "Updating the inverse of a matrix," *SIAM review*, vol. 31, no. 2, pp. 221–239, Jun. 1989.

[33] D. P. Wipf and B. D. Rao, "Sparse Bayesian learning for basis selection," *IEEE Trans. Signal Process.*, vol. 52, no. 8, pp. 2153–2164, Aug. 2004.

[34] M. E. Tipping, "Sparse bayesian learning and the relevance vector machine," *J. Mach. Learn. Res.*, vol. 1, no. Jun, pp. 211–244, Jun. 2001.

[35] S. Ji, Y. Xue, and L. Carin, "Bayesian compressive sensing," *IEEE Trans. Signal Process.*, vol. 56, no. 6, pp. 2346–2356, Jun. 2008.

[36] S. Ji, D. Dunson, and L. Carin, "Multitask compressive sensing," *IEEE Trans. Signal Process.*, vol. 57, no. 1, pp. 92–106, Jan. 2009.

초 록

대규모 사물 통신(massive machine-type communications, mMTC)은 다양한 사물 인터넷(internet of things, IoT) 서비스를 지원하기 위해 차세대 무선 통신 표준에 새로 도입된 서비스 범주이다. 대규모 사물 통신 환경에서는 많은 수의 사물 기기(machine-type device, MTD)가 대부분의 타임 슬롯(time slot)에서 비활성 상태이며 데이터를 전송하지 않는다. 따라서, 활성 및 비활성 기기 모두의 데이터 심볼로 구성된 전송 벡터는 희소(sparse) 벡터로 표현될 수 있다. 희소 벡터로 표현된 다중 사용자 벡터를 복원하기 위해, 압축 센싱 기반 다중 사용자 검출(CS-MUD)이 사용될 수 있다. CS-MUD는 스케줄링(scheduling) 절차가 없는 상향링크(uplink) 비직교 다중 접속(non-orthogonal multiple access, NOMA)을 위한 핵심 기술 중 하나이다. 본 학위 논문에서는 대규모 사물 통신을 위한 새로운 CS-MUD 기술들을 제안한다.

논문의 첫 번째 부분에서는, 희소성을 고려한 정렬 순차적 간섭 제거(sparsity-aware ordered successive interference cancellation, SA-OSIC) 기술을 제안한다. CS-MUD에서 다중 사용자 벡터는 희소성을 고려한 최대 사후 확률(sparsity-aware maximum a posteriori probability, S-MAP) 기준에 따라 검출된다. S-MAP 검출의 계산 복잡성을 줄이기 위해 희소성을 고려한 순차적 간섭 제거(sparsity-aware successive interference cancellation, SA-SIC)를 사용할 수 있습니다. 희소 데이터 벡터 검출의 계산 복잡성을 줄이기 위해 사용되는 희소성 고려 연속 간섭 제거 기술은 오류 전파(error propagation)로 인해 적절한 사용자 정렬 없이는 성능이 좋지 않다. 다중 사용자 벡터가 희소 벡터이고 각 기기가 다른 확률로 활성일 경우 채널 이득(channel gain)에 의해서만 결정된 사용자 검출 순서는 최적이 아닐 수 있다. 본 논문에서는,

오류 전파를 줄이고 희소성 고려 연속 간섭 제거의 성능을 향상하기 위해 각 사물 기기의 활성 확률(activity probability)과 채널 이득을 기반으로 최적의 검출 순서를 찾는 새로운 알고리즘을 제안한다. 시뮬레이션을 통해 제안한 검출 순서 정렬 기술은 데이터 검출의 성능을 크게 향상함을 검증하였다.

논문의 두 번째 부분에서는, 기댓값 전파 기반 활성 사용자 검출 및 채널 추정(expectation propagation based active user detection channel estimation, EP-AUD/CE) 기술을 제안한다. CS-MUD에 관한 몇몇 연구에서, 각 사물 기기로부터 기지국(base station, BS)으로의 상향링크 채널 상태 정보(channel state information, CSI)는 기지국에 완전히 알려져 있다고 가정된다. 그러나 실제로는, 데이터 검출 전에 각 기기로부터 기지국으로의 상향링크 채널 상태 정보를 추정해야 한다. 이 문제를 해결하기 위해 다양한 활성 사용자 검출(active user detection, AUD) 및 채널 추정(channel estimation, CE) 기술이 제안되었다. 대규모 사물 통신에서는 하나의 타임 슬롯에 적은 수의 장치만 활성화되기 때문에 이진(binary)값으로 이루어진 활성 여부 벡터와 채널 벡터의 곱은 희소 벡터가 되어 압축센싱 알고리즘으로 복원이 가능하다. 하지만, 이러한 연구들의 단점 중 하나는 희소 벡터의 사전 분포(prior distritubion)가 활용되지 않는다는 것이다. 희소 벡터의 통계적 사전 분포를 이용하면 활성 사용자 검출 및 채널 추정의 성능을 크게 향상할 수 있다. 본 학위 논문에서는, 기댓값 전파(expectation propagation, EP) 알고리즘을 이용해 희소 채널 벡터의 사후 분포(posterior distribution)의 근사 분포를 찾고, 해당 근사 분포를 이용하여 활성 사용자 검출과 채널 추정을 동시에 수행하는 기술을 제안한다. 시뮬레이션을 통해 제안한 사용자 검출 및 채널 추정 기술은 활성 사용자 검출 및 채널 추정의 성능을 상당히 향상함을 검증하였다.

주요어: 대규모 사물 통신, 압축 센싱, 비직교 다중 접속, 다중 사용자 검출.
학번: 2012-20805