ON THE ENHANCEMENT OF SIGNALS IN THE PRESENCE OF NOISE

WANG QINGZHENG

PhD

The Hong Kong Polytechnic University

2021

The Hong Kong Polytechnic University
Department of Applied Mathematics

On the enhancement of signals in the presence of noise

Wang Qingzheng

A thesis submitted in partial fulfilment of the requirements
for the degree of Doctor of Philosophy

July 2020

Certificate of Originality

I hereby declare that this thesis is my own work and that, to the best of my knowledge and belief, it reproduces no material previously published or written, nor material that has been accepted for the award of any other degree or diploma, except where due acknowledgement has been made in the text.

_____(Signed)

WANG Qingzheng (Name of student)

Dedicate to my parents.

Abstract

This thesis is concerned with the speech enhancement system in the presence of noisy data. The real word presents us with digital data which contains not only the useful signal but also the noise. Noisy data is referred to the unwanted information-bearing signal conveying information about either the sources of the noise or the environment. Unwanted noisy data often causes the algorithms to miss out the useful signal in the data so as to limit the ability of systems. Noise reduction is a broad term where the goal is to remove the information unrelated to the phenomenon we want to study. In this thesis, we resort to tackling the noise reduction problems and seeking the optimal speech enhancement solutions in the present of noisy data.

In the speech communication system, there are two kinds of noise. The first one is the environment noise. The other is the channel noise especially in the wireless communication case. Environment noise can be recognized as the measurement noise which is the result of the imperfection of measurement instruments. The channel noise can be treated as the dynamical noise in the inherent system.

Since speech data are recorded by pre-mounted microphones, one of the main problems when dealing with the speech data is that the microphones will most likely record noisy speech due to the interference of the environment and background noises. In order to extract the pure speech and suppress the noises, there are two popular methods including the beamforming technique and blind source separation technique. In this thesis, we firstly design a novel distributed acoustic beamformer with

blockchain protection such that the interference signals and background noise can be suppressed. After that, we try to improve the performance of the blind source separation system via the optimization of sensor placement in the wireless acoustic sensor network. It observes significant advantages of our proposed methods according to the simulation results.

In order to deal with the channel noise, we investigate the optimization methods in terms of the M-QAM constellation in additive white noise channel. In digital communication systems, channel coding is an extremely important task in determining the performance of the system. In implementing such systems, one needs to map a sequence of bits to symbols in the constellation. A good mapping can show a significant difference in the ultimate performance of the designed system in terms of the error rates. Using the technique of the assignment problem, we derive an optimal mapping rule for the Honeycomb-structured constellation such that the bit-error rate is significantly reduced. Apart from that, we also investigate the optimal position of symbols in the constellation. In order to speed up the optimization process, we propose a novel calculation method of the bit error rate (BER) and the tailor-made optimization method is also proposed. Simulation results show that our proposed BER calculation method is accurate and an optimal mapping rule can be achieved using our tailor-made optimization method.

Acknowledgements

The endeavor of carrying out research is a fascinatingly non-isolated activity. I am grateful to the several individuals who have supported me in various ways during the Ph.D. program and would like to hereby acknowledge their assistance.

First and foremost, I wish to express my heartfelt appreciation to my supervisor, Professor Yiu Ka-fai Cedric, for his enlightening guidance, invaluable discussions, insightful ideas and great patience throughout these years. What I have benefited most from him is the rigorous and diligent attitude to scientific research.

I would like to extend my gratitude to Professor Wong Heung who was my M.Phil. supervisor. Without his encouragement, I would not finish my study in Hong Kong. Furthermore, I shall express my thanks to Dr. Guo Shan, Professor Low Siow-yong, Dr. Li Zhibao and Dr. Wang Hong for their support and help during my research process at the Hong Kong Polytechnic University. I also owe my most sincere gratitude to my wife.

Finally, I would like to express my thanks to my classmates and my friends. Especially, I thank my brother Tang Jing-jung and my sister Au Sim-gyun for their love, encouragement and support in the last seven years.

Contents

Bibliography 129

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Background

Speech enhancement plays an important role in modern digital communication systems. The main objective of speech enhancement is to extract the clear speech as well as to suppress the noise. Noise is referred to the unwanted information-bearing signal conveying information about either the sources of the noise or the environment. For example, there are two teachers in one lecture theater. Teacher A talks about financial time series analysis. Teacher B talks about digital signal processing. When Teacher A and B concurrently offer their lectures, Teacher A is the source of noise for Teacher B and vice versa. In most cases, noise does not provide any help in analyzing our concerned phenomenon. There are two types of noise including measurement noise and dynamical noise. Dynamical noise is the inherent system noise which is caused by the differences between the dynamics of the real and model systems [102]. Measurement noise is the result of the imperfection of measurement instruments [70] and is an additive noise to the system or model [97]. Any digital system involving acquisition, transmission and generation of speech suffers the influence of noise signal that may degrade the performance of the system and deteriorate the quality of the speech signal. Figure 1.1 displays a sketch of modern acoustic communication system. Based on the sites of noise sources, noisy signals can be

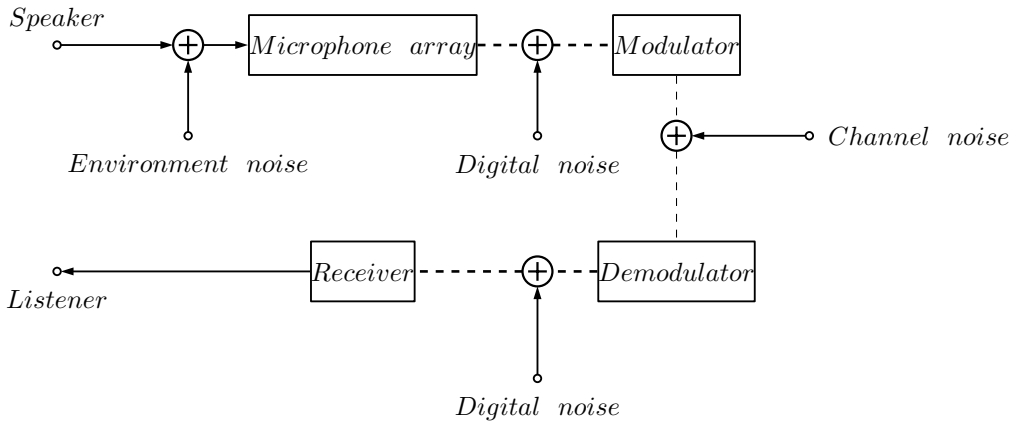


Figure 1.1: Acoustic communication system

categorized as environment noise, digital noise and channel noise.

Environment noises include:

- Interference noise is the unwanted signal from concurrent competitive speakers;
- Background noise is the sound from various ambient sources;
- Reverberation noise is caused by multi-path propagation in an enclosure.

Digital noise is referred to the unwanted signal generated by digital circuits when speech is transferred from analog signal to digital signal. Digital noise may include amplifier noise, quantization noise, round-off noise and so forth. Channel noise refers to any energy interference that affects the quality of signal when signals are transformed from one terminal to another. Channel noise can either be natural such as atmospheric interference or be man-made such as an electrical transformer.

In the presence of noise, many real applications, such as digital signal processing and financial time series analysis, become difficult because some mathematical models have no capability of noise resistance [69, 80] so as to be incompetent. Noise reduction is a ubiquitous topic [15, 63, 38, 94, 93] where the goal is to remove the information which does not provide any help about analyzing our concerned things.

In terms of dealing with noisy data, mathematical models should have the ability to separate the dynamic noise from measurement noise. After that, there are two major problems. One problem is how to capture the dynamic behavior in the noisy data. The other is how to eliminate or mitigate the effects of measurement noises.

This thesis is concerned with the optimization of speech enhancement systems in the presence of noise. This thesis can be split into two main topics. The first topic is the enhancement of received acoustic signals and the reduction of environment noise over a set of acoustic sensors. The second topic is the optimization of communication constellation so as to reduce the probability of transmission error. Chapter 2 and 3 focus on the first topic. Chapter 3 and 4 are on the second topic.

In Chapter 1, we consider the design of distributed acoustic beamforming with blockchain protection in a wireless acoustic sensor network. Nowadays, smart devices flourish everywhere and many of them are networked via various protocols. This enables information exchange to flow between devices via wireless communication. With the availability of inexpensive microphone hardware, ever more devices will be equipped with acoustic sensors. At the same time, we are surrounded by a great deal of noise in real-world environment [64], including additive noise correlated or uncorrelated with the speech signal, and convolutional distortion of the speech signal such as reverberation. Since a microphone array no longer needs to be lined up in a restricted area as the traditional microphone array does but can be placed in any suitable position without being restricted by wires, a newly developed system which is called wireless acoustic sensor network (WASN) [20, 49] can be utilized to enhance the received signal and to reduce noise. When it comes to the designing of beamformer systems in WASN, the key step is to compute the vector of beamforming weight which is the solution of an optimization problem. In this procedure, there are two important aspects. One is to decide the evaluation criterion of the optimization problem, i.e., the objective function. There are commonly two choices:

minimum variance distortionless response (MVDR) [78] and minimum mean square error (MMSE) [84]. While the MVDR beamformer requires an exact steering vector, the MMSE beamformer makes use of reference signals in the design. The other important aspect is to decide the transmission and processing strategy. There are also two typical choices: the centralized beamformer and distributed beamformer. In the centralized beamformer, all raw acoustic signals are collected in a fusion center which conceptually connects to all the acoustic sensors [120]. The fusion center calculates the inverse covariance matrix of acoustic signals so as to derive an optimal beamforming weight. However, the centralized beamformer may not be suitable for the WASN because the total number of microphones is too large to process in a signal device when it is employed in the WASN [13]. Apart from that, the centralized beamformer is limited by the communication bandwidth and transmission power [14]. Moreover, the fusion center could be absent in the WASN due to the uncertain topology of the wireless network [119, 55]. Last but not least, the transmission failure between the fusion center and acoustic sensors cannot be ignored in the WASN. To avoid these shortcomings, the distributed beamformer [42] is widely adopted recently. Since each node in the WASN has its processing unit, it can locally process data and share the results with its neighboring nodes. By cooperating in a distributed fashion, all the nodes share the computational burden. Therefore, we design a distributed MMSE beamformer in which the transmission process is protected by a blockchain technique.

In Chapter 2, we consider a novel optimization method for a blind source separation (BSS) system. When speech signals are recorded by pre-mounted microphones in WASN, it is interest of recovering a set of unobserved signals (signal-of-interest) from several observed mixtures. There are two common popular methods including beamforming and BSS. Beamforming technique as mentioned in Chapter 1 requires geometric information such as the array geometry and the localization information of speech sources. Moreover, beamforming technique is very sensitive to array model

mismatch [12, 75, 28]. With fewer assumptions of the acoustic scene or system, BSS has an upper hand against more conventional beamforming approaches in real world [6, 5]. The term "blind" in BSS system refers to the fact that no information about the array geometry and source localization is needed in its formulation. With the development of indoor GPS techniques, acoustic array geometry can be obtained. We consider improving the performance of BSS system via the optimization of acoustic array geometry.

In Chapter 3, the constellation design of modulations is related to the symbol mapping which is defined as labeling (assigning) constellation symbols with binary digits (bit-sequence). Trellis-coded modulation (TCM) is the first bandwidth-efficient coding approach introduced by [104]. The bit-interleaved coded modulation (BICM) is introduced by [117] and followed by [25]. A wisely mapping scheme can achieve a significant coding gain. For the small constellation, the binary switch algorithm (BSA) can be applied to the constellation design for the BICM. The BSA is a local search algorithm introduced by [116]. To the best of our knowledge, obtaining a suitable mapping for the large constellation is intractable due to high complexity. In this chapter, we want to formulate the constellation design problem as the assignment problem which copes with the allocation of indivisible resources such as assigning n objects to n locations. Usually, assignment problems can be categorized into two groups. One is the linear assignment problem in which n objects are independent of each other. The other is the quadratic assignment problem (QAP) in which the influence caused by the flow or distance between different objects is under consideration. Since the bit error rate (BER) is not only related to the Hamming distance between different bit-sequences but also related to symbol error rate which is mostly depended on the Euclidean distances between different symbols, the constellation design problem can be formulated as the QAP so as to minimize the BER in the digital communication system. Apart from this novel formulation of the

constellation design problem, the heuristic algorithms are also investigated in this chapter such that the global optimal solution is guaranteed.

Since the total BER can be treated as the weighted summation of symbol error rate (SER) where the weight is the Hamming distance between the pair of assigned bit-sequences. Given fixed Hamming distances, moving the position of any symbol changes the probability of symbol decoding error between other symbols. In the noisy channel, the optimization of coordinates can truly reduce SER so as to improve the performance of digital communication systems. In Chapter 4, we formulate the constellation design process as a dynamic optimization problem where the coordinates of symbols are the decision variables and the BER value is the objective function to minimize. Since the classical calculation method of BER is the numerical technique, the calculation burden is heavy especially for a large signal-to-noise ratio (SNR) or a high requirement of accuracy. The tedious calculation slows down our optimization process and may make our optimization method infeasible. Therefore, we investigate an alternative way to approximately compute the SER so as to speed up the calculation process and improve the accuracy of the SER matrix. Numerical results show that the proposed calculation scheme is effective and accurate.

1.2 Contributions of the Thesis

There are four major contributions in this thesis.

Firstly, we design a compound distributed beamformer where nodes are grouped and the system is embedded with blockchain technology to protect the data integrity during transmission. It attempts to provide more possible reliable connections between groups. Simulated experiments show that the distributed beamformer with blockchain protection is able to maintain steady beamforming performance.

Secondly, we optimize the location of each microphone in the wireless acous-

tic sensor network so as to obtain better separated signals. A novel hybrid descent method is proposed for the optimization work. Blind source separation (BSS) method extracts the desired signals from a mixing observed signals. The nonlinear mixing problem in the reverberant environment degrades the performance of BSS model. With the development of the indoor GPS technique, it enables us to enhance the performance of BSS model via the optimization of microphone locations when the nonlinear mixing problem exists. By doing so, spatial information can be fully exploited via an optimized array geometry. Results show that the optimized array consistently yields a greater suppression (> 10dB) across the different reverberation time compared to an unoptimized linear configuration.

Thirdly, we proposed a novel framework where the constellation assignment is formulated as a quadratic assignment problem (QAP) which is also a vital task in the society of combinatorial optimization. Therefore, some universal ways to solve QAP problem could be applied to solve the normal constellation design problem directly. However, some relatively complex constellation design problems can only be converted into QAP problems with variable coefficients and cannot be solved by conventional methods. Therefore, we design a multi-layer search algorithm framework based on simulated annealing (SA) algorithm to solve such problems. Numerical simulations with different settings verify that the proposed model and algorithm are very effective.

Fourthly, we focus on coordinates optimization of constellations and formulate it as a dynamic optimization problem where the coordinates of such constellation are the optimization variables and the BER value is the cost function to minimize. In particular, we propose a novel strategy to compute the BER value accurately and quickly and apply a gradient-based optimization method to accelerate the optimization procedure.

As a result, there are four papers in conjunction with this thesis which are listed

below.

1. Qingzheng Wang, Shan Guo, Ka-fai Cedric Yiu, "Distributed acoustic beamforming with blockchain protection," *IEEE Transactions on Industrial Informatics*, 2020.

2. Qingzheng Wang, Siow Yong Low, Zhibao Li, Ka-fai Cedric Yiu "Sensor placement optimization of blind source separation in a wireless acoustic sensor network," submitted.

3. Shan Guo, Qingzheng Wang, Hong Wang, Ka-fai Cedric Yiu " Coordinates optimization of constellation via gradient-based optimization methods," To be submitted.

4. Shan Guo, Qingzheng Wang, Hong Wang, Ka-fai Cedric Yiu " Optimal assignment of constellations with a novel QAP formulation," to be submitted.

Besides that, Chapter one was presented as a conference talk:

1. Qingzheng Wang and Ka-fai Cedric Yiu, "Speech enhancement via distributed acoustic beamforming," NACA-ICOTA2019, Hakodate, Japan, Augest 26-31, 2019.

1.3 Organization of the Thesis

The rest of this thesis is listed as follows.

- Chapter 2 presents a novel distributed MMSE beamfomer in WASN.

- Chapter 3 introduces an optimization scheme of the blind source separation system via the change of microphone locations.

- Chapter 4 presents a novel optimization scheme for constellation mapping rule in digital communication system.
- Chapter 5 presents a novel optimization method of constellation via the derivation of the optimal position for each symbol.
- Chapter 6 draws conclusions and shows the future work.

Chapter 2

Distributed acoustic beamforming with blockchain protection

Speech is a natural user interface for the Internet of Things system. However, the presence of noise affects severely the performance of such system. With the deployment of smart devices with microphones, one can form a powerful acoustic sensor network to enhance the speech via beamforming techniques. On the other hand, reliability of data transmission also determines the beamforming performance, since faulty data will drift the beamformer steering location randomly. Currently there is no protection scheme for acoustic data transmitted over the wireless network in order to keep steady beamforming performance. In this chapter, we design a compound distributed beamformer where nodes are grouped and the system are embedded with blockchain technology to protect the data integrity during transmission. It attempts to provide more possible reliable connections between groups. Simulated experiments shows that the distributed beamformer with blockchain protection is able to maintain steady beamforming performance.

[1]

[1] This chapter is an extended version of our paper [107].

2.1 Research background

The Internet of Things (IoT) has transformed our daily life in many aspects. By providing a seamless integration of physical objects into the information network [52], this enables information exchange to flow between devices and also allows the devices to be controlled remotely by users via a range of man-to-machine speech interactive systems [35]. People can remotely control the IoT devices using voice commands or natural dialogue [53]. Voice control is an attractive feature that provides a natural design of remote control and becomes the primary user interface for the smart home [81]. However, the interference of the environment and background noises degrade the performance of such devices [106]. In order to have smooth operations, acoustic noise should be suppressed and the required speech to be enhanced.

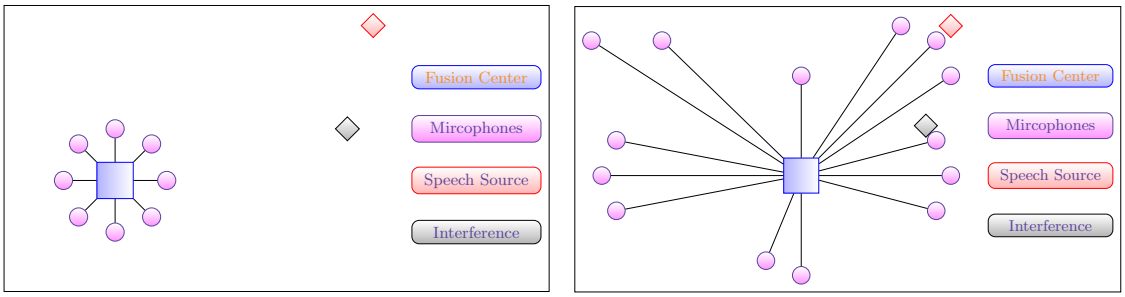
With the advent of wireless smart devices equipped with microphones, a wireless acoustic sensor network (WASN) can be formed and many innovative applications can be developed. One important application is to enhance speech signals and suppress unwanted noise via beamforming techniques [49, 83]. If successful, this can enhance significantly the capability of voice control device. Since the microphone array in WASN no longer needs to be wired in a restricted area as the traditional microphone array does but can be placed in any suitable position, a WASN could accommodate many sensor nodes which are positioned anywhere and each node is allowed to contain a microphone array rather than a single microphone. There are several challenges in developing the distributed beamforming system. First, although the sensor coverage becomes larger and speech signal can be enhanced with the increment of microphone arrays, the increased computational burden cannot be ignored. Second, reliability of data transmission determines the performance of the designed beamformers, since faulty data will drift the filter coefficients quickly and deviated from the target location. In order to enhance robustness and reliability of voice con-

trol system, the transmission reliability via wireless channel should be dealt with. In this chapter, we provide an innovative solution including two major parts:

- A novel beamforming technique which distributes the computational burden over the nodes of the WASN;
- A novel data protection scheme in which blockchain technique ensures the integrity of transmitted data between the nodes of the WASN.

When it comes to the design of beamformer systems, the key step is to compute the vector of beamforming weight which is the solution of an optimization problem. In this procedure, there are two important aspects. One is to decide the evaluation criterion of the optimization problem, i.e., the objective function. There are commonly two choices: minimum variance distortionless response (MVDR) [78] and minimum mean square error (MMSE) [84]. While the MVDR beamformer requires an exact steering vector, the MMSE beamformer makes use of reference signals in the design. The other important aspect is to decide the transmission and processing strategy. There are also two typical choices: the centralized beamformer and distributed beamformer. In the centralized beamformer, all raw acoustic signals are collected in a fusion center which conceptually connects to all the acoustic sensors [120]. The fusion center calculates the inverse covariance matrix of acoustic signals so as to derive an optimal beamforming weight. As shown in Figure 2.1a, the desired sources may far away from the acoustic sensors such that the received signal-to noise ratio is low or the direct-to reverberant ratio is low when reverberation exists. In Figure 2.1b, the coverage area of acoustic sensors is enlarged to overcome the above shortcomings. It is noted that acoustic sensor placement are restricted by the wire connection. Moreover, the computation burden for the fusion center becomes heavier with the increment of the number of sensors. As shown in Figure 2.2a, WASN can

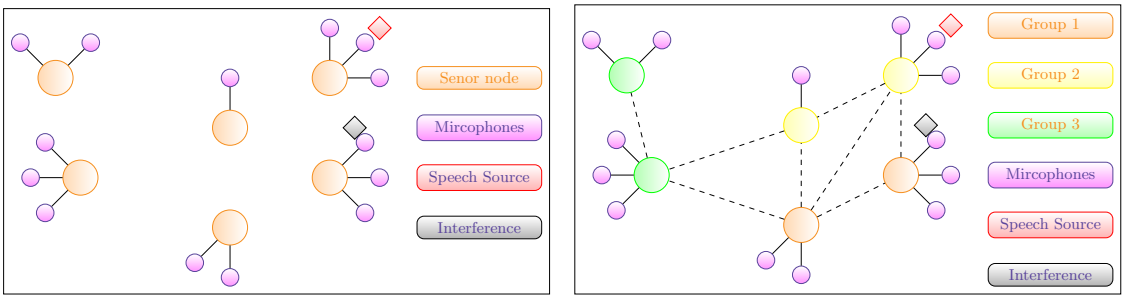
be employed to remove the constriction of the acoustic sensor placement. However, the centralized beamformer may not be suitable for the WASN because the total number of microphones is too large to process in a signal device when it is employed in the WASN [13]. Apart from that, the centralized beamformer is limited by the communication bandwidth and transmission power [14]. Moreover, the fusion center could be absent in the WASN due to the uncertainty topology of the wireless network [119, 55]. Last but not least, the transmission failure between the fusion center and acoustic sensors cannot be ignored in the WASN. To avoid these shortcomings, the distributed beamformer [42], as shown in Figure 2.2b is widely adopted recently. Since each node in the WASN has its own processing unit, it can locally process data and share the results with their neighboring nodes. By cooperating in a distributed fashion, all the nodes share the computational burden.



(a) Centralized Beamformer (Array) (b) Centralized Beamformer (Network)

Figure 2.1: Centralized Beamformer

Although a distributed beamforming system has advantages, it has also created new challenges. One challenge is how to design decentralized schemes so that the computational burden is shared by all the nodes and the entire sampling data is fused iteratively in the WASN. A good decentralized scheme should reduce the wireless data transmission and share the computational burden between nodes. A newly proposed strategy of data transmission for distributed beamformers is called the gos-



(a) WASN (b) Distributed beamformer

Figure 2.2: WASN and distributed Beamformer

sip algorithm where information exchanges between adjacent nodes constantly and successively [119]. With successive iterations, this algorithm can reach the consensus solution for each node [110]. There are variants of the gossip algorithm and will be discussed in detail in Section III. The gossip algorithm has several advantages. First of all, since only one node will communicate with one of its neighbors at each iteration, it is computationally efficient. Second, the algorithm does not require the WASN to remain to be the same throughout the whole process, but allowing new links to append and old ones to exit. However, there exist challenges in the implementation of gossip algorithms. When the number of nodes increases, the number of iterations required to reach convergence will increase rapidly [40]. To cope with this drawback, our approach only employs the gossip method among groups to trade off the increasing complexity. At application level, convergence of the algorithms can be slowed down significantly due to faulty transmission caused by the unstable links in the WASN [60], which in turns degrades the overall performance of the system. Moreover, it is necessary to deal with the consensus issue of nodes inside same groups caused by faulty transmission. Thus, a data protection mechanism is very important for the sake of data integrity.

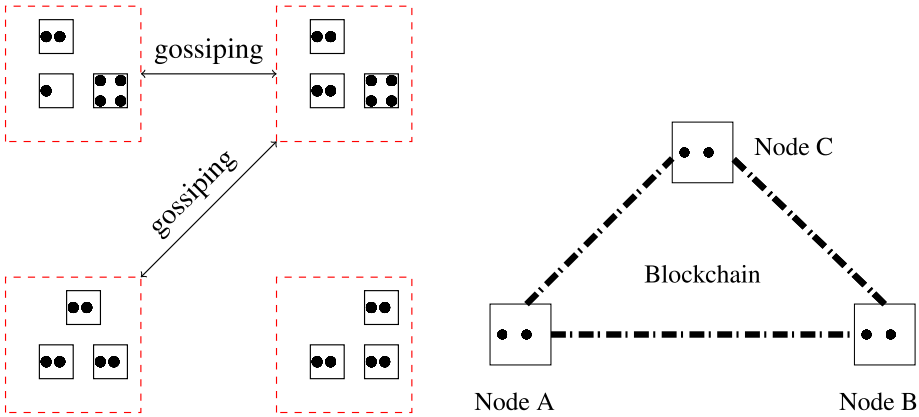
For multimedia applications, transmission is often carried out via protocols like

user datagram protocol (UDP) [121] to increase efficiency; on the other hand, sacrificing data integrity with less verification [91]. Unstable wireless links may yields heavy packet loss. In order to retain transmission reliability, it has must be carried out within the application level. First of all, corrupted data should be rejected or discarded by receivers. There are various ways of detecting faulty transmission. The simplest method is Cyclic Redundancy Check (CRC) but it is limited by its error detection capability [61]. A more elaborated method is called Message Authentication Code (MAC) based on a cryptographic-based algorithm [88, 27]. A lightweight data integrity checking method is to use the watermarking technique upon sensor networks [59, 65]. However, most of existing integrity mechanisms for wireless networks require a base station (or fusion center) which is likely absent for most distributed wireless networks. In addition, this centralized architecture has certain inherent vulnerabilities. For example, the whole system stops working if the base station is down due to maintenance or software failures [46]. Furthermore, the aforementioned methods only detect the corruption of transmitted data rather than improving the data integrity. Transmission Control Protocol (TCP) has been extended and adapted to be deployed in wireless sensor network so that data can be retransmitted to improve the data integrity [19]. However, it relies on the original established links. If one wireless link has become unreliable, correct data still cannot be obtained by the retransmission request.

In view of the above, here we design a data protection scheme at application level for the WASN using blockchain technique. We propose a novel framework of the compound distributed beamformer where nodes are grouped and the system are embedded with blockchain technology [89] to protect the data during transmission. The distributed MMSE beamforming algorithm is developed. The sketch of the basic idea is shown in Figure 2.3, where small black rectangles represent nodes in the WASN, and each node could contain several microphones which are represented

by black dots. WASN is divided into groups which are represented by the dotted red line according to certain preset rules. This is a two-level communication scheme containing the intra-group data sharing based on the blockchain technique as well as the inter-group data communication via gossip algorithms. We first share data within each group and use the blockchain technique to protect the fused data as well as to resolve the consensus problem inside the group. Using the hash function, another group can easily verify the correctness of the received data in inter-group communications. In our proposed framework, we actually can randomly select any node in the group to establish the wireless link for data transmission between groups, since any selected node will have the same data within the group. If the selected link has a problem, we can immediately switch to another link so that we will not lose access to the entire group when one wireless link becomes unreliable. In our proposed framework, the connectivity reliability between groups is enhanced by providing more than one wireless link such that the possibility of faulty transmission is decreased.

Blockchain is a distributed storage system in which data is stored in a decentralized network as blocks and updated using an append-only structure. After the first introduction in 2008 by Satoshi Nakamoto, blockchain is growing with fast popularity [36]. It has been employed successfully in cryptocurrency and some other industries as well. Optimizations of blockchain have been conducted in the resource constrained environment [46]. In the design of the blockchain implementation for beamforming, we need to consider two important aspects including the reduction of computational complexity and the restriction of ledger scalability. In order to reduce the computational complexity, the distributed trust method is employed here to replace proof-of-work [43], since it decreases new block processing overhead while maintaining most of its security benefits. To deal with the problem of scalability, we first employ the short-time Fourier transform to locally compress raw acoustic signals in each node. It reduces the requirement of memory and bandwidth in the



(a) Gossiping among groups

(b) Blockchain structure of the left bottom group in (a)

Figure 2.3: Sketch of the proposed framework

system. Second, we create new ledgers for each time frame and delete them after the computation of beamforming weights. Therefore, the length of ledger is bounded by the maximum iteration of gossip algorithms.

The rest of the chapter is organized as follows. The problem formulation is given in Section 2.2. The distributed computation scheme is introduced in Section 2.3. The data protection based on blockchain technique is illustrated in Section 2.4. The simulation study is demonstrated in Section 2.5. The discussion on experimental results is presented in Section 2.6. Conclusion and further work are shown in Section 2.7.

2.2 Notation and Problem formulation

In this work we consider an enclosed room with acoustic reverberation. In this room, N speech sources are settled at $\gamma_n, n = 0, \ldots, N-1$ and M-elements microphone array settled at $\delta_m, m = 1, \ldots, M$. The M microphones are grouped in U nodes. For the u-th node, it contains M_u microphones such that $M = \sum_{u=1}^{U} M_u$. In addition, we divide all the nodes into V groups. Each group contains U_v nodes with $U = \sum_{v=1}^{V} U_v$.

Without loss of generality, the sensor at γ_0 is denoted as the signal of interest; the others are interferences; and the microphone at δ_1 is the reference microphone. Here, the noise placement information is not considered in beamformer design. Given the room dimension, sound speed, locations of sources and microphones, the time domain room impulse responses (RIR) $\mathbf{h}(\delta_m, \gamma_n)$ from the n-th source to the m-th microphone can be generated by the image method [72]. Let $\mathbf{s}_n$ denote the signal at the source γ_n. The received signal at microphone δ_m is calculated by

$$\mathbf{s}_{m,n} = \mathbf{h}(\delta_m, \gamma_n) \odot \mathbf{s}_n \tag{2.1}$$

where $\odot$ denotes the convolution operator. By the short-time Fourier transform (STFT), the frequency domain coefficient of the m-th microphone is given by

$$Y_m(f,k) = S_{m,0}(f,k) + \sum_{n=1}^{N-1} S_{m,n}(f,k) + N_m(f,k) \tag{2.2}$$

where $S_{m,n}(f,k)$ is the STFT coefficient of $\mathbf{s}_{m,n}$ at frequency-bin index f and time-frame index k. The target source is $S_{m,0}(f,k)$. The interference sources are $\sum_{n=1}^{N-1} S_{m,n}(f,k)$. The noise STFT coefficient of m-th microphone is denoted by $N_m(f,k)$. Let $\mathbf{Y}(f,k) = [Y_1(f,k),\ Y_2(f,k),\ \ldots,\ Y_M(f,k)]^{\mathtt{T}}$ and $\mathbf{S}(f,k) = [S_{1,0}(f,k),\ S_{2,0}(f,k),\ \ldots,\ S_{M,0}(f,k)]^{\mathtt{T}}$. Here, $\mathbf{Y}(f,k)$ is the input data of the beamformer in operation phase, and $\mathbf{S}(f,k)$ is the target signal. Let $w_m(f)$ be the beamforming weight of the m-th microphone at frequency f. The weight vector of the beamformer with frequency f is denoted as $\mathbf{w}(f) = [w_1(f)\ w_2(f)\ \cdots\ w_M(f)]^{\mathtt{T}}$. At the time-frame k and frequency f, the output of the beamformer in the frequency domain is defined by

$$\tilde{Y}(f,k) = \sum_{m=1}^{M} w_m(f) Y_m(f,k) = \mathbf{w}(f)^{\mathtt{H}} \mathbf{Y}(f,k). \tag{2.3}$$

The problem of MMSE beamformer [84] can be recognized as the least square

optimization problem which is formulated as

$$\mathbf{w}_{\mathtt{opt}}(f) = \underset{\mathbf{w}(f)}{\arg\min}\ \mathbb{E}\left\{|\tilde{Y}(f,k) - S_r(f,k)|^2\right\} \tag{2.4}$$

where $|\cdot|$ denotes the absolute value and $\mathbb{E}$ denotes the expectation operator, and $S_r(f,k)$ denotes the STFT coefficient related to the observation from the reference microphone. Let K_1 and K_2 denote the time-frame length in the optimization and operation phase respectively. Assume the reference signal $S_r(f,k)$ is independent of the actual observation $\mathbf{Y}(f,k)$, substituting Equation (2.3) into Equation (2.4), the original problem can be expressed as

$$\begin{aligned}
\mathbf{w}_{\mathtt{opt}}(f) &= \underset{\mathbf{w}(f)}{\arg\min}\ \left\{\sum_{k=0}^{K_1-1}\left[|\mathbf{w}(f)^{\mathtt{H}}\mathbf{S}(f,k) - S_r(f,k)|^2\right]\right. \\
&\qquad \left. + \sum_{k=0}^{K_2-1}|\mathbf{w}(f)^{\mathtt{H}}\mathbf{Y}(f,k)|^2\right\} \\
&= \underset{\mathbf{w}(f)}{\arg\min}\ \left\{\mathbf{w}(f)^{\mathtt{H}}[\hat{\mathbf{R}}_{SS}(f,K_1) + \hat{\mathbf{R}}_{YY}(f,K_2)]\mathbf{w}(f)\right. \\
&\qquad \left. - \mathbf{w}(f)^{\mathtt{H}}\hat{\mathbf{r}}_s(f,K_1) - \hat{\mathbf{r}}_s^{\mathtt{H}}(f,K_1)\mathbf{w}(f) + \hat{r}_{s_r}\right\}
\end{aligned}$$

where $\hat{r}_{s_r}$ is the variance of the reference microphone which can be treated as a constant in this optimization problem. In the optimization phase, the estimated correlation matrix $\hat{\mathbf{R}}_{SS}(f,K_1)$ and cross correlation vector $\hat{\mathbf{r}}_s(f,K_1)$ are calculated by

$$\hat{\mathbf{R}}_{SS}(f,K_1) = \frac{1}{K_1}\sum_{k=0}^{K_1-1}\mathbf{S}(f,k)\mathbf{S}(f,k)^{\mathtt{H}}, \tag{2.5}$$

$$\hat{\mathbf{r}}_s(f,K_1) = \frac{1}{K_1}\sum_{k=0}^{K_1-1}\mathbf{S}(f,k)S_r(f,k)^* \tag{2.6}$$

where $S_r(f,k)^*$ is the complex conjugate of $S_r(f,k)$. In the operation phase, the estimated correlation matrix $\hat{\mathbf{R}}_{YY}(f,K_2)$ is defined by

$$\hat{\mathbf{R}}_{YY}(f,K_2) = \frac{1}{K_2}\sum_{k=0}^{K_2-1}\lambda^{K_2-1-k}\mathbf{Y}(f,k)\mathbf{Y}(f,k)^{\mathtt{H}} \tag{2.7}$$

where λ is an exponential weighting factor.

Given the estimates from Equation (2.5)-(2.7), the optimal weight vector of the MMSE beamformer in Equation (2.4) is obtained by

$$\mathbf{w}_{\mathtt{opt}}(f) = \hat{\mathbf{R}}(f,K_2)^{-1}\hat{\mathbf{r}}_s(f,K_1) \tag{2.8}$$

where $\hat{\mathbf{R}}(f,K_2) = \hat{\mathbf{R}}_{SS}(f,K_1) + \hat{\mathbf{R}}_{YY}(f,K_2)$.

In time-frame k of the operation phase, the known information is all the observations in the optimization phase as well as the observations up to time-frame k in the operation phase. Then, we have

$$\hat{\mathbf{R}}(f,k) = \hat{\mathbf{R}}_{SS}(f,K_1) + \hat{\mathbf{R}}_{YY}(f,k)$$

which can be extended to

$$\begin{aligned}
\hat{\mathbf{R}}(f,k) &= \hat{\mathbf{R}}_{SS}(f,K_1) + \hat{\mathbf{R}}_{YY}(f,k) \\
&= \hat{\mathbf{R}}_{SS}(f,K_1) + \lambda\hat{\mathbf{R}}_{YY}(f,k-1) + \mathbf{Y}(f,k)\mathbf{Y}(f,k)^{\mathtt{H}} \\
&= \lambda\hat{\mathbf{R}}(f,k-1) + \mathbf{Y}(f,k)\mathbf{Y}(f,k)^{\mathtt{H}} + (1-\lambda)\hat{\mathbf{R}}_{SS}(f,K_1) \\
&= \lambda\hat{\mathbf{R}}(f,k-1) + \mathbf{Y}(f,k)\mathbf{Y}(f,k)^{\mathtt{H}} \\
&\quad + \sum_{m=1}^{M}(1-\lambda)\gamma_m(f)\mathbf{q}_m(f)\mathbf{q}_m(f)^{\mathtt{H}}
\end{aligned}$$

where $\gamma_m(f)$ is the m-th eigenvalue and $\mathbf{q}_m(f)$ is the m-th eigenvector of the $M \times M$ correlation matrix $\hat{\mathbf{R}}_{SS}(f,K_1)$. With the use of a rank-one approximation of the

matrix [51], $\hat{\mathbf{R}}(f,k)$ can be updated by

$$\begin{aligned}\hat{\mathbf{R}}(f,k) =& \lambda\hat{\mathbf{R}}(f,k-1) + \mathbf{Y}(f,k)\mathbf{Y}(f,k)^{\mathtt{H}} \\ &+ (1-\lambda)\gamma_i(f)\mathbf{q}_i(f)\mathbf{q}_i(f)^{\mathtt{H}}\end{aligned} \tag{2.9}$$

where $i = (k \ \mathtt{mod}\ M) + 1$.

Using the Matrix Inversion Lemma [10] twice, the inverse correlation matrix $\hat{\mathbf{R}}(f,k)^{-1}$ can be computed iteratively

$$\hat{\mathbf{R}}(f,k)^{-1} = \check{\mathbf{R}}(f,k) - \frac{\gamma_i(f)(1-\lambda)\check{\mathbf{R}}(f,k)\mathbf{q}_i(f)\mathbf{q}_i(f)^{\mathtt{H}}\check{\mathbf{R}}(f,k)}{1+\gamma_i(f)(1-\lambda)\mathbf{q}_i(f)^{\mathtt{H}}\check{\mathbf{R}}(f,k)\mathbf{q}_i(f)} \tag{2.10}$$

where

$$\begin{aligned}\check{\mathbf{R}}(f,k) &= \lambda^{-1}\hat{\mathbf{R}}(f,k-1)^{-1} \\ &- \frac{\lambda^{-2}\hat{\mathbf{R}}(f,k-1)^{-1}\mathbf{Y}(f,k)\mathbf{Y}(f,k)^{\mathtt{H}}\hat{\mathbf{R}}(f,k-1)^{-1}}{1+\lambda^{-1}\mathbf{Y}(f,k)^{\mathtt{H}}\hat{\mathbf{R}}(f,k-1)^{-1}\mathbf{Y}(f,k)}.\end{aligned} \tag{2.11}$$

In order to reduce the influence of the random environmental noise, a first order autoregressive smoothing model is used to iteratively update the weight vector of the beamformer as

$$\mathbf{w}^{\mathtt{k}}(f) = \alpha\mathbf{w}^{\mathtt{k-1}}(f) + (1-\alpha)\hat{\mathbf{R}}(f,k)^{-1}\hat{\mathbf{r}}_s(f,K_1) \tag{2.12}$$

where $\alpha \in (0,1)$ is the smoothing parameter. Therefore, in the operation phase, the output of the MMSE beamformer at time-frame k and frequency f is $\mathbf{w}^{\mathtt{k}}(f)^{\mathtt{H}}\mathbf{Y}(f,k)$.

2.3 Distributed Computation Scheme

In this section, we design a distributed computation scheme of the MMSE beamformer in which the nodes in the WASN are divided into different groups. In our distributed computation scheme, gossip algorithms are used to solve the consensus problem among groups. For different grouping rules, the gossip algorithm with same

number of groups may have different convergence speed. In [40], it was suggested that the convergence time of the gossip algorithm depends on the spectral gap of the graph which consists of groups in our case. Taking the network topology into consideration, one can group the nodes with a larger spectral gap so as to speech up the convergence of the gossip algorithm. If the network topology is fixed, one can minimize the convergence time by optimizing the pairwise gossiping probabilities [110]. In practice, we allocate the geographic adjacent nodes into the same group because nodes in the same group need to synchronize the status of private ledgers when gossip algorithms are applied. Apart from that, it is also necessary to consider the size of group since the number of duplication ledgers is increased with the increment of number of nodes in one group, although less groups speed up the convergence of the algorithm.

2.3.1 Gossip algorithms

Gossip algorithms are widely used to solve the average consensus problem in decentralized network systems. The randomized gossip algorithm has been used to design a distributed delay-and-sum beamformer without the consideration of transmission failure [119] . They allow nodes exchanging information peer-to-peer and updating the parameter by computing the pairwise average. Eventually, all the nodes in the network agree on the value of the parameter. There are several variants of the gossip algorithms. The main difference between them is the choice of neighbors or routings. We introduce two distributed computation algorithms: the randomized gossip algorithm and greedy gossip algorithm. In the randomized gossip algorithm, the neighbor is chosen uniformly at random from a predefined neighbor set. It has been shown, in [17], that this algorithm converges to a consensus if the graph is strongly connected. Like other greedy algorithms, the greedy gossip algorithm [105] makes an optimal choice among neighbors to achieve a fast convergence. In this chapter,

the selection criterion of the greedy gossip algorithm is defined as

$$v_2 = \arg\max_{v_2 \in \mathcal{N}_{v_1}} \|\mathbf{x}_{v_1} - \mathbf{x}_{v_2}\|$$

where v_1 is a random chosen group; $\mathcal{N}_{v_1}$ is a predefined neighbor set of Group v_1; v_2 is the chosen neighbor of Group v_1; and $\|\mathbf{x}_{v_1} - \mathbf{x}_{v_2}\| = \sqrt{(\mathbf{x}_{v_1} - \mathbf{x}_{v_2})^{\mathtt{H}}(\mathbf{x}_{v_1} - \mathbf{x}_{v_2})}$ is the Euclidean norm between complex vectors $\mathbf{x}_{v_1}$ and $\mathbf{x}_{v_2}$. It means that the greedy gossip algorithm always chooses the neighbor with the most different value. Comparing with the randomized gossip algorithm, the greedy gossip algorithm accelerates the convergence to a consensus state in the network. However, an additional bandwidth is needed to eavesdrop the information from neighbors. Therefore, both gossip algorithms are investigated in the simulation study.

2.3.2 Distributed Computation of MMSE beamformer

Our objective in this subsection is to estimate the $\check{\mathbf{R}}(f,k)$ in Equation (2.11) distributively but a consensus should be achieved in the network. With the estimation of $\check{\mathbf{R}}(f,k)$ as well as other estimations estimated in the time frame $k-1$ and the optimization phase, it is easy to calculate the optimal beamformer weight in Equation (2.12) so as to derive the output of MMSE beamformer $\tilde{Y}(f,k)$ in Equation (2.3).

We rewrite Equation (2.11) as

$$\check{\mathbf{R}}(f,k) = \lambda^{-1}\hat{\mathbf{R}}(\mathbf{f},\mathbf{k}-\mathbf{1})^{-1} - \frac{\lambda^{-2}\mathbf{a}\mathbf{a}^{\mathbf{H}}}{1+\lambda^{-1}b} \tag{2.13}$$

where

$$\begin{aligned} \mathbf{a} &= \hat{\mathbf{R}}(f,k-1)^{-1}\mathbf{Y}(f,k), \\ b &= \mathbf{Y}(f,k)^{\mathrm{H}}\hat{\mathbf{R}}(f,k-1)^{-1}\mathbf{Y}(f,k) = \mathbf{Y}(f,k)^{\mathrm{H}}\mathbf{a}. \end{aligned}$$

In the sequel, the gossip algorithm is applied to estimate both $\mathbf{a}$ and b sequentially.

For the v-th group, let $\hat{\mathbf{R}}_v(f,k-1)^{-1}$ be the local estimate of $\hat{\mathbf{R}}(f,k-1)^{-1}$ and $\mathbf{c}_{v,1},\ldots,\mathbf{c}_{v,N}$ be the columns of $\hat{\mathbf{R}}_v(f,k-1)^{-1}$. Let M_v be the set of microphones belonging to the v-th group. Without communication to other groups, we can compute a part of $\mathbf{a}$ in the v-th group by

$$\mathbf{a}^{(v)} = \sum_{i\in M_v} \mathbf{c}_{v,i} Y_i(f,k). \tag{2.14}$$

Since

$$\mathbf{a} = \sum_{v=1}^{V}\sum_{i\in M_v} \mathbf{c}_{v,i} Y_i(f,k) = \sum_{v=1}^{V} \mathbf{a}^{(v)} = \frac{1}{V}\sum_{v=1}^{V} \tilde{\mathbf{a}}^{(v)}$$

where V is the number of groups and $\tilde{\mathbf{a}}^{(v)} = V\mathbf{a}^{(v)}$, it can be recognized that $\mathbf{a}$ is the arithmetic mean of $\tilde{\mathbf{a}}^{(v)}$. Therefore, $\mathbf{a}$ can be calculated by the gossip algorithm.

Let $a^{(v_i,t)}$ denote the local estimate of $\mathbf{a}$ in Group v_i which are selected to exchange information at the t-th time by the gossip algorithm. The initial value of the local estimate of $\mathbf{a}$ in Group v_i is defined as

$$\mathbf{a}^{(v_i,0)} = \tilde{\mathbf{a}}^{(v_i)}. \tag{2.15}$$

In one iteration of the gossip algorithm in which Group v_i is selected at the t_i-th time and Group v_j is selected at the t_j-th time, the local estimate of $\mathbf{a}$ in Group v_i and v_j are updated by

$$\mathbf{a}^{(v_i,t_i)} = \mathbf{a}^{(v_j,t_j)} = \frac{1}{2}\left(\mathbf{a}^{(v_i,t_i-1)} + \mathbf{a}^{(v_j,t_j-1)}\right). \tag{2.16}$$

Let $T_{\mathbf{a},v}$ be the number of times that the v-th Group is selected by the gossip algorithm when $\check{\mathbf{R}}(f,k)$ is estimated. The final local estimate of $\hat{\mathbf{R}}(f,k-1)^{-1}\mathbf{Y}(f,k)$ in the v-th Group is

$$\mathbf{a}^{(v,T_{\mathbf{a},v})} = [a_1^{(v,T_{\mathbf{a},v})},\ldots,a_M^{(v,T_{\mathbf{a},v})}].$$

After the estimation work of $\mathbf{a}$, we can estimate b using the same method. Without communication to other groups, we can compute a part of b in the v-th Group by

$$b^{(v)} = \sum_{i \in M_v} Y_i(f,k)^* a_i^{(v,T_{\mathbf{a},v})}. \tag{2.17}$$

Then,

$$b = \sum_{v=1}^{V} \sum_{i \in M_v} Y_i(f,k)^* a_i^{(v,T_v)} = \frac{1}{V} \sum_{i=1}^{V} V b^{(v)} = \frac{1}{V} \sum_{i=1}^{V} \tilde{b}^{(v)}.$$

It is obvious that b is the arithmetic mean of $\tilde{b}^{(v)}$. Therefore, b can be calculated by the gossip algorithm.

Let $b^{(v_i,t)}$ denote the local estimate of b in Group v_i which are selected to exchange information at the t-th time by the gossip algorithm. The initial value of the local estimate of b in Group v_i is defined as

$$b^{(v_i,0)} = \tilde{b}^{(v_i)}. \tag{2.18}$$

For the iteration of the gossip algorithm in which Group v_i is selected at the t_i-th time and Group v_j is selected at the t_j-th time, the local estimate of b in Group v_i and v_j are calculated by

$$b^{(v_i,t_i)} = b^{(v_j,t_j)} = \frac{1}{2}\left(b^{(v_i,t_i-1)} + b^{(v_j,t_j-1)} \right). \tag{2.19}$$

Let $T_{b,v}$ be the number of times that the v-th Group is selected by the gossip algorithm when $\check{\mathbf{R}}(f,k)$ is estimated. The final local estimate of $\mathbf{Y}(f,k)^{\mathrm{H}} \hat{\mathbf{R}}(f,k-1)^{-1} \mathbf{Y}(f,k)$ in the v-th Group is $b^{(v,T_{b,v})}$.

To summarize, in the gossip algorithm, the local estimates of $\mathbf{a}$ and b have exchanged among groups iteratively. The main purpose of the exchange is to calculate the matrix $\check{\mathbf{R}}(f,k)$. After sufficient exchanges, the local estimates of $\check{\mathbf{R}}(f,k)$ converge to the same matrix because $\mathbf{a}$ and b are converge to the same vector and scalar

respectively. The algorithm can be summarized in Algorithm 1. Using the local estimate of $\check{\mathbf{R}}(f,k)$, a local beamformer weight vector is derived. The local output of MMSE beamformer in each group is derived using Equation (2.10), (2.12) and (2.3). It is noted that $\mathbf{Y}(f,k)$ is partially unknown for an individual group but its local estimate in the v-th group can be computed by

$$\mathbf{Y}_v(f,k) = [\hat{\mathbf{R}}_v(f,k-1)^{-1}]^{-1}\mathbf{a}^{(v,T_{a,v})}. \tag{2.20}$$

Algorithm 1 Estimate $\check{\mathbf{R}}(f,k)$ using the gossip algorithm

1: Initialize $\mathbf{a}^{(v,0)}$ for Group v using Equation (2.14) and (2.15), where $v = 1,\ldots,V$, and let $T = 0$.

2: Select two groups and update the local estimates of $\mathbf{a}$ using Equation (2.16).

3: $T = T+1$. $T > T_{\mathbf{a},\max}$ where $T_{\mathbf{a},\max}$ is the hyperparamter denoting the maximum iteration number when $\mathbf{a}$ is estimated.

4: Initialize $b^{(v,0)}$ for Group v using Equation (2.17) and (2.18), where $v = 1,\ldots,V$, and let $T = 0$.

5: Select two groups and update the local estimates of b using Equation (2.19).

6: $T = T+1$. $T > T_{b,\max}$ where $T_{b,\max}$ is the hyperparamter denoting the maximum iteration number when b is estimated.

7: Calculate $\check{\mathbf{R}}_v(f,k)$, the local estimate of $\check{\mathbf{R}}(f,k)$ in the v-th group, by the substitution of $\mathbf{a}^{(v,T_{\mathbf{a},v})}$ and $b^{(v,T_{b,v})}$ into Equation (2.13).

2.3.3 Convergence of the Gossip algorithm

In each iteration of the Gossip algorithm, a pair of nodes exchange information so as to compute the pairwise average. Let $\mathbf{x}(t)$ denote the vector of values on network at the t-th time. Gossip algorithms can be denoted by the equation as follows.

$$\mathbf{x}(t) = \mathbf{W}(t)\mathbf{x}(t-1) \tag{2.21}$$

where

$$\mathbf{W}(t) = \mathbf{I} - \frac{(\mathbf{e}_i - \mathbf{e}_j)(\mathbf{e}_i - \mathbf{e}_j)^{\mathrm{T}}}{2} \tag{2.22}$$

is a randomly selected averaging matrix which is independently selected at t-th time; $\mathbf{I}$ is the identity matrix; $\mathbf{e}_i = [0, \ldots, 0, 1, 0, \ldots, 0]^{\mathrm{T}}$ is the one-hot vector where only the i-th component equal to one; and i as well as j are randomly selected.

It is easy to see that $\mathbf{W}(t)$ satisfies the conditions of the double stochastic matrix,

$$\mathbf{1}^{\mathrm{T}}\mathbf{W}(t) = \mathbf{1}^{\mathrm{T}}, \tag{2.23}$$

$$\mathbf{W}(t)\mathbf{1} = \mathbf{1}. \tag{2.24}$$

which ensure that the average is preserved at every iteration because

$$\mathbf{x}(t-1)^{\mathrm{T}}\mathbf{1} = \mathbf{x}(t-1)^{\mathrm{T}}\mathbf{W}(t)\mathbf{1} = \mathbf{x}(t)^{\mathrm{T}}\mathbf{1}.$$

It is noted that $\mathbf{W}(t)$ is a symmetric matrix according to its definition. Besides that, $\mathbf{W}(t)$ is an idempotent matrix, i.e., $\mathbf{W}^2(t) = \mathbf{W}(t)$, because the average of the pairwise average is the same as itself. Thus, $\mathbf{W}(t)$ is a projection matrix so as to be positive semi-definite.

The definition in Equation (2.21) can be expended to

$$\mathbf{x}(t) = \mathbf{W}(t)\mathbf{x}(t-1) = \prod_{s=0}^{t} \mathbf{W}(s)\mathbf{x}(0)$$

where $\mathbf{x}(0)$ is the initial value on the network. In this section, $\mathbf{x}(0)$ consists of either b or elements in $\mathbf{a}$ for all the sensors. The average value for the whole network is

$$\mathbf{x}_{\mathrm{ave}} = \mathbf{1}\mathbf{j}^{\mathrm{T}}\mathbf{x}(0)$$

where $\mathbf{j}$ is the vector in which all the entries are equal to $1/n$, $\mathbf{j} = [1/n, 1/n, \ldots, 1/n]^{\mathrm{T}}$. The desired result is that

$$\mathbf{x}(t) \to \mathbf{x}_{\mathrm{ave}}$$

when t is sufficient. Here, we define the error vector at t-th time as

$$\mathbf{e}(t) = \mathbf{x}(t) - \mathbf{x}_{\text{ave}} = \big(\prod_{s=0}^{t} \mathbf{W}(s) - \mathbf{1}\mathbf{j}^{\mathrm{T}}\big)\mathbf{x}(0). \tag{2.25}$$

If $\mathbf{x}(t)$ converges to $\mathbf{x}_{\text{ave}}$, $\mathbf{e}(t) \to \mathbf{0}$. Since $\mathbf{W}(t)$ are independently selected according to Equation (2.22), we investigate the expectation of $\mathbf{W}(t)$, $\mathbb{E}[\mathbf{W}(t)]$, which can be calculated by

$$\mathbb{E}[\mathbf{W}(t)] = \sum_{(i,j)\in\mathcal{E}} p_{(i,j)}\mathbf{W}(i,j) \tag{2.26}$$

where $\mathcal{E}$ denotes the edge set; $p_{(i,j)}$ is the probability that the edge between node i and node j is selected; and $\mathbf{W}(i,j)$ denotes $\mathbf{W}(t)$ when node i and node j are selected. The probability $p_{(i,j)}$ is calculated by

$$p_{(i,j)} = \pi(i)p(i,j) + \pi(j)p(j,i)$$

where $\pi(i)$, $i = 1, \ldots, n$, is the probability that node i is selected; $p(i,j)$ is the probability that the gossip communication from node i to node j is established. In this chapter, the acoustic sensor network is simply set as a complete graph. Each node has the same probability to be selected such that $\pi(i)$ follows a discrete uniform distribution. For each node, the probabilities that the gossip communication is established to others are the same. Thus, the probability mass functions, $\pi(i)$ and $p(i,j)$ can be denoted as

$$\begin{aligned} \pi(i) &= \frac{1}{n} \qquad i = 1, \ldots, n, \\ p(i,j) &= \frac{1}{n-1} \qquad (i,j) \in \mathcal{E}. \end{aligned}$$

Under these assumptions, $\pi(i)p(i,j) = \pi(j)p(j,i)$ and

$$
\begin{aligned}
\mathbb{E}[\mathbf{W}(t)] &= \sum_{(i,j)\in\mathcal{E}} p_{(i,j)}\left(\mathbf{I} - \frac{(\mathbf{e}_i - \mathbf{e}_j)(\mathbf{e}_i - \mathbf{e}_j)^{\mathrm{T}}}{2}\right) \\
&= \sum_{(i,j)\in\mathcal{E}} p_{(i,j)}\mathbf{I} - \sum_{(i,j)\in\mathcal{E}} p_{(i,j)}\frac{(\mathbf{e}_i - \mathbf{e}_j)(\mathbf{e}_i - \mathbf{e}_j)^{\mathrm{T}}}{2} \\
&= \mathbf{I} - \frac{2}{n(n-1)}\sum_{(i,j)\in\mathcal{E}} \frac{(\mathbf{e}_i - \mathbf{e}_j)(\mathbf{e}_i - \mathbf{e}_j)^{\mathrm{T}}}{2} \\
&= \mathbf{I} - \frac{1}{n(n-1)}\sum_{(i,j)\in\mathcal{E}} (\mathbf{e}_i - \mathbf{e}_j)(\mathbf{e}_i - \mathbf{e}_j)^{\mathrm{T}}
\end{aligned}
$$

Recall the incidence matrix $\mathbf{D}$ which is defined as

$$
\mathbf{D}_{il} = \begin{cases} 1, & \text{if node } i \text{ is tail on edge } l, \\ -1, & \text{if node } i \text{ is head on edge } l, \\ 0, & \text{otherwise.} \end{cases}
$$

The Laplacian matrix of the graph is defined as $\mathbf{L} = \mathbf{D}\mathbf{D}^{\mathrm{T}}$. It is noted that $(\mathbf{e}_i - \mathbf{e}_j)$ denote one edge from node i to node j which is one column of the incidence matrix $\mathbf{D}$. Thus, $(\mathbf{e}_i - \mathbf{e}_j)^{\mathrm{T}}$ is one row of the matrix $\mathbf{D}^{\mathrm{T}}$. The Laplacian matrix $\mathbf{L}$ can be computed by

$$
\mathbf{L} = \sum_{(i,j)\in\mathcal{E}} (\mathbf{e}_i - \mathbf{e}_j)(\mathbf{e}_i - \mathbf{e}_j)^{\mathrm{T}}.
$$

The expectation of $\mathbf{W}(t)$ can be rewritten as

$$
\mathbb{E}[\mathbf{W}(t)] = \mathbf{I} - \frac{1}{n(n-1)}\mathbf{L}. \tag{2.27}
$$

It is noted that the time index can be dropped in $\mathbb{E}[\mathbf{W}(t)]$. Let $\overline{\mathbf{W}}$ refer to expected averaging matrix $\mathbb{E}[\mathbf{W}]$. Let $\bar{\mathbf{e}}(t)$ denote the expected error vector. Since $\mathbb{E}[\mathbf{W}(t)]$ is a weighted sum of matrix $\mathbf{W}(i,j)$ in Equation (2.26), $\overline{\mathbf{W}}$ is a double stochastic,

projection and positive semi-definite matrix. The expectation of the error vector in Equation (2.25) can be denoted by

$$\bar{\mathbf{e}}(t) = \left(\overline{\mathbf{W}}^t - \mathbf{1}\mathbf{j}^{\mathrm{T}}\right)\mathbf{x}(0).$$

Since $\overline{\mathbf{W}}$ is a double stochastic matrix which satisfies the condition in Equation (2.23) and (2.24),

$$\bar{\mathbf{e}}(t) = \overline{\mathbf{W}}^t\left(\mathbf{I} - \mathbf{1}\mathbf{j}^{\mathrm{T}}\right)\mathbf{x}(0).$$

Since $\mathbf{I} - \mathbf{1}\mathbf{j}^{\mathrm{T}}$ is the linear combination of matrices which satisfies the Condition 2 in Theorem 2.1, $\mathbf{I} - \mathbf{1}\mathbf{j}^{\mathrm{T}}$ is an idempotent matrix.

$$\bar{\mathbf{e}}(t) = \overline{\mathbf{W}}^t\left(\mathbf{I} - \mathbf{1}\mathbf{j}^{\mathrm{T}}\right)^t\mathbf{x}(0)$$

Since both $\overline{\mathbf{W}}^t$ and $\mathbf{I} - \mathbf{1}\mathbf{j}^{\mathrm{T}}$ are symmetric,

$$\begin{aligned}
\bar{\mathbf{e}}(t) &= \left(\overline{\mathbf{W}}\left(\mathbf{I} - \mathbf{1}\mathbf{j}^{\mathrm{T}}\right)\right)^t\mathbf{x}(0) \\
&= \left(\overline{\mathbf{W}} - \mathbf{1}\mathbf{j}^{\mathrm{T}}\right)^t\mathbf{x}(0) \\
&= \left(\overline{\mathbf{W}} - \mathbf{1}\mathbf{j}^{\mathrm{T}}\right)^{t-1}\mathbf{e}(1) \\
&\vdots \\
&= \left(\overline{\mathbf{W}} - \mathbf{1}\mathbf{j}^{\mathrm{T}}\right)\mathbf{e}(t-1) \qquad (2.28)
\end{aligned}$$

The limitation $\lim_{t\to+\infty}\bar{\mathbf{e}}(t) = \mathbf{0}$ holds, if and only if $\rho\left(\overline{\mathbf{W}} - \mathbf{1}\mathbf{j}^{\mathrm{T}}\right) < 1$ [16]. The verification details are shown as follows.

Assume $\rho\left(\overline{\mathbf{W}} - \mathbf{1}\mathbf{j}^{\mathrm{T}}\right) < 1$.

$$\|\bar{\mathbf{e}}(t)\|_2 = \|\left(\overline{\mathbf{W}} - \mathbf{1}\mathbf{j}^{\mathrm{T}}\right)^t\mathbf{x}(0)\|_2 \leqslant \|\left(\overline{\mathbf{W}} - \mathbf{1}\mathbf{j}^{\mathrm{T}}\right)^t\|_2\|\mathbf{x}(0)\|_2 \leqslant \|\overline{\mathbf{W}} - \mathbf{1}\mathbf{j}^{\mathrm{T}}\|_2^t\|\mathbf{x}(0)\|_2$$

According to Theorem 2.2 on page 33, we have

$$\|\bar{\mathbf{e}}(t)\|_2 \leqslant \rho\left(\overline{\mathbf{W}} - \mathbf{1}\mathbf{j}^{\mathrm{T}}\right)^t \|\mathbf{x}(0)\|_2.$$

When $\rho\left(\overline{\mathbf{W}} - \mathbf{1}\mathbf{j}^{\mathrm{T}}\right) < 1$, $\lim_{t\to+\infty}\rho\left(\overline{\mathbf{W}} - \mathbf{1}\mathbf{j}^{\mathrm{T}}\right)^t = 0$. It is evident that $\|\bar{\mathbf{e}}(t)\|_2 \to 0$ when $t \to +\infty$ so that $\lim_{t\to+\infty}\bar{\mathbf{e}}(t) = \mathbf{0}$ holds. Since $\lim_{t\to+\infty}\bar{\mathbf{e}}(t) = \mathbf{0}$, it follows that $\lim_{t\to+\infty}\left(\overline{\mathbf{W}} - \mathbf{1}\mathbf{j}^{\mathrm{T}}\right)^t \mathbf{x}(0) = \mathbf{0}$ for any given vector $\mathbf{x}(0)$. If $\rho\left(\overline{\mathbf{W}} - \mathbf{1}\mathbf{j}^{\mathrm{T}}\right) > 1$, there must be an eigenvector $\mathbf{u}$ corresponding to eigenvalue λ with $|\lambda| > 1$. Since

$$\left(\overline{\mathbf{W}} - \mathbf{1}\mathbf{j}^{\mathrm{T}}\right)\mathbf{u} = \lambda\mathbf{u}, \left(\overline{\mathbf{W}} - \mathbf{1}\mathbf{j}^{\mathrm{T}}\right)^2\mathbf{u} = \lambda^2\mathbf{u}, \ \cdots \ , \left(\overline{\mathbf{W}} - \mathbf{1}\mathbf{j}^{\mathrm{T}}\right)^t\mathbf{u} = \lambda^t\mathbf{u},$$

it is evident that $\lim_{t\to+\infty}\left(\overline{\mathbf{W}} - \mathbf{1}\mathbf{j}^{\mathrm{T}}\right)^t\mathbf{u} = \mathbf{0}$ cannot hold if $|\lambda| > 1$. By contradiction, $\rho\left(\overline{\mathbf{W}} - \mathbf{1}\mathbf{j}^{\mathrm{T}}\right) < 1$.

The eigenvalue of matrix $\overline{\mathbf{W}}$ in terms of those of matrix $\mathbf{L}$ is [110]

$$\lambda_i(\overline{\mathbf{W}}) = 1 - \frac{1}{n(n-1)}\lambda_{n-i+1}(\mathbf{L})$$

where $\lambda_i(\cdot)$ denotes the i-th largest eigenvalue of a symmetric matrix. For a complete graph, the eigenvalue of the Laplacian matrix are 0, with multiplicity 1, and n, with multiplicity n-1 [4]. The eigenvalues of matrix $\mathbf{L}$ can be denoted as $[\underbrace{n, \ldots, n}_{n-1}, 0]$. Since the eigenvalue of $\mathbf{1}\mathbf{j}^{\mathrm{T}}$ is 1 with the eigenvector $\mathbf{1}$, the spectral radius of $\overline{\mathbf{W}} - \mathbf{1}\mathbf{j}^{\mathrm{T}}$ is [110]

$$\rho\left(\overline{\mathbf{W}} - \mathbf{1}\mathbf{j}^{\mathrm{T}}\right) = \max\left\{1 - \frac{1}{n(n-1)}\lambda_{n-1}(\mathbf{L}), \ \frac{1}{n(n-1)}\lambda_1(\mathbf{L}) - 1\right\}.$$

Therefore,

$$\rho\left(\overline{\mathbf{W}} - \mathbf{1}\mathbf{j}^{\mathrm{T}}\right) = \frac{n-2}{n-1} < 1.$$

The gossip algorithm is convergent in expectation in our wireless acoustic sensor network. We require the following theorems to analysis the error vector, which are given by [8] and [47] respectively.

Theorem 2.1. *Given two different non-zero idempotent matrices* $\mathbf{P}_1$ *and* $\mathbf{P}_2$*, let* $\mathbf{P}$ *denote the linear combination of the form*

$$\mathbf{P} = c_1\mathbf{P}_1 + c_2\mathbf{P}_2$$

with non-zero scalars c_1 *and* c_2*. In three conditions demonstrated as follows,* $\mathbf{P}$ *is an idempotent matrix:*

(1) $c_1 = 1$, $c_2 = 1$, $\mathbf{P}_1\mathbf{P}_2 = \mathbf{0}$*;*

(2) $c_1 = 1$, $c_2 = -1$, $\mathbf{P}_1\mathbf{P}_2 = \mathbf{P}_2$*;*

(3) $c_1 = -1$, $c_2 = 1$, $\mathbf{P}_1\mathbf{P}_2 = \mathbf{P}_1$.

Theorem 2.2. *If* $\mathbf{W}$ *is a symmetric matrix, the 2-norm of* $\mathbf{W}$*,* $\|\mathbf{W}\|_2$ *is equal to the spectral radius of* $\mathbf{W}$*,* $\rho(\mathbf{W})$.

2.4 Blockchain protection

Due to the complexity of computation and network involved in the normal blockchain version, we only use the basic blockchain functions including: adding block, hashing block, block validations and the longest chain rule. Each blockchain has one ledger system to store data. The ledger is an append-only block structure in which the block cannot be removed or modified once it has been added. In order to cooperate with gossip algorithms, there are two ledger systems to separately store the local estimates

of both $\mathbf{a}$ and b in one group. When the network begins to estimate $\check{\mathbf{R}}(f,k)$, the blockchains related to $\check{\mathbf{R}}(f,k-1)$ are removed and new blockchains are initialized. Therefore, the blockchains do not need much memory space.

Figure 2.4 demonstrates the ledger structure of the blockchain group in Figure 2.3(b) when $\hat{\mathbf{R}}(f,k)^{-1}$ is estimated. Each node in the group has a copy of both Ledger 1 and 2. One block in the blockchain contains index, timestamp, data and hash value. The block except for the genius block also contains a pervious hash value. The data stored in the genius block contains $\hat{\mathbf{R}}_v(f,k-1)^{-1}, \gamma_i(f)$ and $\mathbf{q}_i(f)$ which are the essential data to calculate the local estimates $\hat{\mathbf{R}}_v(f,k)^{-1}$ using Equation (2.13) and (2.10). It is noted that, for different groups, $\mathbf{a}^{(v,0)}$ and $b^{(v,0)}$ defined by Equation (2.15) and (2.18) should be different such that the hash value for each group cannot be the same. Moreover, the inter-group communication is dominated by gossip algorithms. The selected time for each group in the inter-group communication is at random. Thus, the lengths of Ledger 1 and 2 in each blockchain group could be different. In this chapter, we use the fully private blockchain which is open for other groups for reading but the permission to write it belongs to the nodes inside the group itself.

The hash value is generated based on the index, timestamp, data and the previous hash. It can be considered as a fingerprint of the block. When anything in the block is broken, the hash value will be changed. Since the hash value is linked to the next block, the hash values of all the blocks after this block should also be changed. Compared to other healthy chain, it is easy to find out the broken chain or broken block. In the inter-group communication, another groups can easily verify the correctness of the received data using the hash function.

The advantages of using blockchain in the WASN are the following:

- It perfectly resolves the data consensus problem inside the group since data is immutable and tamper-proof in blockchain.

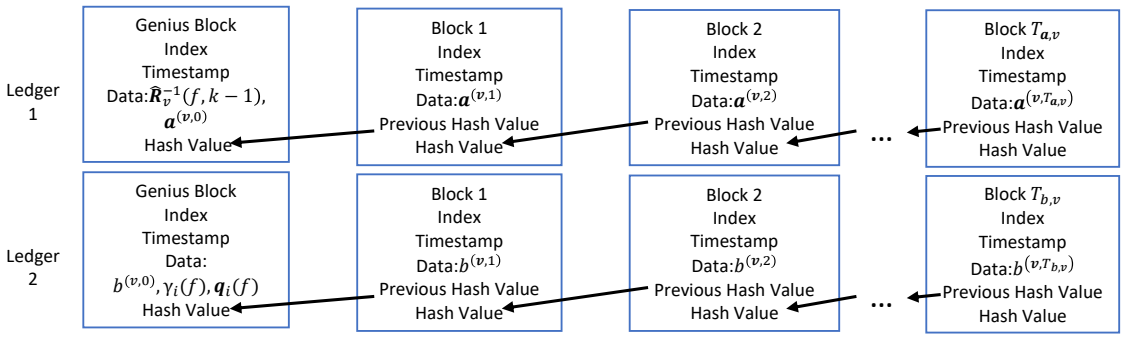


Figure 2.4: Designed Data Structure of one blockchain group in WASN

- It accelerates the gossip algorithm since the graph is compressed by the non-overlapping blockchain group.
- Once a communication link is broken, same data can be transmitted from others in the same blockchain group.
- Broken data in one node can be recovered by duplicated data from other nodes in the same blockchain group.

With the increment of number of nodes in the blockchain group, the number of duplication ledgers is increased. It leads to more bandwidth consumption because blockchain technique synchronizes the status of duplication ledgers. Therefore, we use a two-level communication scheme and employ the blockchain technique in small separate groups rather than in the whole WASN.

2.5 Simulation

In this section, we illustrate the performance of the designed beamformer in a simulated room. Firstly, we consider a $10m \times 10m \times 3m$ square office room with a reverberation time of $T_{60} = 0.2s$. The heights of microphones and sources are 1.5 meters. The horizontal positions of both sources and microphones are shown in

Figure 2.5. The size of our simulated room is a common specification for a large conference room or a small lecture theater. A similar size of our acoustic system is popular in the literature of studying the distributed beamforming [118]. We construct the blockchain groups according to the set $\{(\texttt{Node 1}, \texttt{Node 2}, \texttt{Node 3}), (\texttt{Node 4}, \texttt{Node 5}, \texttt{Node 6}), (\texttt{Node 7}, \texttt{Node 8}), (\texttt{Node 9}, \texttt{Node 10})\}$. Since signals are generated based on a signal propagation model with a known source location via Equation (3.1) as in [72], we can randomly select a node to be the reference signal in our model. One microphone of `Node` 8 is randomly selected to be the reference microphone. Figure 2.6 displays the RIR vector for the reference microphone. Furthermore, both source speech and interference speech contain $4s$ voice signals sampled at 16kHz. All signals are transformed in the frequency domain by a 256-tap FIR filter. The over-lapping rate is 50%. In this experiment, the signal-to-interference (SIR) ratio is fixed at -5dB.

Four performance measures are used to evaluate the performance in different cases. Define $\hat{P}_Y(\omega)$ as the spectral power estimate of the source signal; $\hat{P}_{\tilde{Y}}(\omega)$ as the spectral power estimate of the output of beamformer; $\hat{P}_{Y_I}(\omega)$ as the spectral power estimate of the interference speech; $\hat{P}_{\tilde{Y}_I}(\omega)$ as the spectral power estimate of the output of beamformer when the interference speech is active alone; $\hat{P}_{Y_{\texttt{WN}}}(\omega)$ as the spectral power estimate of the white noise; $\hat{P}_{\tilde{Y}_{\texttt{WN}}}(\omega)$ as the spectral power estimate of the output of beamformer when the white noise is active alone. Then, the first performance measure is the normalized distortion which is formulated as

$$\texttt{Distortion} = \frac{1}{\pi}\int_{-\pi}^{\pi} |C_d \hat{P}_{\tilde{Y}}(\omega) - \hat{P}_Y(\omega)| d\omega$$

where C_d is defined as

$$C_d = \frac{\int_{-\pi}^{\pi} \hat{P}_Y(\omega) d\omega}{\int_{-\pi}^{\pi} \hat{P}_{\tilde{Y}}(\omega) d\omega}.$$

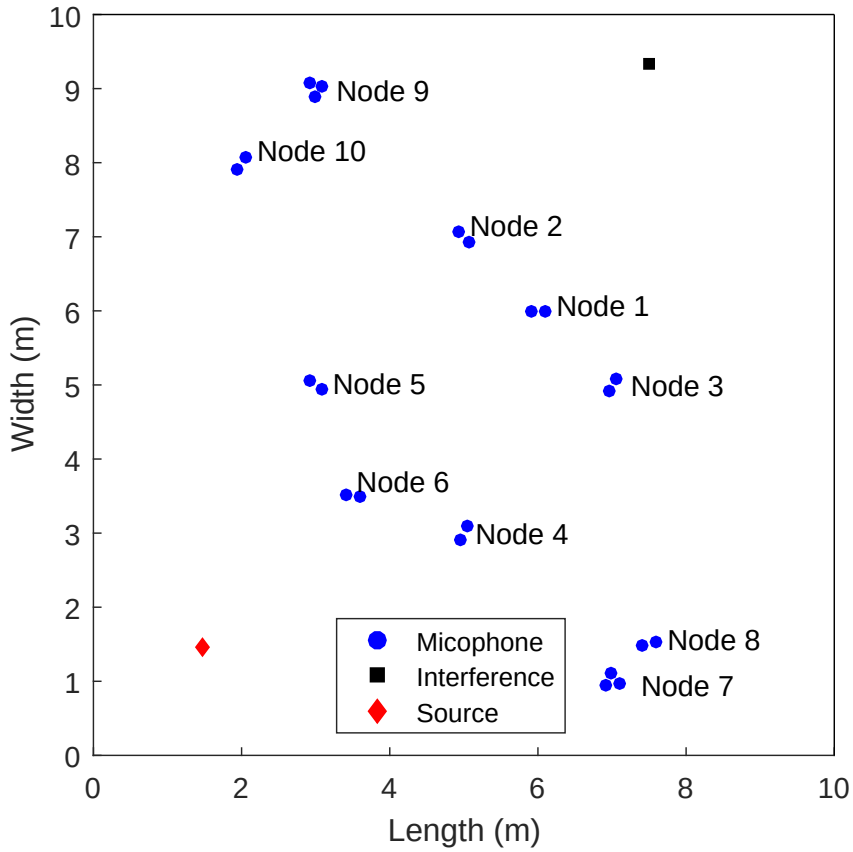


Figure 2.5: Locations of 10 nodes (22 microphones), 1 source speech and 1 interference speech in the simulated room

The second and third performance measures are the normalized white noise suppression and normalized interference suppression which are respectively defined by

$$\mathtt{SUPP}_{\mathtt{WN}} = \frac{\int_{-\pi}^{\pi} \hat{P}_{\tilde{Y}_{\mathtt{WN}}}(\omega) d\omega}{C_d \int_{-\pi}^{\pi} \hat{P}_{Y_{\mathtt{WN}}}(\omega) d\omega},$$

$$\mathtt{SUPP}_{\mathtt{I}} = \frac{\int_{-\pi}^{\pi} \hat{P}_{\tilde{Y}_{\mathtt{I}}}(\omega) d\omega}{C_d \int_{-\pi}^{\pi} \hat{P}_{Y_{\mathtt{I}}}(\omega) d\omega}.$$

The fourth measure is the segmental SNR ratio computed by

$$\mathtt{SNR}_{\mathtt{seg}} = \frac{1}{K_2} \sum_{k=1}^{K_2} 10 \log_{10} \frac{\sum_{f=1}^{F} |Y_i(f,k)|^2}{\sum_{f=1}^{F} |\tilde{Y}(f,k) - Y_i(f,k)|^2}$$

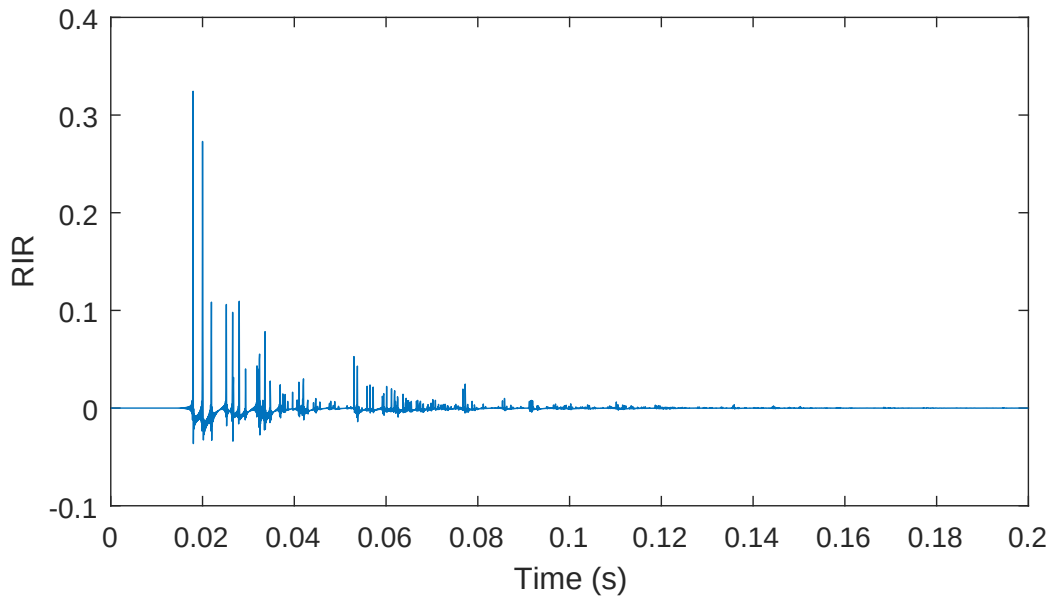


Figure 2.6: The room impulse response vector for the reference microphone

where $Y_i(f,k)$ is the STFT coefficient of the clean speech received by the i-th microphone and F is the total number of frequency bins. The first three methods have been used to measure the performance of centralized MMSE beamformer in [112].

In order to show the importance of blockchain protection, we simulate a poor network environment with heavy packet loss. The rate of transmission failure in the WASN is $1e-4$. If the transmission fails, the receiver will get the null data from the network. We proposed three distributed computational methods below.

- Method 1 (M1): The MMSE beamformer is distributively computed by the randomized gossip algorithm.
- Method 2 (M2): The MMSE beamformer is distributively computed by the randomized gossip algorithm. Blockchain protection is active.
- Method 3 (M3): The MMSE beamformer is distributively computed by the greedy gossip algorithm. Blockchain protection is active.

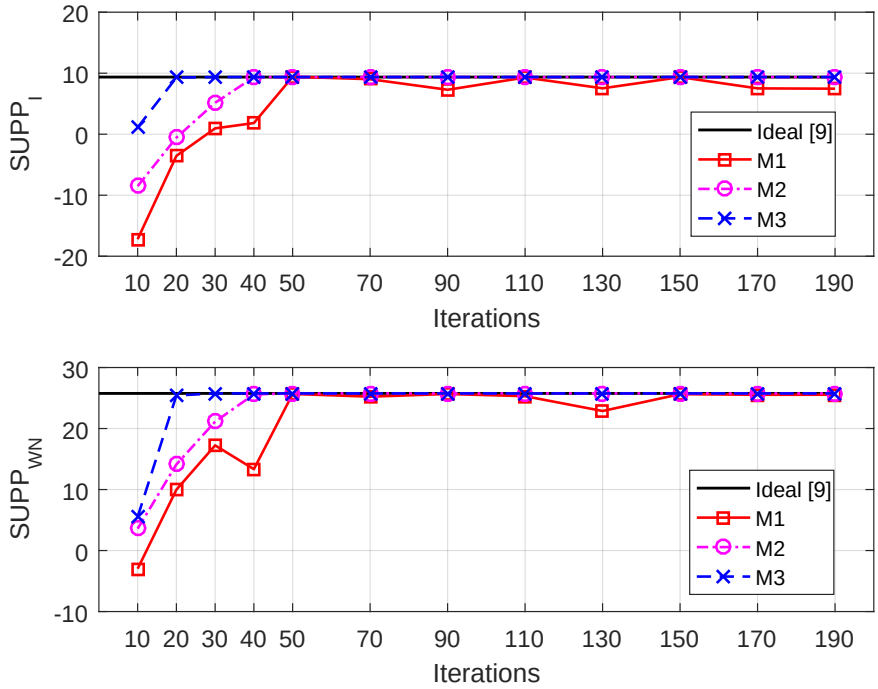
For comparison, we calculate the optimal beamformer with all the data readily

available in one node, that is, like the centralized counterpart beamformer in which each node would have access to the full set of microphone signals. This is the ideal situation in which all data can be accessed and without any lost of precision due to communication failure. This ideal beamformer has been investigated in [84] and will be used as a benchmark in the following comparison. Moreover, we make a comparison with the distributed delay-and-sum beamformer (DDSB) in [119] as well as the distributed minimum variance distortionless response beamformer (DMVDRB) in [118]. Both DDSB and DMVDRB are calculated by the randomized gossip algorithm.

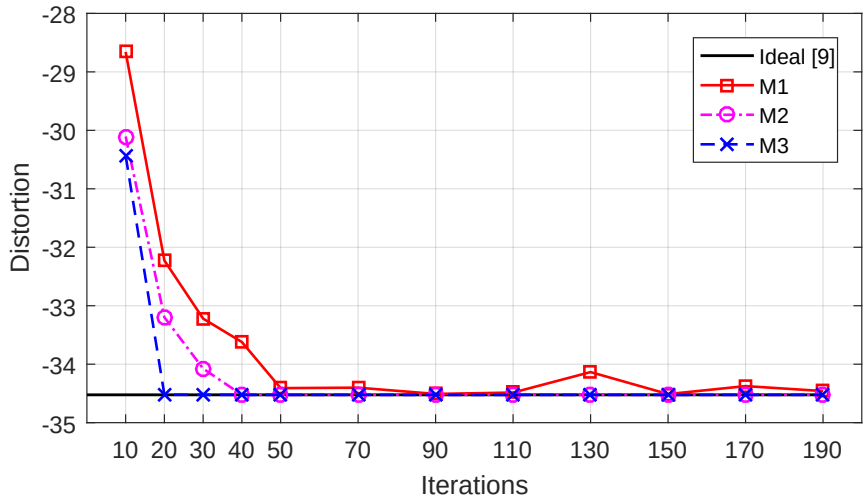
2.5.1 Results with a fixed SNR

The input SNR is fixed as -5dB. We investigate the performance of the distributed MMSE beamformer with and without the blockchain protection. Besides that, we compare difference between the randomized gossip algorithm and greedy gossip algorithm when the blockchain protection is active.

Figure 2.7 shows the values of interference suppression, white noise suppression and the distortion with the change of iteration numbers in gossip algorithms. As we can see from Figure 2.7(a), the beamformer reduces both the white noise and interference speech when the iteration number in gossip algorithms is greater than 30. It is clear that the performance of beamformer is improved over iterations. As we mentioned in Section III, the greedy gossip algorithm used in M3 converges faster because optimal choices among neighbors have been selected. However, the greedy gossip algorithm needs additional resources to eavesdrop the information of neighbors in real time. The red line (M1) denotes the performance of the beamformer which is distributively calculated without blockchain data protection. From Figure 2.7(a) and 2.7(b), we can observe that distributed beamformers with blockchain protection outperform the one without protection. Moreover, the performance of the



(a) Suppression measures of the interference speech (upper) and white noise (lower) with the change of the iteration numbers in gossip algorithms



(b) Speech distortion measure with the change of the iteration numbers in gossip algorithms

Figure 2.7: The suppression measures and distortion measure in three methods when the SIR is -5dB and the SNR is -5dB

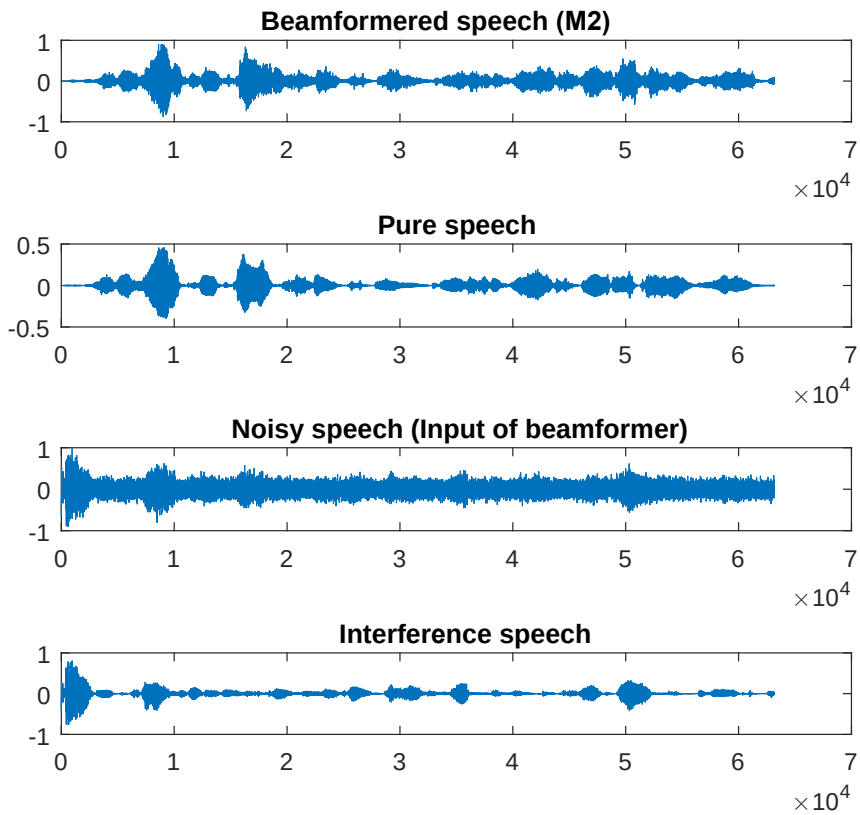


Figure 2.8: Beamforming performance of the distributed MMSE beamformer with the blockchain protection (M2)

distributed beamformer is unstable without the blockchain protection. Significant departure on all three performance measures can be observed in M1 even when the iteration number is large.

Figure 2.8 depicts the result using M2, namely the beamformered speech (the output of the beamformer), pure speech, noisy speech and the interference speech. It is observed that the distributed beamformer with blockchain protection is able to reduce the noise significantly.

2.5.2 Results with different SNRs

In this subsection, the SNR are chosen as -10dB, -5dB, 0dB and 5dB. The iteration number in the randomized gossip algorithm is 110. Figure 2.9 shows the segmental

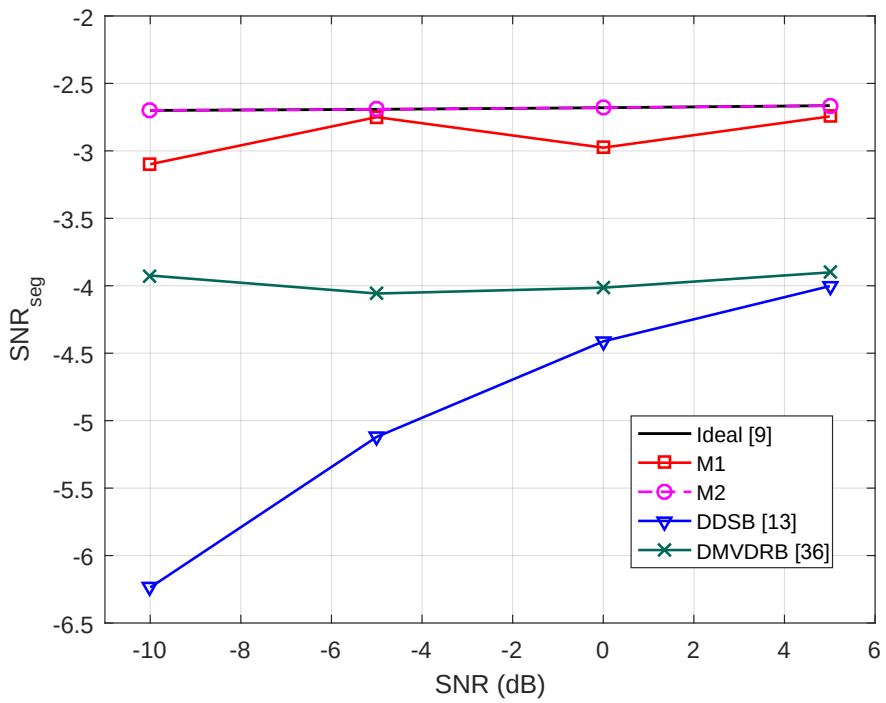


Figure 2.9: Segmental SNR with the input SNR chosen as -10dB, -5dB, 0dB and 5dB

SNR versus the input SNR. DMVDRB is calculated by the CbDECM$_1$ algorithm in [118]. DDSB is calculated by the randomized gossip algorithm with clique in [119]. It is observed that the distributed MMSE beamformers outperform both DDSB and DMVDRB in the reverberation environment. Furthermore, the distributed MMSE beamformer using M2 can achieve the performance of the ideal MMSE beamformer. Note that the performances of both the distributed beamformer using M2 and ideal beamformer are the same and therefore these two plots are overlapped in the Figure. Comparing with M2, the performance of M1 is obviously poorer due to the lack of blockchain protection when transmission errors exist.

2.5.3 Error analysis

Since $\mathbf{a}$ and b are transmitted to calculate $\check{\mathbf{R}}(f,k)^{-1}$, we define the estimated error rate of $\check{\mathbf{R}}(f,k)^{-1}$, in the v-th blockchain group, as

$$\mathtt{Err}_v(k) = \frac{1}{F}\sum_{f=1}^{F}\frac{\|\check{\mathbf{R}}_v(f,k)^{-1} - \check{\mathbf{R}}_i(f,k)^{-1}\|_{\mathtt{F}}}{\|\check{\mathbf{R}}_i(f,k)^{-1}\|_{\mathtt{F}}} \tag{2.29}$$

where $\|\cdot\|_{\mathtt{F}}$ denotes the Frobenius Norm of the matrix. $\check{\mathbf{R}}_v(f,k)^{-1}$ and $\check{\mathbf{R}}_i(f,k)^{-1}$ are the estimates of $\check{\mathbf{R}}(f,k)^{-1}$ from the distributed beamformer and ideal beamformer. In this case, we focus on the error rate of the third blockchain group when SNR is equal to -5dB. Figure 2.10 displays the comparison of the cumulative mean of the estimated error rate of $\check{\mathbf{R}}(f,k)^{-1}$ in both M1 and M2. It is observed that the cumulative mean of $\mathtt{Err}_3$ in M2 roughly behaves like a horizontal line. It means, in M2, that the randomized gossip algorithm attains a stable state and the estimate of $\check{\mathbf{R}}(f,k)^{-1}$ converges to the estimate of $\check{\mathbf{R}}(f,k)^{-1}$ from the ideal beamformer. Without the blockchain protection, the cumulative means of $\mathtt{Err}_v$ in M1 are larger than the counterpart in M2. Apart from that, in the result for M1, there are several jumps which are caused by the transmission failure in the WASN.

2.6 Discussion

From the comparison results depicted in Figure 2.7, we observe that the performance of the distributed MMSE beamformer without blockchain protection (M1) is worst and unstable when it is put under a poor network environment with heavy packet loss. The designed beamformer drifts away from the optimal performance from time to time, and it takes a while for it to converge again. On the other hand, the proposed distributed MMSE beamformer with blockchain protection (M2) can approach the performance of the ideal beamformer and stay at the optimal level.

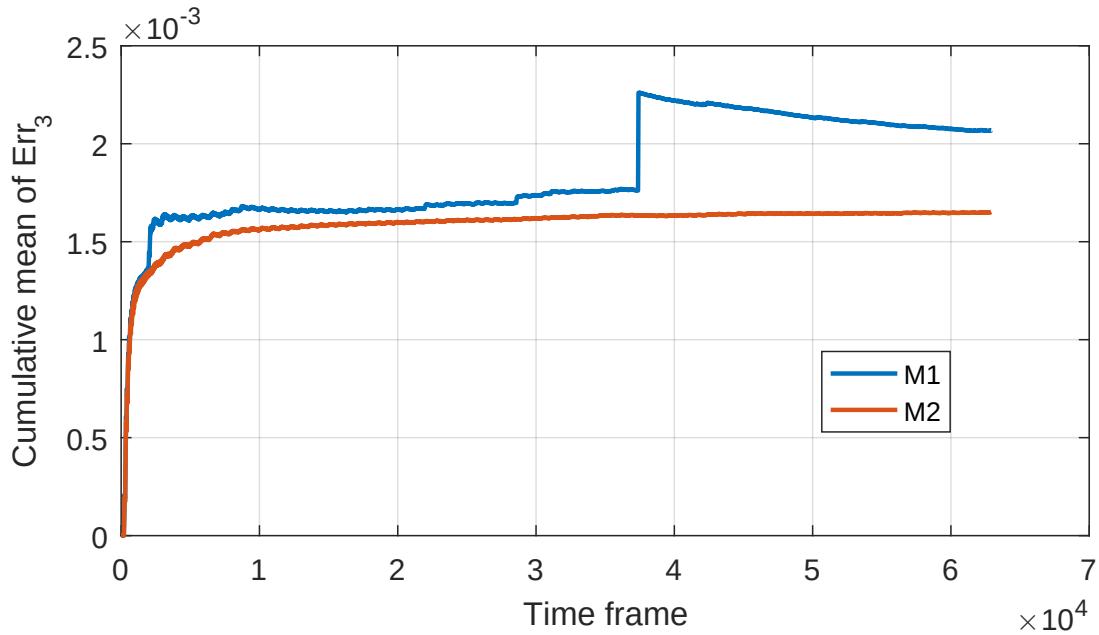


Figure 2.10: Estimation of error rate of $\check{\mathbf{R}}(f,k)^{-1}$ on the third blockchain group for M1 and M2 at SNR=-5dB.

Moreover, it we elaborate on the iterative technique further by employing the greedy gossip algorithm instead of the randomized gossip algorithm, the distributed MMSE beamformer with blockchain protection (M3) can converge even faster, with the trade-off for using additional resources to make the optimal choice among neighbors in each iteration. The signal outputs from the distributed MMSE beamformer with blockchain protection (M2) is shown in Figure 2.9. From the beamformed speech output, it can be seen that the white noise and interference speech have been by and large filtered out, leaving a good estimate of the required speech signal.

Figure 2.8 displays the segmental SNR performance of output signals for a range of input SNRs. Compared with two existing implementations of distributed beamforming in the literature, namely DDSB [119] and DMVDRB [118], the distributed MMSE beamformers (M1 and M2) outperforms them in the indoor environment with reverberation. Furthermore, the proposed distributed MMSE beamformer with blockchain protection (M2) is the only one that can achieve the performance of the

ideal beamformer for the whole range of SNRs in the study. Without blockchain protection, the performance of M1 often deviates from the optimal performance.

To further understand the cause of the deviation, we conduct an error analysis and find that it is mainly due to errors in the computation of $\check{\mathbf{R}}(f,k)^{-1}$. In Figure 2.10. An error measure is introduced in Equation (2.29) to quantify the effect. It can be seen that faulty data causes the matrix to drift away from the correct value and hence induce performance degradation in the designed beamformer. It is also evident that this problem can be avoided with blockchain protection, which reduces the computational error caused by poor transmission.

2.7 Final remark

In this chapter, we propose three distributed MMSE beamformers using gossip algorithms. Our proposed distributed MMSE beamformers outperform the distributed delay-and-sum beamformer and MVDR beamformer in a simulated reverberation environment. Based on the blockchain technique, a data protection scheme is also proposed to avoid faulty data transmissions. To illustrate the effectiveness of the proposed beamformer with the blockchain protection, we simulated a typical scenario in a square office room with reverberation. The experimental results show that blockchain data protection is able to secure the quality of the output signals from a distributed beamformer in a relatively poor network environment. In addition, we showed that greedy algorithm can perform better than the randomized gossip algorithm if additional sources are used to receive information from all neighbors.

2.8 Appendix: MVDR beamformer

The problem of minimum variance distortionless (MVDR) beamformer can be treated as a linear constrained problem which is defined as

$$\mathbf{w}_{\mathtt{opt}}(f) = \underset{\mathbf{w}(f)}{\arg\min}\ \mathbf{w}(f)^{\mathrm{H}}\hat{\mathbf{R}}_{YY}(f,k)\mathbf{w}(f); \tag{2.30}$$

$$s.t. \quad \mathbf{w}(f)^{\mathrm{H}}\mathbf{d} = 1 \tag{2.31}$$

where $\hat{\mathbf{R}}_{YY}(f,k)$ is the estimate of the correlation matrix of input signals, and $\mathbf{d} = [\mathbf{d}_1, \ldots, \mathbf{d}_M]^{\mathrm{T}}$ is the steering vector representing the acoustic transfer function from the desired speech source to all microphones.

The recursive exponential smoothing method is applied to calculate the correlation matrix $\hat{\mathbf{R}}_{YY}(f,k)$, that is,

$$\hat{\mathbf{R}}_{YY}(f,k) = \lambda\hat{\mathbf{R}}_{YY}(f,k-1) + (1-\lambda) + \mathbf{Y}(f,k)\mathbf{Y}(f,k)^{\mathrm{H}}$$

where λ is the exponential weighting factor. The steering vector $\mathbf{d}$ can be derived from the acoustic transfer function in [114].

Using the method of Lagrange multipliers, The problem of MVDR beamformer can be transformed into an unconstrained optimization problem. The Lagrangian function is given by

$$\mathcal{L}(\mathbf{w}(f), \lambda_L) = \mathbf{w}(f)^{\mathrm{H}}\hat{\mathbf{R}}_{YY}(f,k)\mathbf{w}(f) + \lambda_L(\mathbf{w}(f)^{\mathrm{H}}\mathbf{d} - 1) + (\mathbf{d}^{\mathrm{H}}\mathbf{w}(f) - 1)\lambda_L^{\star}$$

The gradients of the Lagrangian function are

$$\bigtriangledown\mathcal{L}(\mathbf{w}(f)^{\mathrm{H}}, \lambda_L)_{\mathbf{w}(f)^{\mathrm{H}}} = \hat{\mathbf{R}}_{YY}(f,k)\mathbf{w}(f) + \lambda_L\mathbf{d}; \tag{2.32}$$

$$\bigtriangledown\mathcal{L}(\mathbf{w}(f), \lambda_L)_{\lambda_L} = \mathbf{w}(f)^{\mathrm{H}}\mathbf{d} - 1. \tag{2.33}$$

Let both $\bigtriangledown\mathcal{L}(\mathbf{w}(f)^{\mathrm{H}}, \lambda_L)_{\mathbf{w}(f)^{\mathrm{H}}}$ and $\bigtriangledown\mathcal{L}(\mathbf{w}(f), \lambda_L)_{\lambda_L}$ equal zero. we can obtain

$$\hat{\mathbf{R}}_{YY}(f,k)\mathbf{w}(f) + \lambda_L\mathbf{d} = 0 \tag{2.34}$$

$$\mathbf{w}(f)^{\mathrm{H}}\mathbf{d} - 1 = 0 \tag{2.35}$$

According to Equation (2.34), we obtain

$$\mathbf{w}(f) = -\lambda_L \hat{\mathbf{R}}_{YY}^{-1}(f,k)\mathbf{d} \tag{2.36}$$

It remains the only problem of evaluating the Lagrange multiplier λ_L. To solve λ_L, we substitute the result of Equation (2.36) into Equation (2.35) such that

$$\lambda_L = -\frac{1}{\mathbf{d}^{\mathrm{H}}\hat{\mathbf{R}}_{YY}^{-1}(f,k)\mathbf{d}} \tag{2.37}$$

Substituting this value of λ_L into Equation (2.34), the optimal MVDR weight $\mathbf{w}(f)$ is given by

$$\mathbf{w}_{\mathrm{opt}}(f) = \frac{\hat{\mathbf{R}}_{YY}^{-1}(f,k)\mathbf{d}}{\mathbf{d}^{\mathrm{H}}\hat{\mathbf{R}}_{YY}^{-1}(f,k)\mathbf{d}} \tag{2.38}$$

Using the Matrix Inversion Lemma, the inverse correlation matrix $\hat{\mathbf{R}}_{YY}^{-1}$ can be computed iteratively

$$\hat{\mathbf{R}}_{YY}^{-1}(f,k) = \lambda^{-1}\hat{\mathbf{R}}_{YY}^{-1}(f,k-1) - \frac{\lambda^{-2}(1-\lambda)\hat{\mathbf{R}}_{YY}^{-1}(f,k-1)\mathbf{Y}(f,k)\mathbf{Y}(f,k)^{\mathrm{H}}\hat{\mathbf{R}}_{YY}^{-1}(f,k-1)}{1+\lambda^{-1}(1-\lambda)\mathbf{Y}(f,k)^{\mathrm{H}}\hat{\mathbf{R}}_{YY}^{-1}(f,k-1)\mathbf{Y}(f,k)} \tag{2.39}$$

Both $\hat{\mathbf{R}}_{YY}^{-1}(f,k-1)\mathbf{Y}(f,k)$ and $\mathbf{Y}(f,k)^{\mathrm{H}}\hat{\mathbf{R}}_{YY}^{-1}(f,k-1)\mathbf{Y}(f,k)$ can be computed in a distributed way. Let

$$\mathbf{a} = \hat{\mathbf{R}}_{YY}^{-1}(f,k-1)\mathbf{Y}(f,k) = \sum_{m=1}^{M} \mathbf{r}_m Y_m \tag{2.40}$$

$$b = \mathbf{Y}(f,k)^{\mathrm{H}}\hat{\mathbf{R}}_{YY}^{-1}(f,k-1)\mathbf{Y}(f,k) = \sum_{m=1}^{M} Y_m^{\star} a_m \tag{2.41}$$

where $\mathbf{r}_m$ is the m-th column of $\hat{\mathbf{R}}_{YY}^{-1}(f,k-1)$; $Y_m^{\star}$ is the complex conjugate of the m-th element of $\mathbf{Y}(f,k)$; and a_m is the m-th element of $\mathbf{a}$. It is noted that $\mathbf{a}$ and b are in the form of weighted summation so as to be computed by the gossip algorithm.

Chapter 3

Optimization of sensor placement of blind source separation in a wireless acoustic sensor network

Blind source separation (BSS) method extracts the desired signals from a mixing observed signals. The nonlinear mixing problem in the reverberant environment degrades the performance of BSS model. With the development of the indoor GPS technique, it enables us to enhance the performance of BSS model via the optimization of microphone locations when the nonlinear mixing problem exists. By doing so, spatial information can be fully exploited via an optimized array geometry. In this chapter, we optimize the location of each microphone in the wireless acoustic sensor network so as to obtain better separated signals. A novel hybrid descent method is proposed to the optimization work. Results show that the optimized array consistently yield a greater suppression ($>$ 10dB) across the different reverberation time compared to an unoptimized linear configuration.

3.1 Research background

The cocktail party effect which was coined by Cherry in the early fifties illustrates humans' ability to focus on a specific talker in a multi-talker situations [31]. Part of

the explanation for this focussing capability lies in the spatial sampling performed by the two human ears. This spatial diversity makes use of the fact that the origins of the desired and interfering signals originate from different locations in space. With the development of communication equipments, speech separation or extraction may be realized electronically by sampling in space. A classical and typical application scenario is when speech signals are recorded by pre-mounted microphones in the wireless acoustic sensor network (WASN). Two common popular methods are blind source separation (BSS) and beamforming but the latter is very sensitive to array model mismatch [12, 75, 28]. With less assumptions of the acoustic scene or system, BSS has an upper hand against more conventional beamforming approaches in real world [6, 5].

BSS involves extracting and recovering the underlying source signals from multivariate statistical data [115]. It is widely used in the wireless digital communication, image processing and recognition as well as geological spatial information processing. Blind implies that the source localization information is not known. Generally, BSS manifests in two general cases. The first case is the underdetermined situation where the number of sensors is less than that of sources. Another is the determined or overdetermined situation where the number of sensors is greater than or equal to the number of sources. In the first case, nonnegative matrix factorization has received much attention [92]. For the second case, independent component analysis (ICA) is the method most commonly applied but requires use of higher-order statistics [57]. As opposed to ICA, the second-order statistics (SOS) method is generally preferred to separate nonstationary signals due to its computational simplicity [73, 86, 87]. For WASN settings, sensors can be placed in any suitable position without being restricted by the wired connections and large number of sensor nodes.

With the assumption that source signals are statistically independent, ICA methods separate source signals by expressing a set of random variables as a linear com-

bination of statistically independent variables [34]. ICA methods have been widely applied in many field and a variety of the ICA methods have been proposed. A nerual network based ICA algorithms was presented in [11]. Hyvarinen and Oja proposed an ICA algorithm based on the negentropy maximization of random variables method [56]. Masnadi-Shirazi and Rao proposed a state-based ICA method aimed to the non-stationarity of the signal [79]. With the contribution of these pioneering authors, the theoretical framework and algorithms for ICA methods have matured. However, there are still some unsolved problems of ICA method due to its restricted assumptions or lack of information. For instance, ICA algorithms perform poorly when the high-order statistics for original signals are dependent or the stationarity condition is violated.

A SOS method has been proposed to separate nonstationary signals by joint decorrelation [99, 86]. Compared to the higher-order method (ICA), this method makes no assumption about the cumulative densities of signals and puts itself to more robust second-order statistics. Owing to second-order processing only, this method is computationally efficient. A fast convergent BSS algorithm based on the SOS method was presented in [37]. Beamspace SOS method was introduced by [77] where a priori spatial information was embedded in the formulation as a preprocessor to help improve the separation performance. The performance of SOS method has been investigated in the case of reverberant acoustic environment [76].

In Araki et al.'s landmark contribution, they established the equivalence between BSS and a set of adaptive beamformers [5]. This means that BSS is no other than a set of spatial filters, albeit different in its formulation compared to a spatial filter. As such, the performance of BSS is highly dependent on the configuration of the sensor arrays similar to a beamformer. If so, the research question here is to ascertain if there is an optimum sensor configuration for BSS. This is akin to viewing sensor placement as a preprocessor for BSS to improve its separation capability. However,

the literature thus far have been limiting as the focus on BSS has been for a fixed set of linear and non-linear array configurations [77, 98]. The sensor placement investigation is particular relevant with the rise of wireless acoustic sensor networks (WASN). In such a WASN setting, the multiple cooperative devices may be optimized to provide a better set of sensors for the task at hand.

This chapter aims to fill the research gap by studying the effect of sensors placement on the performance of BSS, i.e., how to best place the sensors to fully exploit the spatial information for BSS. As opposed to focusing on a specific configuration, an optimization strategy is proposed to optimize the performance of BSS for the best sensor placement on the allowable placement space as the number of sensors and reverberation time change. Interestingly, the results show that there is largely a demarcation point where the sensors placement will have a certain configuration before and after a certain reverberation time (RT). The results also reveal that the sensors tend to cluster on the desired source when the RT is less than 200ms and spread out otherwise. Importantly, there results consistently show a marked increase in the separation performance of an optimized placement compared to a fixed placement.

The rest of this chapter is organized as follows. The problem formulation is given in Section 3.2. The ICA-based approach is introduced in Section 3.3. The hybrid descent method is introduced in Section 3.4. The simulation study is demonstrated in Section 3.5. Conclusion and potential future extensions are discussed in Section 3.6.

3.2 Notation and Problem formulation

In this work we consider an enclosed room with acoustic reverberation. In this room, N speech sources are settled at $\gamma_n, n = 0, \ldots, N-1$, and an M-element microphone

array is settled at positions δ_m, where $m = 1, \ldots, M$. Given the room dimension, sound speed, locations of sources and microphones, the time domain room impulse responses (RIR) $\mathbf{h}(\delta_m, \gamma_n)$ from n-th source to m-th microphone can be generated by the image method [72]. Let $\mathbf{s}_n$ denote the signal at the source γ_n. At time t, the received signal from n-th source to m-th microphone is given as

$$s_{m,n}(t) = \sum_{\tau=0}^{L-1} h_{m,n}(\tau) s_n(t-\tau) = \mathbf{h}(\delta_m, \gamma_n) * \mathbf{s}_n \tag{3.1}$$

where $h_{m,n}(\tau)$ is the $(\tau+1)$-th element in vector $\mathbf{h}(\delta_m, \gamma_n)$; L is the length of the RIR vector; $\mathbf{s}_n$ denotes the vector $[s_n(t), s_n(t-1), \ldots, s_n(t-L+1)]$; and $*$ is the convolution operator. The observed signal at the m-th microphone is mixed by

$$x_m(t) = \sum_{n=0}^{N-1} \mathbf{s}_{m,n} = \sum_{n=0}^{N-1} \mathbf{h}(\delta_m, \gamma_n) * \mathbf{s}_n. \tag{3.2}$$

The observed mixtures from the WASN is denoted by

$$\mathbf{x}(t) = \mathbf{H} \odot \mathbf{s} \tag{3.3}$$

where $\odot$ is the element-wise convolution operator; $\mathbf{x}(t) = [x_1(t), x_2(t), \ldots, x_M(t)]^{\mathrm{T}}$; $\mathbf{s}$ denotes the vector $[\mathbf{s}_0, \mathbf{s}_1 \ldots, \mathbf{s}_{N-1}]^T$ in Equation (3.3); and $\mathbf{H}$ is the mixing matrix consisting of RIR vectors,

$$\mathbf{H} = \begin{bmatrix} \mathbf{h}(\delta_1, \gamma_0)^{\mathrm{T}} & \cdots & \mathbf{h}(\delta_M, \gamma_{N-1})^{\mathrm{T}} \\ \vdots & \ddots & \vdots \\ \mathbf{h}(\delta_M, \gamma_0)^{\mathrm{T}} & \cdots & \mathbf{h}(\delta_M, \gamma_{N-1})^{\mathrm{T}} \end{bmatrix}.$$

Since the convolution operation in the time domain becomes multiplication in the frequency domain [96], the problem can be elegantly transformed to the simple instantaneous case if the Fourier transform (STFT) is applied on the observed signals. Rewriting Equation (3.3) in the frequency domain gives

$$\mathbf{X}(f, k) = \mathbf{H}(f)\mathbf{S}(f, k) \tag{3.4}$$

where $\mathbf{X}(f,k)$ and $\mathbf{S}(f,k)$ are the f-th subband transformations of $\mathbf{x}(t)$ and $\mathbf{s}$ respectively. $\mathbf{H}(f)$ is a matrix containing the elements of the mixing matrix $\mathbf{H}$ at the f-th subband and k is the frequency sampling index.

The objective here is to unmix the mixtures so as to recover the original signal $\mathbf{S}(f,k)$ from $\mathbf{X}(f,k)$. Assuming that the f-th subband of the mixing matrix is invertible, then the unmixing process is

$$\mathbf{Y}(f,k) = \mathbf{W}(f)\mathbf{X}(f,k) \tag{3.5}$$

where $\mathbf{W}(f)$ is the unmixing matrix in the f-th subband and $\mathbf{Y}(f,k)$ is the estimated separated signals in the frequency domain. Substituting Equation (3.4) into Equation (3.5), we derive

$$\mathbf{Y}(f,k) = \mathbf{W}(f)\mathbf{H}(f)\mathbf{S}(f,k). \tag{3.6}$$

As we can see, the separated signals $\mathbf{Y}(f,k)$ not only depends on the unmixing matrix $\mathbf{W}(f)$ in the separation system but also relates to the mixing matrix $\mathbf{H}(f)$ in the mixing matrix. Figure 3.1 displays the block diagram of our convolutive BSS model. The separation system is denoted by the dotted box whilst the acoustic mixing process is denoted by the solid box.

3.3 SOS method

In order to derive the optimal unmixing matrix $\mathbf{W}(f)$ in the separation system denoted by the dotted box in Figure 3.1, the SOS method in [77, 86] is adopted. This method exploits non-stationarity of the source signals to provide more information to separate the sources. To be specific, the covariance matrix of the non-stationary source signals, at different time intervals, are linearly independent such that additional information can be used to perform the separation process. In this case, the covariance matrix $\mathbf{R}_X(f,k)$ of the observed mixtures can be estimated on N

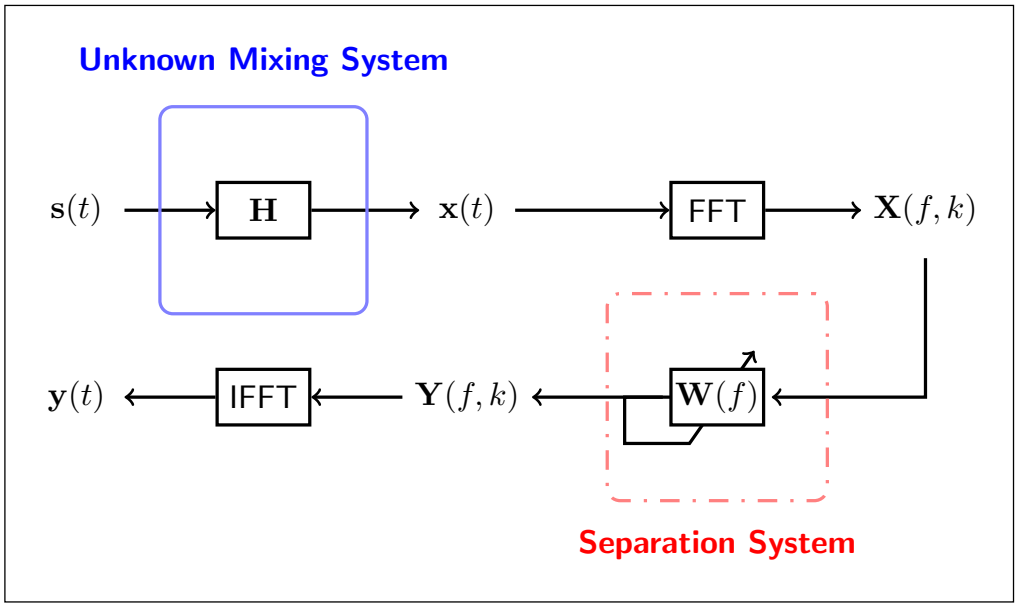


Figure 3.1: The block diagram of convolutive BSS model

successive intervals as

$$\mathbf{R}_X(f,n) = \frac{1}{I}\sum_{k=0}^{I-1}\mathbf{X}(f,nI+i)\mathbf{X}(f,nI+i)^H \tag{3.7}$$

where $n = 0,\ldots,N-1$; I is the length of the interval for estimating the covariance matrix; and $(\cdot)^H$ is the Hermitian transpose operator. By the projection of the unmixing matrix $\mathbf{W}(f)$, the covariance matrix $\mathbf{R}_Y(f,k)$ of the separated output is calculated by

$$\begin{aligned}\mathbf{R}_Y(f,n) &= \frac{1}{I}\sum_{k=0}^{I-1}\mathbf{Y}(f,nI+i)\mathbf{Y}(f,nI+i)^H \\ &= \frac{1}{I}\sum_{k=0}^{I-1}\mathbf{W}(f)\mathbf{X}(f,nI+i)\mathbf{X}(f,nI+i)^{\mathrm{H}}\mathbf{W}(f)^H \\ &= \mathbf{W}(f)\mathbf{R}_X(f,n)\mathbf{W}(f)^H \end{aligned} \tag{3.8}$$

If the source signals are successfully separated, the estimated source signal should be statistically independent such that the covariance matrix $\mathbf{R}_Y(f,k)$ should be a

diagonal matrix. Then, the BSS problem can be recognized as the least square optimization problem which is formulated as follows [86].

$$\mathbf{W}_{\texttt{opt}}(f) = \arg\min_{\mathbf{W}(f)} \sum_{n=1}^{N} \|\mathbf{E}(f,n)\|_{\mathrm{F}}^2 \tag{3.9}$$

where

$$\begin{aligned}\mathbf{E}(f,n) &= \mathbf{R}_Y(f,n) - \mathbf{\Lambda}(f,n) \\ &= \mathbf{W}(f)\mathbf{R}_X(f,n)\mathbf{W}(f)^H - \mathbf{\Lambda}(f,n)\end{aligned}$$

and $\|\cdot\|_{\mathrm{F}}^2$ is the squared Frobenius norm operator. $\mathbf{\Lambda}(f,n)$ is a diagonal matrix in which each diagonal element represents the power of each source. If the filed is homogeneous, all sources have the same power. Then, $\mathbf{\Lambda}(f,n) = S(f,n)\mathbf{I}$ where (f,n) is the power constant of source. The homogeneous assumption can be achieved by pre-whitening operation in [33].

With the implementation of the gradient descent method, the gradient update scheme with respect to the optimization problem in Equation (3.9) is denoted by

$$\begin{aligned}\mathbf{W}^{(m+1)}(f) = \mathbf{W}^{(m)}(f) - 2\mu \sum_{n=0}^{N-1} \big[&\mathbf{W}^{(m)}(f)\mathbf{R}_X(f,n)\mathbf{W}^{(m)^{\mathrm{H}}}(f) \\ &- \mathbf{\Lambda}(f,n)\big]\mathbf{W}^{(m)}(f)\mathbf{R}_X(f,n)\end{aligned} \tag{3.10}$$

where μ is the step-size and m is the iteration index. The convergence analysis of this method has been illustrated in [77].

3.4 Hybrid descent method

As we mentioned in Section II, the output of the convolutive BSS model depends on both the mixing system and the separation system. Given a fixed mixing system, the separation system is optimized using the SOS method. In this section, we introduce a hybrid descent method to optimize the mixing system via the change of

the sensor locations. The propagation of speech sound in reverberant environment is not instantaneous but rather a convolutive mixtures as a result of different room impulse responses (RIRs) between each source speech and microphone. Given the information of the speaker locations, microphone locations and other coefficients of acoustic scene, RIRs related to a small room can be modelled by using the image method [68]. By the adoption of RIRs, the mixing process of acoustic signals can be finished in Equation (3.2). In reality, however, we can merely receive the mixed signals rather than derive the explicit form of RIRs due to the lack of speaker locations or coefficients of acoustic scene.

In order to optimize the mixing system without the knowledge of RIRs, we formulate the optimization problem as

$$\delta_{\text{opt}} = \arg\max_{\delta \in \mathbf{\Delta}} \; g(\delta) \tag{3.11}$$

$$\text{s.t.} \;\; \mathbf{L}_{\mathbf{\Delta}} \leqslant \mathbf{\Delta} \leqslant \mathbf{U}_{\mathbf{\Delta}} \tag{3.12}$$

where $g(\delta) = \rho(\mathbf{y}_m, \mathbf{s}_0)$; $\mathbf{\Delta} = \{\delta_1, \delta_2, \ldots, \delta_M\}$ representing the set of location variables for all the microphones; $\mathbf{L}_{\mathbf{\Delta}}$ and $\mathbf{U}_{\mathbf{\Delta}}$ are respectively the lower bound and upper bound of location variables which depend on the room dimension; $\rho(\mathbf{y}_m, \mathbf{s}_0)$ is the correlation function between $\mathbf{y}_m$ and $\mathbf{s}_0$; $\mathbf{y}_m$ is m-th output of the covolutive BSS model and $m \in [1, 2, \ldots, M]$; and $\mathbf{s}_0$ is the desired speech. Given an candidate solution $\tilde{\delta}$, convoltive BSS method generates the optimal separated signals using the cooresponding mixtures $\mathbf{x}_{\tilde{\delta}}(t)$. Since the mixing matrix is unknown in the optimization process, the gradient related to the problem in Equation (3.11) cannot be expressed in an explicit form. Therefore, its global maxima cannot be directly located using a gradient-based approach and a better global optimization technique should be developed [113].

There are two major points to note when finding the global optima. The solution should avoid local optima and the optimization technique has the speed of conver-

gence to approach stationary points. The heuristic method could find the global optima due to its capability of avoiding the local optima [101]. In reality, it is used to find the global optima without pre-defined precision due to its slow convergence rate. One of the reasons for the slow convergence is because the heuristic method becomes slow when it tries to approach or descent to stationary points. On the other hand, a gradient-based numerical method is much more efficient in finding a stationary point such that it can be used to speed up the local search. Since the location variables impact the objective value via the change of the RIR vector, it is difficult to write an explicit form of Equation 3.11 with respect to the location variables. The optimization problem is highly nonlinear and is essentially nonconvex with respect to the location variables. To solve the aforementioned issues, this chapter proposes an efficient hybrid descent algorithm embedding a gradient-based numerical optimization algorithm into the heuristic method. The heuristic method is used to locate a descent point from a previous converged local solution and the gradient-based numerical algorithm is used to find better local optima. By doing so, we can retain the robustness of the heuristic method and the convergence speed of gradient-based numerical method.

There is a variety of heuristic methods such as the genetic algorithm [30], Tabu search algorithm [2], simulated annealing algorithm [41] and so forth. The main difference between these methods is the way to traverse the whole parametric space to reach a global peak in the case of unevenly distributed, nonuniform, multiple-peak space [95]. In this chapter, we propose the use of genetic algorithm. The genetic algorithm was developed in [71] to solve the sensor placement problem in the beamformer configuration design. Comparing to others, the genetic algorithm has a nice parallel computation structure in which the candidate solution is generated by the random perturbation of a population rather than by moving from one point to the next. Apart from that, multiple initial points enable the GA algorithm to traverse

the whole parametric space more efficiently. The whole optimization scheme for the BSS model is stated in Algorithm 2. The details of genetic algorithm are illustrated in Algorithm 3.

Algorithm 2 Hybrid descent method for the BSS model

Set the hyperparameters for the genetic Algorithm and $T = 0$. Create a random initial solution δ^{Initial}.
repeat
    Execute the genetic algorithm to get the improved solution $\tilde{\delta}$.
    Starting from $\tilde{\delta}$, execute the sequential quadratic programming (SQP) method to derive the local optima δ_{opt}.
    if $g(\delta^{\text{Initial}}) < g(\delta_{\text{opt}})$ **then**
        $\delta^{\text{Initial}} = \delta_{\text{opt}}$;
        $g(\delta^{\text{Initial}}) = g(\delta_{\text{opt}})$;
        $T = T + 1$;
    else
        break;
    end if
until $T > T_{\max}$ where $T_{\max}$ is the hyperparamter denoting the maximum iteration number when $\mathbf{\Delta}$ is estimated.

Algorithm 3 Genetic algorithm

$g(\tilde{\delta}) = g(\delta^{\text{Initial}})$; $\tilde{\delta} = \delta^{\text{Initial}}$; $J = 1$;
Generate the initial population via the random perturbations for the initial solution δ^{Initial}.
repeat
    Evaluate the objective function for individuals on the J-th generation.
    Scale the values of objective function.
    Store the optimize solution of the J-th generation $\delta^{(J)}$.

7: Generate the elite population $\mathcal{E}$.

8: Generate the sub-population $\mathcal{C}$ using Crossover operation.

9: Generate the sub-population $\mathcal{M}$ using Mutation operation.

10: Construct the population of the next generation as $\{\mathcal{E}; \mathcal{C}; \mathcal{M}\}$.

11: **if** $g(\tilde{\delta}) < g(\delta^{(J)})$ **then**

12: $\tilde{\delta} = \delta^{(J)}$;

13: $g(\tilde{\delta}) = g(\delta^{(J)})$;

14: **end if**

15: $J = J + 1$.

16: **until** Stopping criteria have been satisfied.

In Step 4 of Algorithm 3, we first optimize the unmixing matrix $\mathbf{W}(f)$ for each individuals in the population using Equation (3.10). Then, we calculate the maxima of $\rho(\mathbf{y}_m, \mathbf{s}_0)$ where $m \in [1, 2, \ldots, M]$.

In Steps 7-10 of Algorithm 3, the population of the next generation is generation. The details are stated below.

- The elite population consists of the individuals with the objective function value great than or equal to the B_1-th percentile with respect to its current population.
- The Crossover operation is to combine the vector of a pair of individuals whose objective function values are between the B_1-th and B_2-th percentiles with respect to its current population.
- The mutation operation is to introduce the random permutation to each individual whose objective function values are less than B_3-th percentile.

In Step 15 of Algorithm 3, there are two stopping criteria stated below.

- $J > J_{\max}$ where $J_{\max}$ is the hyperparameter denoting the maximum generation number in GA.

- $J_{\text{stall}} < J \leqslant J_{\max}$ and $\frac{1}{J-J_{\text{stall}}}\sum_{j=J-J_{\text{stall}}}^{J}[g(\delta^{(j)}) - g(\delta^{(j-1)})] < \zeta$ where ζ is the hyperparameter denoting the function tolerance of the average change of the objective function value in the last J_{stall} generations.

It is noted that the generated population should satisfy the constraint in Equality 3.12 if the random perturbations are introduced to both Step 2 and 9 of Algorithm 3.

3.5 Simulation

3.5.1 Experimental settings

In this section, we illustrate the performance of our convolutive BSS method in a simulated room. We consider a 6m × 6m × 3m square office room with a RT T_{60} chosen as 0ms, 100ms, 200ms, 300ms, 400ms and 500ms. Figure 3.2 displays our acousitc scene. The heights of microphones and sources are 1.5 meters. The horizontal coordinates of signal of interest (SOI) and interference signal (INT) are $(1.5, 4.5)$ and $(1.5, 1.5)$ respectively. The number of microphone in the BSS model is chosen as 2, 3 and 4. The horizontal coordinates of microphones are varied in the area encircled by the black dashed line. Furthermore, both source speech and interference speech contain $6.25s$ voice signals sampled at 8kHZ. The objective function of optimizing convolutive BSS model is the interference suppression. Define $\hat{P}_Y(\omega)$ as the spectral power estimate of the source signal; $\hat{P}_{\tilde{Y}}(\omega)$ as the spectral power estimate of the output of BSS; $\hat{P}_{Y_I}(\omega)$ as the spectral power estimate of the interference speech; $\hat{P}_{\tilde{Y}_I}(\omega)$ as the spectral power estimate of the output of BSS when the interference speech is active alone. The normalized interference suppression is defined by

$$\texttt{SUPP}_{\texttt{I}} = 10 \times \log_{10} \frac{\int_{-\pi}^{\pi} \hat{P}_{\tilde{Y}_{\texttt{I}}}(\omega) d\omega}{C_d \int_{-\pi}^{\pi} \hat{P}_{Y_{\texttt{I}}}(\omega) d\omega}$$

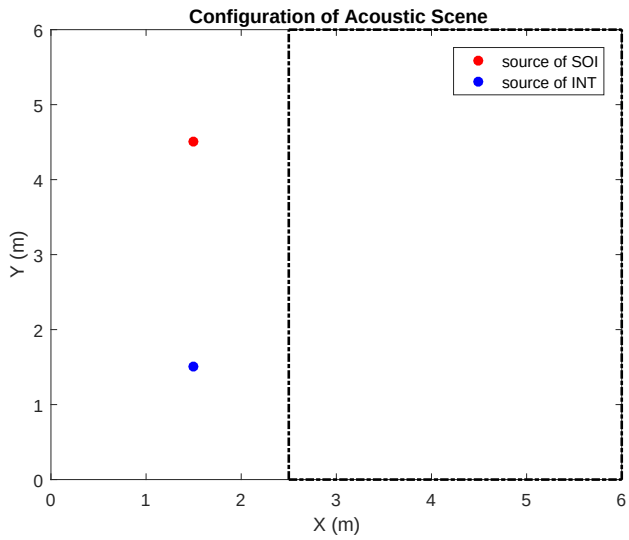


Figure 3.2: Configuration of acoustic scene

where C_d is defined by

$$C_d = \frac{\int_{-\pi}^{\pi} \hat{P}_Y(\omega) d\omega}{\int_{-\pi}^{\pi} \hat{P}_{\tilde{Y}}(\omega) d\omega}.$$

We consider the convolutive BSS model containing 2, 3 or 4 microphones. The initial horizontal coordinates of microphone array are shown in table 3.1. All the microphones can be varied among the the area encircled by the black dashed line in Figure 3.2.

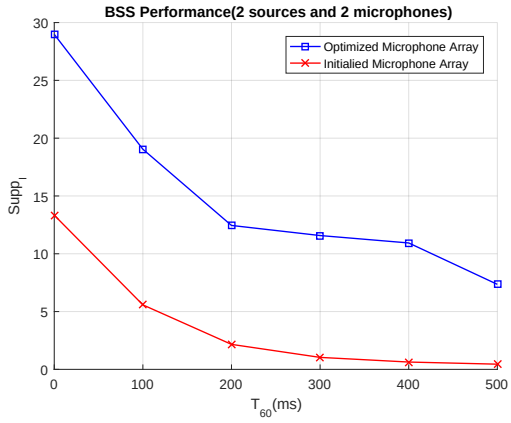
2 Microphones	3 Microphones	4 Microphones
(3,1.5)	(3,1.5)	(3,1.5)
(3,4.5)	(3,3)	(3,2.17)
-	(3,4.5)	(3,3.83)
-	-	(3,4.5)

Table 3.1: Initial horizontal coordinates of microphone array

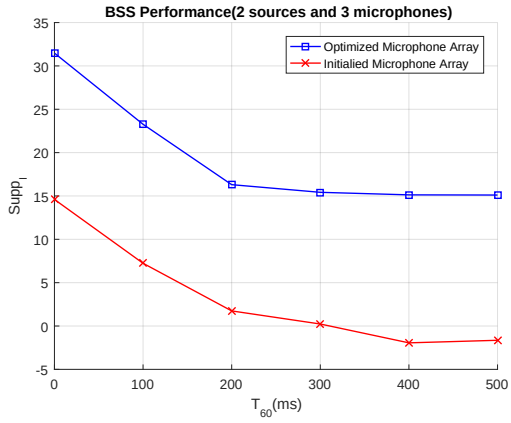
3.5.2 Simulation results

Figure 3.3a, 3.3b and 3.3c demonstrate the suppression measure of the interference signals when the convolutive BSS model is applied with both the optimized position

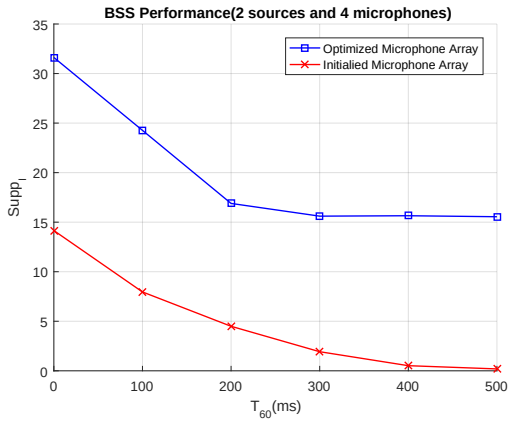
and the initialized position of the microphone array. It is observed that the convolutive BSS model can be improved significantly with the implementation of our optimization method for the microphone placement.



(a) Microphone array contains 2 microphones



(b) Microphone array contains 3 microphones



(c) Microphone array contains 4 microphones

Figure 3.3: Suppression measure of interference signal with respect to the BSS model

Figure 3.4 is the summary plot of BSS performance. The dash lines denote the $\mathtt{SUPP_I}$ value with respect to the initial microphone arrays in Table 3.1. The solid lines denote the $\mathtt{SUPP_I}$ value with respect to the optimized microphone arrays derived from Algorithm 2. For either the initial microphone arrays or the optimized microphone arrays, the BSS performance reduces with the increasing of the reverberation time T_{60}. In the presence of acoustic reverberation, the optimized microphone array with

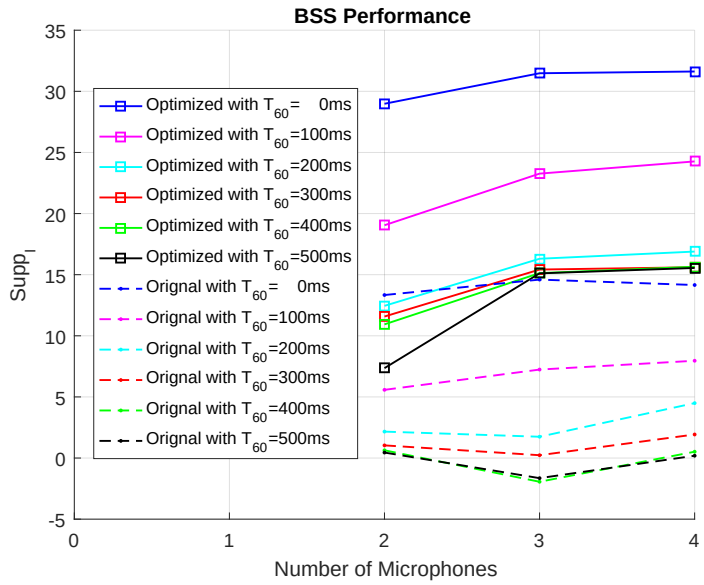


Figure 3.4: Summary plot for the suppression measure of interference signal

2 microphones can beat all the initial microphone arrays.

Figure 3.5-3.7 show the optimized locations of microphones when the number of microphones is chosen as 2, 3 and 4. It is noted that the optimized microphone array tends to cluster around the interference source when T_{60} is less than or equal to 200ms. Taking advantage of close placement, the BSS system can achieve a significant suppression for the interference signal. When T_{60} is greater than 200ms, more and more acoustic signals are reflected by the walls and the optimized microphone array spreads out. This is because more and more reverberant signals need to be suppressed. This is consistent to that reported by [5] where BSS tries to suppress the interference signal by forming a null on it. As the reverberation time increases, then the array element spreads itself. Interestingly, the optimized geometry is consistent with the suppression performance for the case of T_{60} equaling 400ms and 500ms, where very little improvement was observed. This can be understood from the optimized location for 2, 3 and 4 microphones for both RTs as they are very similar with the converged geometry.

3.5.3 Performance analysis

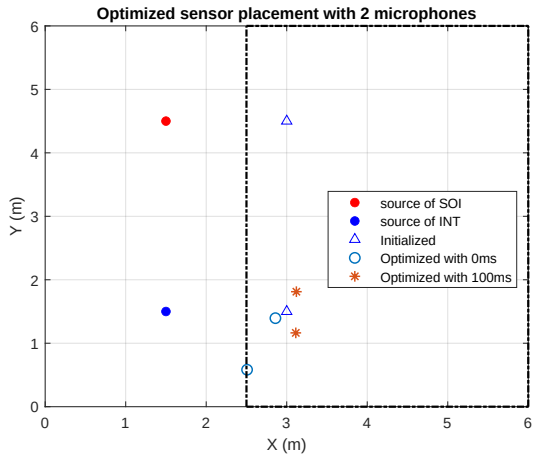
In this subsection, we first study how the uncertainty in the performance of the BSS model by the change of microphone locations. We consider the case that the number of microphones is three and the reverberant time T_{60} is $200ms$. Figure 3.8a is the corresponding optimized placement. We consider three cases:

- Case 1: Microphone 1 can be varied among the area encircled by the black dashed line in Figure 3.8a. Microphone 2 and 3 are fixed in Figure 3.8a;
- Case 2: Microphone 2 can be varied among the area encircled by the black dashed line in Figure 3.8a. Microphone 1 and 3 are fixed in Figure 3.8a;
- Case 3: Microphone 3 can be varied among the area encircled by the black dashed line in Figure 3.8a. Microphone 1 and 2 are fixed in Figure 3.8a.

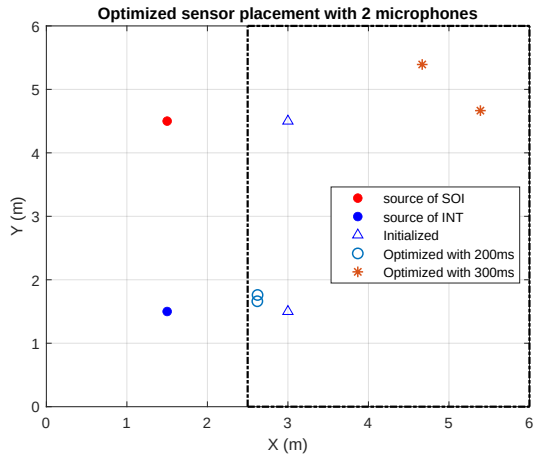
Figure 3.8b, 3.8c and 3.8d are the mesh plots with respect to the above three cases. It is evident in Case 1 and 2 that the suppression measure drops down rapidly if the microphone location deviates from the optimal location. The BSS model is very sensitive to the change of microphone locations. In case 3, the BSS model looks like more robust. A good performance can be retained when microphone 3 is varied in a small region. From these three mesh plots, we can see that the BSS model behaves very nonlinearly when the microphone moves in the acoustic scene. The optimization of microphone placements in BSS model can be possibly trapped in the local optima [9]. By using our proposed hybrid descent method, the optimized placement can archive the maximum value of suppression measures.

3.5.4 Discussion

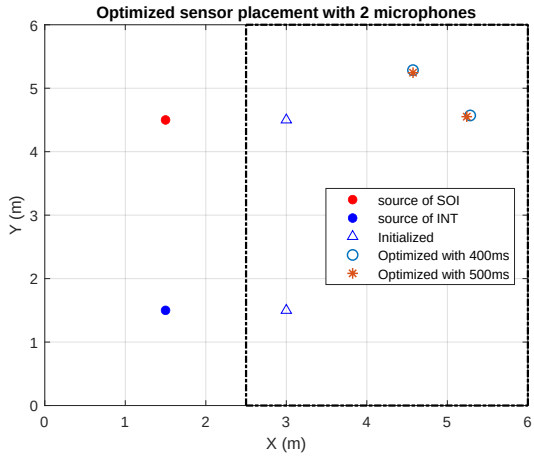
From the experimental results, we can observe that a bigger reverberation time leads to a worse separation performance. The main reason is that the number of reflected



(a) $T_{60} = 0ms,\ 100ms$



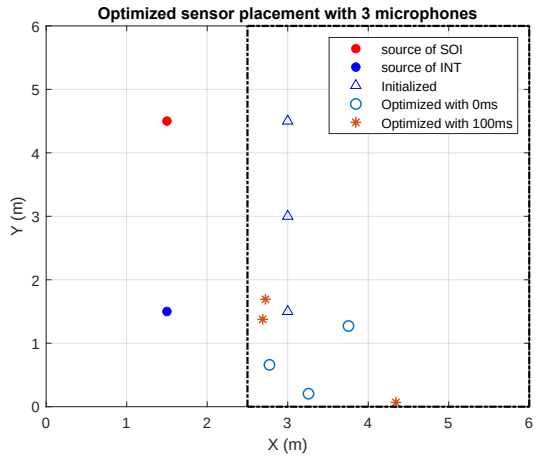
(b) $T_{60} = 200ms,\ 300ms$



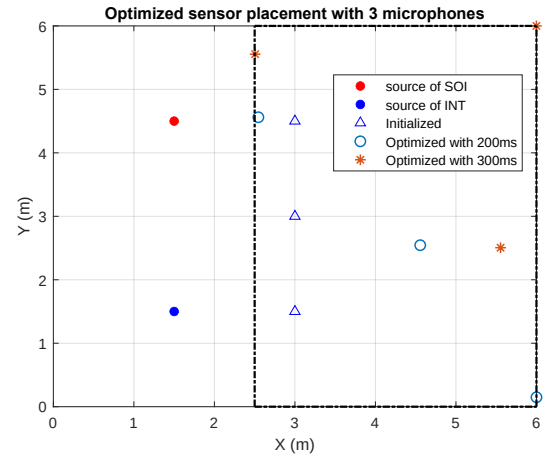
(c) $T_{60} = 400ms,\ 500ms$

Figure 3.5: Optimized locations of 2 microphones with the reverberation time T_{60} chosen as $0ms$, 100ms, 200ms, 300ms, 400ms and 500ms

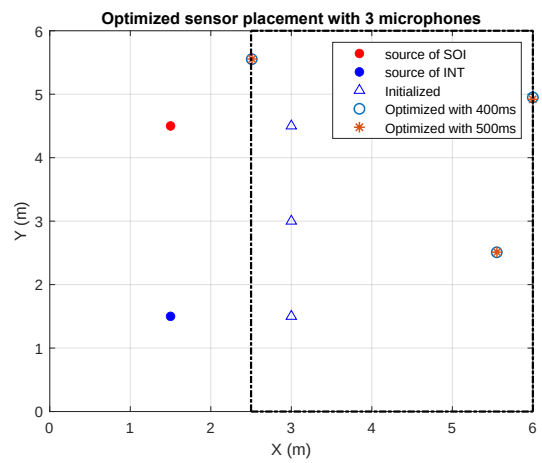
signals increases as the reverberation time increases. The overdetermined BSS system quickly reverts to the underdetermined BSS system. Moreover, the separation problem becomes more difficult as many more sources need to be separated. Reverberant acoustic signals hamper us to extract the signal of interest from the mixing system. Given a fixed reverberation time T_{60}, it can be observed that the performance of optimized BSS system becomes better with the increment of number of microphones. However, without the optimization of microphone locations, system



(a) $T_{60} = 0ms,\ 100ms$



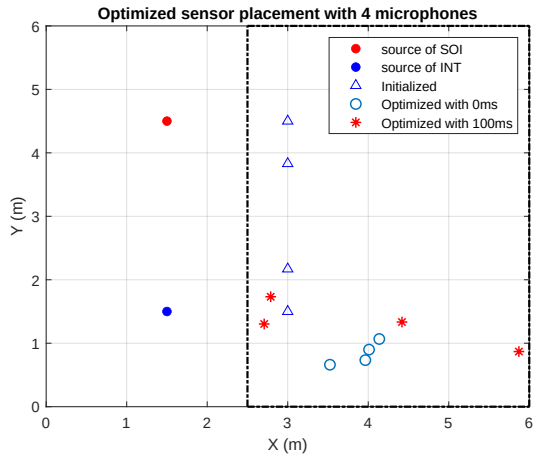
(b) $T_{60} = 200ms,\ 300ms$



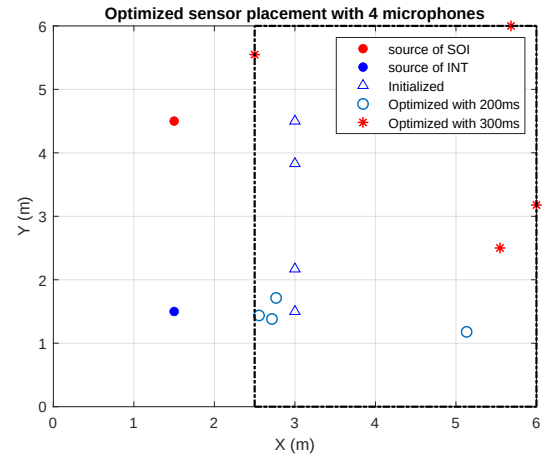
(c) $T_{60} = 400ms,\ 500ms$

Figure 3.6: Optimized locations of 3 microphones with the reverberation time T_{60} chosen as $0ms$, 100ms, 200ms, 300ms, 400ms and 500ms

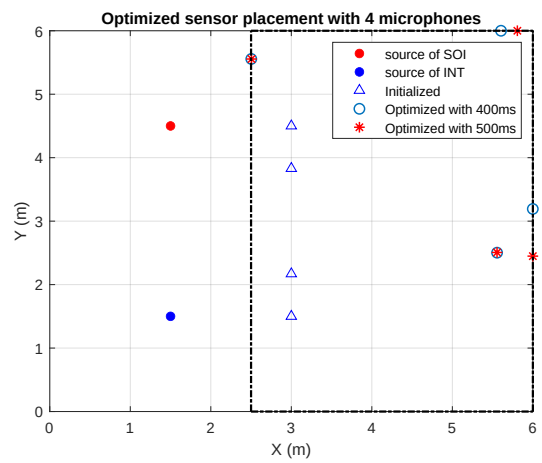
performance cannot be simply improved by increasing the number of microphones. Moreover, an inappropriate placement of microphone array can undermine the BSS system. For example, when the reverberation time T_{60} is chosen as $200ms$, $300ms$, $400ms$ and $500ms$, the initial microphone array with 3 microphones has a worse performance than the initial microphone array with 2 microphones. On the other hand, the optimized microphone array consisting of 2 microphones can easily outperform the initialized microphone array containing 4 microphones when the acoustic



(a) $T_{60} = 0ms,\ 100ms$



(b) $T_{60} = 200ms,\ 300ms$

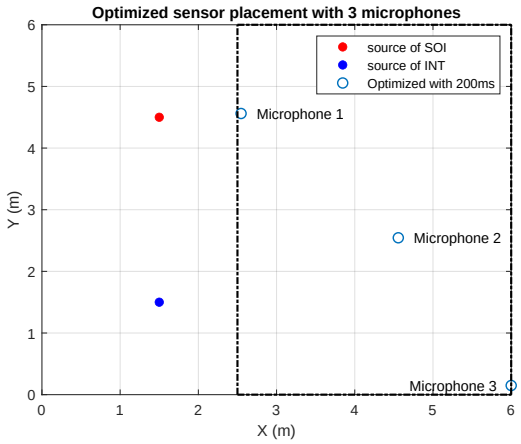


(c) $T_{60} = 400ms,\ 500ms$

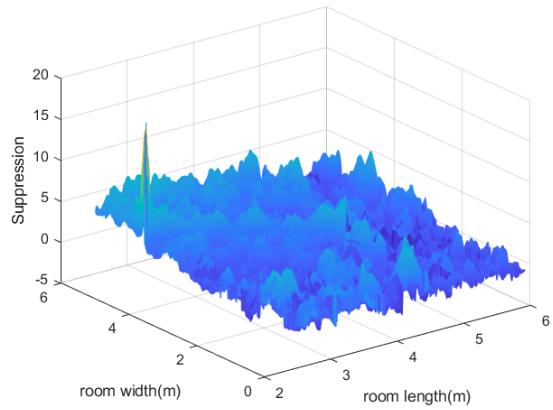
Figure 3.7: Optimized locations of 4 microphones with the reverberation time T_{60} chosen as $0ms$, 100ms, 200ms, 300ms, 400ms and 500ms

reverberation exists.

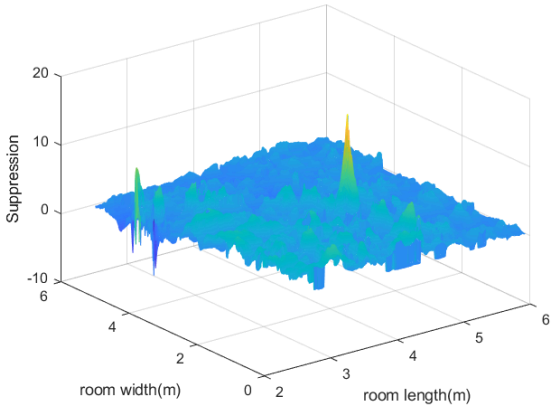
The optimized microphone arrays have a slight pattern. For a lower reverberation time (0 and 100ms), the microphones cluster closer to the interference source. The BSS model behaves like a null beamformer. As reverberation time increases, the microphones is more scattered as more and more reverberant signals are deemed to the interference sources. Up to a certain reverberation time (400ms and 500ms), the change of optimized microphone arrays is hardly observed. It is more obvious when



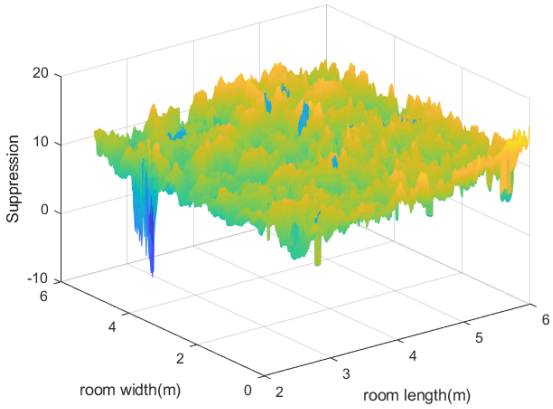
(a) Optimized locations of 3 microphones with a reverberation time T_{60} chosen as 200ms



(b) Case 1



(c) Case 2



(d) Case 3

Figure 3.8: Mesh plots of suppression measure when no. of microphone is 3 and reverberant time $T_{60} = 200ms$

the array contains 2 or 3 microphones as there is even less degrees of freedom, akin to saturate very earlier.

3.6 Final remark

In this chapter, we propose an enhancement method for the performance of BSS model via the optimization of microphone locations when the nonlinear mixing problem exists. Spatial information can be fully exploited via an optimized array geom-

etry. In order to solve the nonlinear mixing problem, we introduce a novel hybrid descent method to find the location optima for each microphone. Results show that the optimized array consistently yield a greater suppression across the different reverberation time compared to an unoptimized linear configuration.

Chapter 4

Optimal assignment of constellations with a novel QAP formulation

The design of the constellation plays an important role in the communication system, especially in digital signal transmission. In implementation process of the digital signal transmission, one need to map a binary sequence to a symbol, which is categorized as an assignment process. In this chapter, we propose a novel framework in which the symbol assignments in constellations are formulated as a quadratic assignment problem (QAP) which can be solved by the combinatorial optimization technique. Due to the nonlinear property of the QAP, the assignment problem in constellations can be a nondeterministic polynomial-time hard (NP-hard) problem when a large number of symbols are inside the constellation. In order to tackle the QAP, apart from introducing the traditional heuristic algorithm, such as the greedy randomized adaptive search procedure (GRASP) algorithm, we propose a tailor-made algorithm based on the simulated annealing method. Simulation results show that our proposed algorithm can derive the optimal assignment solution for different constellations.

4.1 Research background

Constellations, which intuitively show the mapping rule between the bit-sequence and the modulated signal, play an important role in communication society. In a typical constellation, the angle of measured counterclockwise from the x-axis represents the phase shift of the carrier wave and the polar diameter $||x(t), y(t)||$ is the measurement of the signal power. In the digital system, binary information can be mapped to the symbols in constellations via the modulation process. After being modulated by the sender, signals are transmitted via the communication channel, i.e., the additive white Guassian noise (AWGN) channel. After receiving the modulated signal, the receiver should demodulate (or decode) the signals according to the specified mapping rule between the symbols and the bit-sequences. Due to the existence of noise, such as the additive white Guassian noise, the received signal is contaminated by noise so as to be different to the original signal transmitted from the sender. Noisy signals lead to an inaccurate demodulation result such that the error arises. The transmission error not only depends on the noise magnitude (or quantity) but also is related to the mapping rule. A good mapping rule can show significant difference in ultimate performance of the communication system. The objective of constellation design is to minimize the transmission error via the optimization of the mapping rule between the bit-sequences and the constellation symbols.

The design method of constellations can be divided into two main categories including geometric constellation shaping (GCS) [58] and probabilistic constellation shaping (PCS) [32]. The former one mainly focuses on either the optimization of symbol positions inside the constellation or the acquisition of mapping rule with good performance. In the latter one, under the assumption that the transmission frequency for each bit-sequence is different, the bit-sequence with high transmission frequency is assigned to the symbol with small energy. In both design methods,

constellation diagrams are designed to reduce the error proportion in transmission process. Normally, the error proportion is measured by the bit-error-rate (BER) which is the fraction of dividing the number of error bits by the total number of transmission bits. It is reasonable to reduce the expectation (expected value) of BER for the designed constellation rather than the BER in terms of a short random bit-sequences. However, the common practice to calculate the expectation of BER is to use the sampling method (Monte Carlo method). To be more specific, a series of randomly generated bit-sequences should be transmitted in channel simulator (or in real channel), the estimate of BER is derived by counting the number of the error bits. In order to obtain an accurate BER value, it is usually necessary to transmit a large number of bit-sequences until the statistical estimate of BER tends to be stable. Especially for the case with a larger signal-to-noise ratio (SNR), it needs to randomly generate more bit-sequences because the value of BER can be very small. The sampling method is suitable to the verification of the designed constellation diagram because it is very simple. Since the sampling method is very time-consuming,however, it is undesirable for the optimization process in which the value of BER are repeatedly calculated for different mapping rules.

In order to kick off the optimization of constellations in terms of BER, we proposed a novel method to compute the BER by formulating it with symbol error rate (SER) and Hamming distance which will be illustrated in details in next section. Meanwhile, we propose an approximation method to calculate SER matrix. The benefit of doing this is not only to speed up the computation progress, but also to construct a quadratic assignment problem (QAP) form for the constellation design. QAP is one of the most classical combinatorial optimization problems. It belongs to the category of the facilities location problem which is first introduced by Koopmans and Beckmann in 1957 [62]. QAP has been applied in various fields such as travelling salesman problem (TSP) [23], backboard wiring problem [100], campus

planning problem [39], typewriter keyboard design problem [24], hospital planning problem [44] and so forth. In addition, its computational complexity has also been studied by some scholars [67, 74, 21, 26, 22]. QAP is an NP-hard problem [48]. The exact algorithms cannot attain the polynomial computation time to solve QAP in the worst case. When the dimension of the QAP is large, exact algorithms are not practical because exact algorithms employ a lot of time to marginally improve the performance of solutions after a good incumbent solution has been derived. Therefore, heuristic algorithms are proposed to solve QAPs, including greedy randomized adaptive search procedures (GRASP) algorithm [45] and simulated annealing (SA) algorithm [50] [1]. The SA algorithm is the stochastic optimization technique.

It is noted that, for complex constellations, the SER matrix does not stay the same since the total energy of the chosen symbols is changeable. In the classical QAP framework, both the distance matrix and the flow matrix are assumed to be fixed during the whole search procedure. To solve this problem, we propose a novel SA-based framework where the SA framework is extended to a multi-tier search. Among them, SER matrix is changed in outer search, while it remains unchanged in inner loop to improve search efficiency.

The rest of this chapter is organized as follows. Section 4.2 shows the QAP formulation. In Section 4.3 and 4.4, we illustrate the search framework with GRASP and SA-based algorithms for different types of assignment problems along with the different frameworks to compute BER. In Section 4.5, numerical simulations are demonstrated to verify both the approximation computation of BER and the proposed search framework. Section 4.6 is the final mark of this chapter.

4.2 QAP formulation for BER

The quadratic assignment problem is set up with an objective function which measures the influence exerted from the flow or distance between different facilities. For each facility, its contribution to the objective function stems from the relative flow or distance with others. Therefore, it is straightforward to formulate the QAP as a summation form in connection with quantities of both flow and distance. i.e.,

$$\min_{\phi \in S_n} \sum_{i=1}^{n} \sum_{j=1}^{n} f_{ij} d_{\phi_{(i)}\phi_{(j)}}, \tag{4.1}$$

where $i, j \in N = \{1, 2, \ldots, n\}$ denote the i-th and j-th facilities, respectively; S_n is the set of all permutations $\phi : N \to N$; $\phi(i)$ and $\phi(j)$ denote the locations where the i-th and j-th facilities are placed; f_{ij} measures the flow between the i-th and j-th facility; $d_{\phi(i)\phi(j)}$ stands for the distance between location $\phi(i)$ and $\phi(j)$.

In practice, a matrix form of permutation is introduced. Define an $n \times n$ permutation matrix $X = (x_{ik})$ where

$$x_{ik} = \begin{cases} 1, & \text{if } \phi(i) = k, \\ 0, & \text{otherwise.} \end{cases}$$

Denote $A = (f_{ij})$ as the flow matrix and $B = (d_{ij})$ as the distance matrix. One can obtain that

$$\sum_{i=1}^{n} \sum_{j=1}^{n} f_{ij} d_{\phi_{(i)}\phi_{(j)}} = \langle B, XAX^\top \rangle$$

where $\langle \cdot, \cdot \rangle$ represents the inner product. Therefore, the QAP can be formulated as

$$\min_{X \in X_N} \quad \langle B, XAX^\top \rangle, \tag{4.2}$$

where X_N denote the set of all the permutation matrices for the set N.

On the other hand, BER represents the expected proportion of error bits in the transmitted bit-sequences. In general, since each symbol has the same probability of being transmitted, BER represents the average percentage of error-bits for all the symbols. For one transmitted symbol, the number of error-bits is equal to the cumulative number of error-bits which are transmitted to other symbols, i.e.,

$$\mathrm{BER}(s_i) = \frac{e_b(s_i, s_0) + \ldots + e_b(s_i, s_{i-1}) + e_b(s_i, s_{i+1}) + \ldots + e_b(s_i, s_{n-1})}{n_b(s_i)} \tag{4.3}$$

where $n_b(s_i)$ is the total number of bits transmitted as symbol s_i and $e_b(s_i, s_j),\ j = 0, \ldots, n-1$, denotes the number of error bits in the case that symbol s_i is transmitted and the received symbol is s_j. Mathematically, $e_b(s_i, s_j)$ can be computed by

$$e_b(s_i, s_j) = e_s(s_i, s_j) d_{B_i, B_j} \tag{4.4}$$

where $e_s(s_i, s_j)$ is the number of times that symbol s_i is wrongly received as symbol s_j; and d_{B_i,B_j} is the Hamming distance between the bit-sequences B_i and B_j which are assigned to s_i and s_j respectively. The value of d_{B_i,B_j} can be derived by

$$d_{B_i,B_j} = \sum_{k=1}^{N} |B_i(k) - B_j(k)|, \tag{4.5}$$

where $B_i(k)$ and $B_j(k)$ are the kth binary variable in the bit-sequence B_i and B_j, respectively; and N is the length of the bit-sequences. Therefore, Equation (4.3) becomes

$$\mathrm{BER}(s_i) = \frac{\sum_{j=0}^{n-1} e_s(s_i, s_j) d_{B_i,B_j}}{n_b(s_i)} = \frac{1}{N} \sum_{j=0}^{n-1} \frac{e_s(s_i, s_j)}{n_b(s_i)/N} d_{B_i,B_j}. \tag{4.6}$$

Note that $\frac{e_s(s_i,s_j)}{n_b(s_i)/N}$ is the expectation of the error that symbol s_i is wrongly received as symbol s_j, which is also known as the symbol-error-rate (SER).

Compared to the calculation of BER, in the AWGN channel, SER can be obtained via the integral of probability density functions. The received signal, in the AWGN

channel, can be denoted by

$$r(t) = s(t) + n(t),$$

where $n(t)$ is the additive white Gaussian noise with mean zero and variance $N_0/2$, and $n(t)$ is independent of $s(t)$. When it comes to the demodulation phase, all the Euclidean distances between the received signal and the symbols in the constellation diagram are computed. The received signal is categorized as symbol s_i when the Euclidean distance between the received signal and s_i is smallest. Thus, each symbol in the constellation has its own region in which signals should be categorized to it. Given a certain symbol, the SER could be computed by the cumulative probability of the Gaussian distribution. For a symbol s_{i_0} to be transmitted, SER is calculated by

$$P(s_i|s_{i_0}) = \iint\limits_{s \in D_{s_i}} p\big(s|s_{i_0}\big)\mathrm{d}x\mathrm{d}y \tag{4.7}$$

where D_{s_i} is the integral region of s_i denoted as

$$D_{s_i} = \{s\big|||s - s_i|| < ||s - s_j||, j \in \{1, \ldots, n\},\ i \neq j)\};$$

and $p\big(s|s_{i_0}\big)$ is the probability density function of the bivariate Gaussian distribution with the mean vector $(x(s_{i_0}), y(s_{i_0}))^\top$ and variance matrix $\Sigma = \mathrm{diag}(N_0/2, N_0/2)$.

By the substitution of Equation (4.7) into Equation (4.6), we can write the BER as

$$\mathrm{BER}(s_i) = \frac{1}{N}\sum_{j=0}^{n-1} P(s_j|s_i) d_{B_i,B_j}. \tag{4.8}$$

It is noted that $P(s_j|s_i)$ is the element in the $(i+1)$-th row and $(j+1)$-th column of the SER matrix; and d_{B_i,B_j} is the element in the $(i+1)$-th row and $(j+1)$-th column of the Hamming distance matrix. As a result, the total BER can be computed by

$$\mathrm{BER} = \frac{1}{nN}\sum_{j=0}^{n-1}\sum_{i=0}^{n-1} \mathrm{BER}(s_i|s_j) = \frac{1}{nN}\sum_{j=0}^{n-1}\sum_{i=0}^{n-1} P(s_i|s_j) d_{s_i,s_j}. \tag{4.9}$$

If we compare Equation (4.9) with QAP, we can see that the constellation assignment problem can be a QAP if we denote the Hamming distance matrix as A and the SER matrix as B, i.e.,

$$A = \begin{bmatrix} d_{s_0,s_0} & d_{s_1,s_0} & \cdots & d_{s_{n-1},s_0} \\ d_{s_0,s_1} & d_{s_1,s_1} & \cdots & d_{s_{n-1},s_1} \\ d_{s_0,s_2} & d_{s_1,s_2} & \cdots & d_{s_{n-1},s_2} \\ \vdots & \vdots & \ddots & \vdots \\ d_{s_0,s_{n-1}} & d_{s_1,s_{n-1}} & \cdots & d_{s_{n-1},s_{n-1}} \end{bmatrix}. \tag{4.10}$$

$$B = \begin{bmatrix} P(s_0|s_0) & P(s_1|s_0) & \cdots & P(s_{n-1}|s_0) \\ P(s_0|s_1) & P(s_1|s_1) & \cdots & P(s_{n-1}|s_1) \\ P(s_0|s_2) & P(s_1|s_2) & \cdots & P(s_{n-1}|s_2) \\ \vdots & \vdots & \ddots & \vdots \\ P(s_0|s_{n-1}) & P(s_1|s_{n-1}) & \cdots & P(s_{n-1}|s_{n-1}). \end{bmatrix} \tag{4.11}$$

4.3 BER computation for constellations of constant total energy and GRASP algorithm

In this section, regular constellations, such as QAM constellations, are considered. In many cases, the constellation design problem is a standard assignment problem which means the number of candidate symbols is the same as the number of bit-sequences to be assigned. In this case, the position of selected symbol is fixed, which means that the elements of the B matrix is not change as long as we determine the SNR value. The critical point is how to calculate the B matrix as accurate as possible. For small instances, the sampling method can be implemented to calculate B matrix. For large instances, it is desirable to apply the integral method to compute the SER matrix. It is because, in most cases, the shape of the constellation map is regular and the integral region is easy to determine. Recall the definition of SER in Equation (4.7), the SER value is essentially computed by a double integral. Since

the received signal follows a Gaussian distribution, the form of integrand is given by

$$p\big(s|s_{i_0}\big) = \frac{1}{2\pi\sigma^2}\exp\left\{-\frac{(x(s)-x(s_{i_0}))^2+(y(s)-y(s_{i_0}))^2}{2\sigma^2}\right\}. \tag{4.12}$$

Note that the probability density function is independently identically distributed in both directions, and they are both normally distributed. The binary integral expression of SER can be written as a simple expression of $erfc(\cdot)$ function when the integral area is a rectangle, so it can be calculated quickly in this situation. In addition, the Hamming distance matrix A is quite easy to compute. With A and B in hand, we can solve it directly by adopting traditional methods for QAP.

Since QAP is an NP-hard problem when the number of considered facilities is large, we kick off the searching process in a heuristic way. For example, we apply the greedy randomized adaptive search procedure (GRASP) algorithm to solve the QAP in terms of the constellation design [85]. The GRASP algorithm is an iterative algorithm with each GRASP iteration consisting of two phases: the construction phase and the local search phase. The construction phase aims to find out a relatively good assignment as the initial solution in the local search phase. The local search phase focuses on optima-seeking via the randomly exchange of assigned bit-sequences between neighbor symbols.

When the assignment problem is linear, each assignment operation is independent with others such that one assignment cannot affect the decision of other assignments. If the constellation design problem is linear, two bit-sequences with the smallest Hamming distance should be separately assigned to two symbols which have the biggest SER. Unfortunately, the constellation design is QAP which is a nonlinear assignment problem but we can use this idea to construct the first assignment. We sort the Hamming distances in an increasing order and the SER values in a decreasing order, i.e.,

$$d_{i_1,j_1} \leqslant d_{i_2,j_2} \leqslant \ldots \leqslant d_{i_n,j_n}, \quad P(k_1|l_1) \geqslant P(k_2|l_2) \geqslant \ldots \geqslant P(k_n|l_n).$$

The candidates of the first assignment can be represented by

$$\mathcal{C} = \{P(k_1|l_1)d_{i_1,j_1}, P(k_2|l_2)d_{i_2,j_2}, \ldots, P(k_c|l_c)d_{i_c,j_c}\}$$

where c is hyper-parameter controlling the degree of greedy for the GRASP algorithm. If $c = 1$, the GRASP algorithm always choice the optimal assignment in the first step. If c is greater than 1, such as $c = n/2$, the GRASP algorithm may not have the best performance in the first several steps but it is possible to escape from the local optima.

We first randomly select one assignment from $\mathcal{C}$. In the next step, we assign other symbols one by one. For any unassigned bit-sequence i and unused symbol k, we compute C_{ik} which is defined by

$$C_{ik} = \sum_{(j,l)} P(k|l)d_{i,j}$$

where (j, l) denote all the assigned pairs. Then, we select (i, k) from the several smallest C_{ik} randomly and assign the pair. This step should be repeated until all the bit-sequences are assigned. It is obvious that the choosing criteria in the construction phase follow a greedy algorithm with a certain randomness. On the local search phase, we exchange two symbols randomly and compute the cost function to see if we have an improvement. Algorithm 4 shows more details associated with implementing the GRASP algorithm.

4.4 BER computation for constellations of inconstant total energy and SA-based algorithm

In this section, we consider the more complex instances where the integral region for assigned symbols might be changed on the optimization process. For example, two

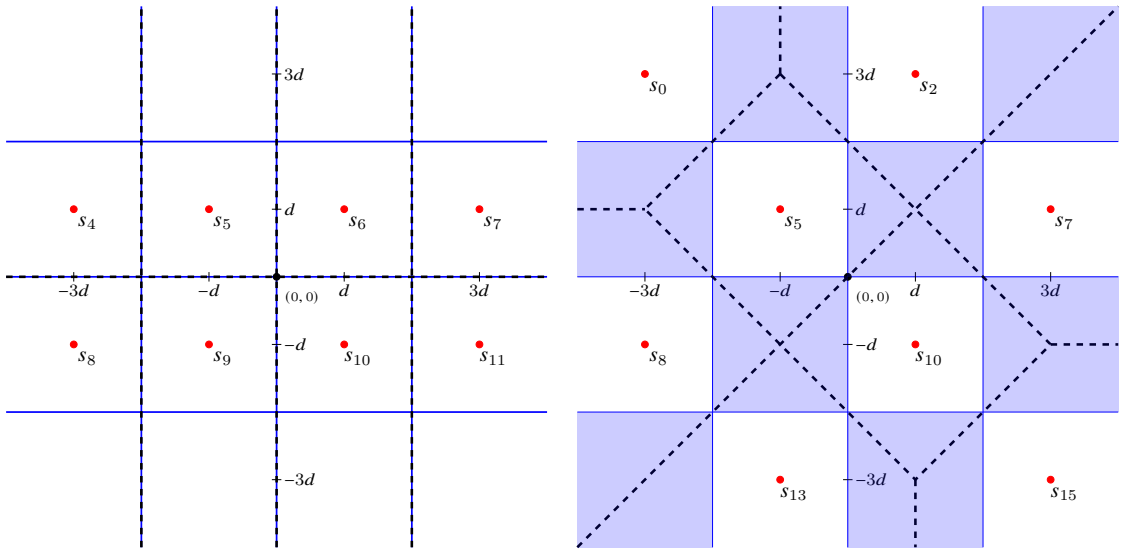
Algorithm 4 GRASP for constellation assignment

```
Sort Hamming distances in increasing order: d_{i_1,j_1} ⩽ d_{i_2,j_2} ⩽ ... ⩽ d_{i_n,j_n}
Sort SER in decreasing order: P(k_1|l_1) ⩾ P(k_2|l_2) ⩾ ... ⩾ P(k_n|l_n)
Select a pair P(k_0|l_0)d_{i_0,j_0} from {P(k_1|l_1)d_{i_1,j_1}, P(k_2|l_2)d_{i_2,j_2}, ..., P(k_n|l_n)d_{i_n,j_n}}
randomly
Assign bit-sequences i_0, j_0 to k_0, l_0 respectively and let Γ = {(i_0, k_0), (j_0, l_0)}
for iter = 3 : n do
    for i = 1 : n do
        for j = 1 : n do
            if (i, k) ∉ Γ then
                C_ik = Σ_{(j,l)∈Γ} P(k|l)d_{i,j}
            end if
        end for
    end for
    Select (i, k) from the s smallest C_ik randomly
    Update Γ = Γ ∪ {(i, k)}
end for
if BER of Γ is not locally optimal then
    Exchange a pair of assignments
end if
```

16-QAM constellations carrying 3 bits are shown in Figure 4.1. Integral regions are divided by black dashed line. In Figure 4.1a, the integral regions are regular and the value of SER can be computed by the $erfc$ function. In Figure 4.1b, the boundaries of the integral region of some symbols, i.e. s_2, are irregular such that SER cannot be directly computed by the $erfc$ function. Thus, before the computation of SER, the integral region of each symbol should be determined. For the irregular integral region, the computation of SER becomes difficult and time-consuming. Moreover, it is noted that, since the total energy of assigned symbols are different in Figure 4.1a and 4.1b, the variance in Equation (5.12) is changed so as to affect the computation of the SER matrix.

It is concluded that there are two major difficulties of the BER computation:

- The partition structure may differ from time to time, which increases the difficulty of the SER computation using the integral method.



(a) Assignment scheme 1 (b) Assignment scheme 2

Figure 4.1: Assignment schemes for 16-QAM constellation carrying 3 bits

- The total energy of symbols in constellation may vary from case to case, which aggravates the computation burden of SER.

In order to tackle the SER computation problem, we propose a new approach to approximate the SER which can be treated as a weighted sampling method based on integration results. In our novel approach, the whole constellation plane is divided into small square regions (or cells). For each cell, we compute the probabilities that the cell is categorized as each symbols. These probabilities can be derived from the integral method in which the $erfc$ function is applied. A new SER matrix, B_0 , can be formed as

$$B_0 = \begin{bmatrix} P(c_0|s_0) & P(c_1|s_0) & P(c_2|s_0) & \cdots & P(c_{N-1}|s_0) \\ P(c_0|s_1) & P(c_1|s_1) & P(c_2|s_1) & \cdots & P(c_{N-1}|s_1) \\ P(c_0|s_2) & P(c_1|s_2) & P(c_2|s_2) & \cdots & P(c_{N-1}|s_2) \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ P(c_0|s_{M-1}) & P(c_1|s_{M-1}) & P(c_2|s_{M-1}) & \cdots & P(c_{N-1}|s_{M-1}) \end{bmatrix}$$

which is a M-by-N matrix. In this matrix, $c_0, c_1, c_2, \ldots,$ and c_{N-1} represent the cells regions; $s_0, s_1, s_2, \ldots,$ and s_{M-1} are the candidate symbols.

The original SER matrix B can be constructed using the matrix B_0. Figure 4.2 shows an example of 16-QAM constellation carrying 3 bits. The constellation plane is divided by 256 cells. Eight bit-sequences are assigned to 8 of 16 symbols in the constellation. The assigned symbols can be denoted by the set $\mathcal{A} = \{s_0, s_2, s_5, s_7, s_8, s_{10}, s_{13}\}$. Firstly, we compute the 16×256 matrix B_0 using the integral method. Secondly, according to the set $\mathcal{A}$, Row 1,3,6,8,9,11,14 and 16 of B_0 are preserved to construct the matrix B. Thirdly, the matrix B is constructed by the combination of columns in matrix B_0. Note that, for each column in matrix B_0, the elements are the probabilities that one cell are assigned to different symbols. The biggest value suggests that the cell is closest to that symbol. For example, if $p(c_0|s_i)$ is the largest one in Column 1, c_0 should be categorized as s_i such that the value in Column 1 should be added to the $(i+1)$-th column in matrix B. As we can see, the third step is natural for parallel computing.

However, problems may occur in the third step. As shown in Figure 4.2, the blue cells cannot be assigned to a single symbol since they are equally divided by two symbols. The blue cells bring errors into the computation of the SER matrix. This computation error can be reduced by using smaller cells. For example, Figure 4.1b and Figure 4.2 shows the same constellation. The area of blue cells is reduced in Figure 4.2 because the size of cells is smaller. Since the constellation design belongs to the combinatorial optimization, the objective function is changed in a discrete way. When the number of cells is big enough, the objective function can be used to correctly discriminate different assignment solutions. Therefore, the computation error can be omitted.

As we mentioned before, both the total energy of the symbols and the threshold lines can be changed. Note that, if we only exchange 2 symbols among all the

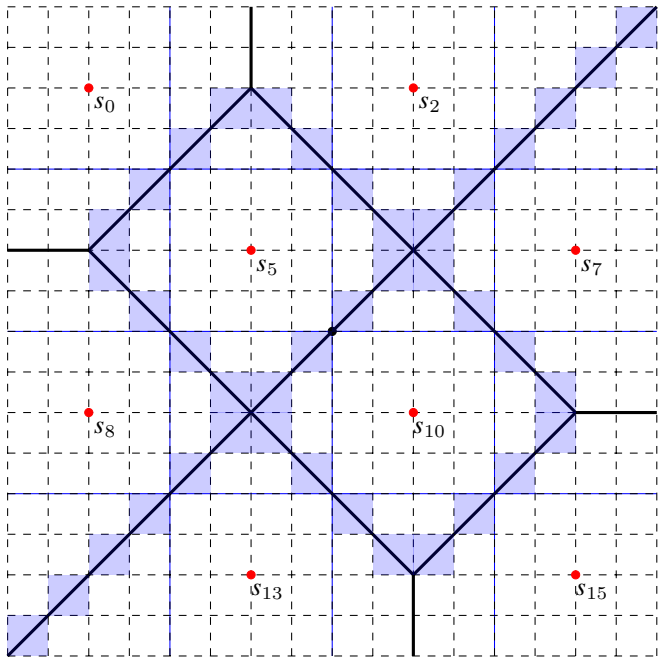


Figure 4.2: Grid blocks to approximate the SER matrix

assigned symbols, we only need to edit the SER matrix by changing the row and column corresponding to the swapped elements in local search. On the contrary, when we replace symbols of the current assignment with unassigned symbols, all elements of the SER matrix could be different such that we need to recalculate the matrix from the beginning. There are two different situations when replacing symbols with the unassigned ones. If the newly added symbol shares the same energy with the replaced one, the total energy level of the assignment stays the same. Otherwise, the total energy level changes. These two cases need to be treated in different ways. The comparison of these operations is listed in Table 4.1 where s_1, s_2, A represent the replaced symbol, the newly added symbol and the current assignment, respectively.

Operations	Related symbols	Total energy	Elements of B_0	Elements of B
O_1	$s_1 \in A, s_2 \in A$	unchanged	unchanged	unchanged
O_2	$s_1 \in A, s_2 \notin A, E(s_1) = E(s_2)$	unchanged	unchanged	changed
O_3	$s_1 \in A, s_2 \notin A, E(s_1) \neq E(s_2)$	changed	changed	changed

Table 4.1: Comparison of different operations

It is evident that the replacing operation consumes much more time than the exchange operation. Therefore, in order to speed up the searching process, we sperate the search into two layers. To be precise, we regard the exchange operation as a kind of local search and only employ the replacement operation after repeating the exchange operation for times.

In addition, SA algorithm is applied on the searching process. SA algorithm is a probabilistic technique for approximating the global optimum. It is one of the meta-heuristic which has been applied in different fields especially the cases with large search space. The main idea is shown in Algorithm 5.

Algorithm 5 Main steps of SA

```
input: terminal temperature T_min, initial temperature T_max, cooling factor T_factor,
    accept factor α, iteration number of every temperature L, expected lowest cost
    G_value, maximum of consecutive unaccepted times G_max, initial solution X and
    its corresponding cost C_X
t = T_max, g = 0
while t > T_min do
    for iteration = 1 : L do
        Generate a new solution X' and its corresponding cost C_X'
        if C_X' < C_X then
            q = 1
        else
            q = α × t × exp(−(C_X' − C_X)/t)
        end if
        if q > rand then
            X = X', C_X = C_X', g = 0
        else
            g = g + 1
        end if
        if C_X ⩽ G_value || g > G_max then
            break
        end if
    end for
    t = t × T_factor
end while
```

In order to escape from the local optima, meta-heuristic methods involves accept-

ing solution which is slightly worse. In our problem, from each current assignment, we select a solution following our proposed strategy which is random to some extent. If we find out a new assignment with a smaller value of BER, the SA algorithm accepts this new assignment with probability one. For a new assignment with worse performance, the acceptable probability is set to a function of the current temperature t. Note that the probability is not constant during the whole search procedure. As the search progresses, the temperature gradually decreases from the initial positive value to zero. As the temperature goes down, the probability to accept a worse assignment drops down. After a large number of iterations, the SA algorithm finds an optimal solution with probability one. On the other hand, B_0 only changes when the total energy level changes. For example, we only have at most 133 energy levels for the 64-QAM constellation carrying 4 bits case during the whole search procedure. In order to reduce the computational complexity, we save B_0 for different total energy levels and recall them to compute B.

Taking into account the above factors, we propose an tailor-made algorithm where the simulated annealing strategy has been embedded in the outer loop of local search. Figure 4.3 displays the flowchart of our proposed algorithm. The pseudo-code is demonstrated by Algorithm 6 in the Appendix.

4.5 Simulation

In this section, we use 2 examples to illustrate how the proposed framework works and verify the effectiveness of our algorithm at the same time.

4.5.1 Example 1: 16QAM-4bits via GRASP

In the first example, we consider the 16-QAM constellation where each symbol carries 4-bit information. The 16-QAM constellation is a ubiquitous constellation in digital communication industry [111] and its gray coding can be found in the open literature

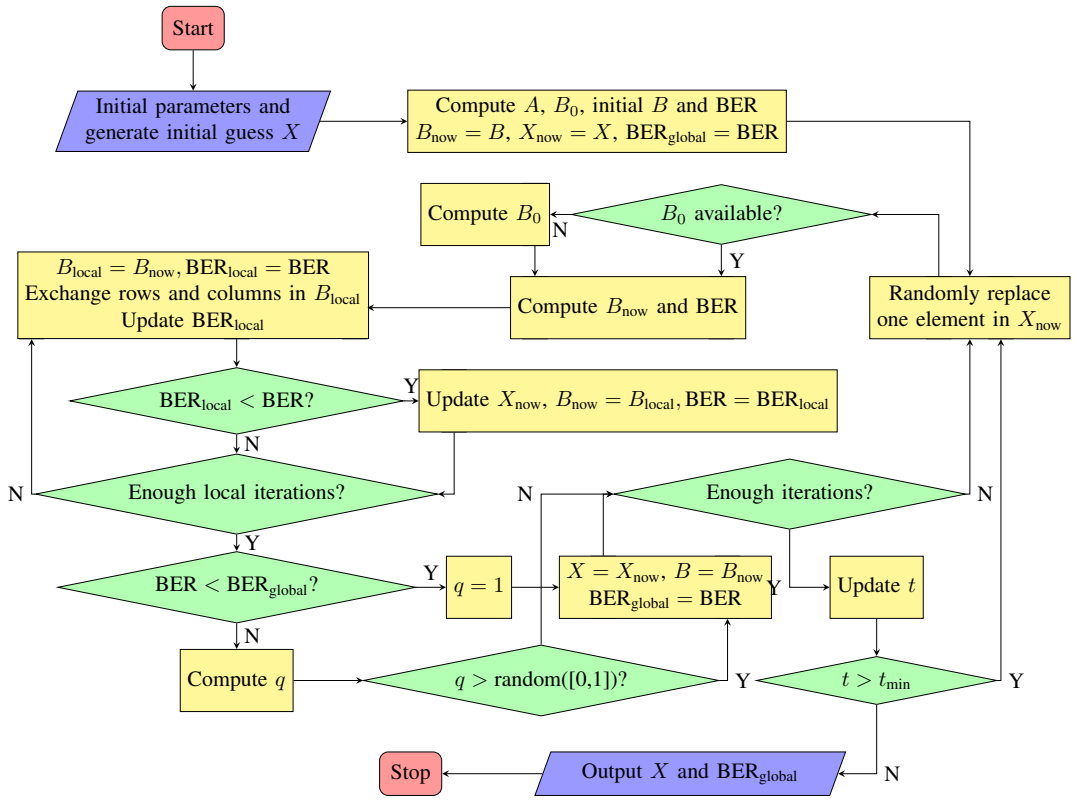


Figure 4.3: Flowchart in terms of the SA-based algorithm

[29]. The set of symbols in 16-QAM is $S = \{s_0, s_1, \ldots, s_{15}\}$. Figure 4.4a shows the coordinate of each symbol. Symbols are separated by the threshold line denoted by the black lines. The threshold lines divide the QAM plane into 16 areas according to the coordinates of symbols. When the received signal falls into one of these areas, it can be directly accepted as the symbol in the same area. Since the considered facilities is less than 20 and the computation process of BER has already given with details, we apply the GARSP algorithm (Algorithm 4) to solve this example.

Note that we use the approximation of BER as the cost function. There is no change of the integral domain such that the elements in SER matrix B is unchanged on the searching process. We are able to find different combinatorial results which are all the gray coding. Partial results are listed in Table 4.2. Figure 4.4b shows the

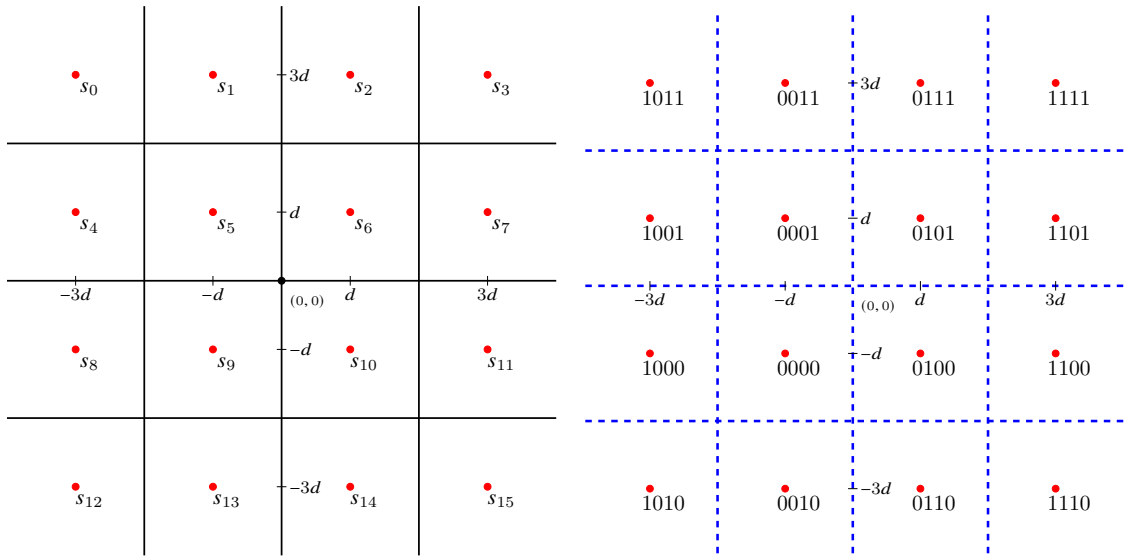
assigned symbols in terms of Result 1 in Table 4.2. It is noted that the Hamming distance in terms of any pair of neighbor symbols in Table 4.2 is equal to one such that the mapping rule in Result 1 leads to the gray coding of 16-QAM constellation. Same conclusion can be derived from Result 2 to Result 7.

	s_0	s_1	s_2	s_3	s_4	s_5	s_6	s_7	s_8	s_9	s_{10}	s_{11}	s_{12}	s_{13}	s_{14}	s_{15}
Result 1	1011	0011	0111	1111	1001	0001	0101	1101	1000	0000	0100	1100	1010	0010	0110	1110
Result 2	1001	1011	1111	1101	1000	1010	1110	1100	0000	0010	0110	0100	0001	0011	0111	0101
Result 3	1100	1101	0101	0100	1000	1001	0001	0000	1010	1011	0011	0010	1110	1111	0111	0110
Result 4	1110	0110	0111	1111	1100	0100	0101	1101	1000	0000	0001	1001	1010	0010	0011	1011
Result 5	1110	0110	0111	1111	1010	0010	0011	1011	1000	0000	0001	1001	1100	0100	0101	1101
Result 6	1110	1010	1011	1111	0110	0010	0011	0111	0100	0000	0001	0101	1100	1000	1001	1101
Result 7	1001	1000	0000	0001	1101	1100	0100	0101	1111	1110	0110	0111	1011	1010	0010	0011

Table 4.2: Searching results by the GRASP algorithm

4.5.2 Example 2: Honeycomb-3bits via SA

In the second example, we consider the honeycomb-structured constellation containing 31 symbols (HC31). The set of symbols in HC31 constellation is $S = \{s_0, s_1, \ldots, s_{30}\}$. Figure 4.6a shows the position of each symbol. Each symbol is separated by the threshold line denoted by the black lines. The threshold lines divide the constellation plane into 31 areas according to the coordinates of symbols. Compared to square constellations, the honeycomb-structured constellation has a lower average signal power due to the advantage of geometric structure [109]. In this example we try to assign 8 bit-sequences to honeycomb-structured constellation which contains 31 symbols. There are 23 symbols which are not assigned the bit-sequences. When the received signal falls into the region of these 23 symbols, the bit-sequence is mapped onto the closest symbol in the constatation. We have



(a) Symbols in the 16-QAM constellation (b) Assignment solution of Result 1 in Table 4.2

Figure 4.4: The 16-QAM constellation carrying 4 bits

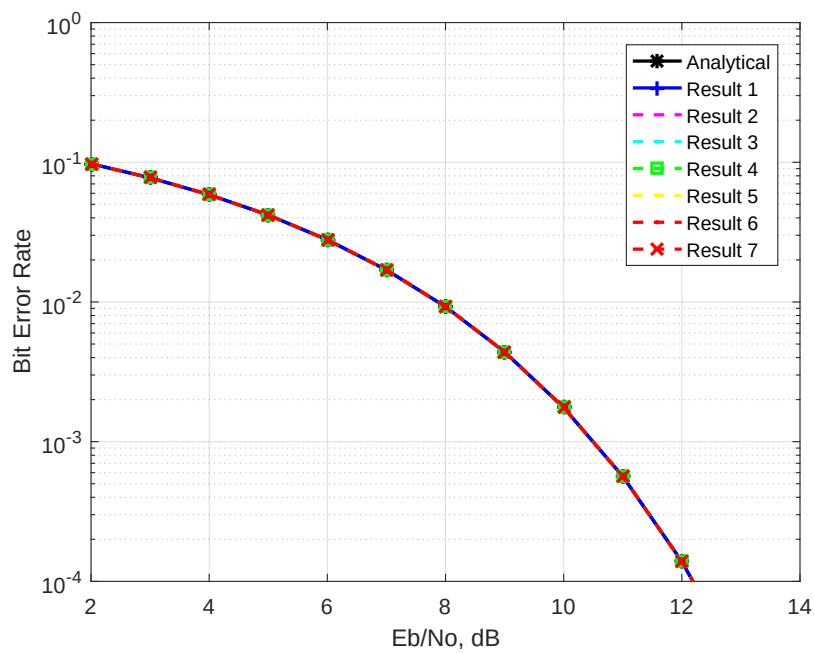


Figure 4.5: BER plot in terms of both analytical best assignment and results in Table 4.2

to verify the accuracy of our approximation calculation method of BER when the constellation does not follow the one-to-one mapping rule. We randomly generate 4 assignment solutions (trials) for the HC31 constellation. For Trial 1 and 2, bit-sequences '0000' to '1111' have randomly assigned to 16 symbols. For Trial 3 and 4, bit-sequences '000' to '111' have randomly assigned to 8 symbols. The assignment details are given in Table 4.3.

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Assignment 1 (Trial 1)	s_5	s_{25}	s_{28}	s_{27}	s_9	s_2	s_{26}	s_{22}	s_{24}	s_{20}	s_{18}	s_{23}	s_{10}	s_8	s_{11}	s_{30}
Assignment 1 (Trial 2)	s_{21}	s_{13}	s_{11}	s_{12}	s_{27}	s_{30}	s_{19}	s_{20}	s_9	s_1	s_3	s_{29}	s_6	s_{22}	s_{15}	s_{10}
Assignment 1 (Trial 3)	s_8	s_{24}	s_0	s_{25}	s_{20}	s_{29}	s_3	s_1	-	-	-	-	-	-	-	-
Assignment 1 (Trial 4)	s_{26}	s_{14}	s_9	s_{23}	s_7	s_{10}	s_8	s_{19}	-	-	-	-	-	-	-	-

Table 4.3: Four randomly assignment solutions (trials) for the HC31 constellation

Table 4.4 shows the comparison between the simulation BER and the BER computed by our approximation method. The total number of cells in the approximation method is denoted by q. It is noted that the approximation method behaves poorly if the number of cells is small,i.e., $q = 64$. With the increase of q, the approximation method becomes better and better. When q is equal to 65536, the maximum error rate for all the four trials is 0.25%. It is concluded that the approximation method is accurate enough for the optimization process when the number of cells is greater or equal to 4096. Apart from that, we can not observe the different pattern from 4 trials such that the different length of bit-sequences has no influence of the approximation method.

Since the number of candidate symbols in HC31 constellation is greater than 20, the optimization of assignments becomes an NP-hard problem. The simply heuristic algorithm, such as the GARSP algorithm, cannot effectively find out the global

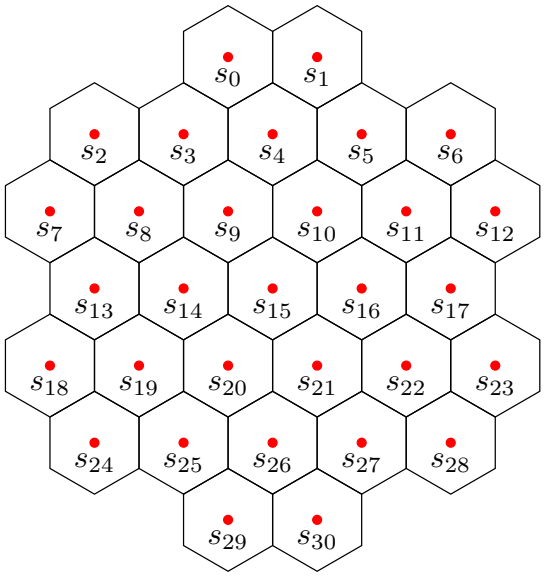
optima in this example. Besides that, the integral domain of each assigned symbol can be changed in the optimization procedure because 23 of 31 symbols are not assigned bit-sequences. In this complex example, the SA-based algorithm (Algorithm 6) is applied to derive the optimal assignment solution. The optimization is deployed when E_b/N_0 is equal to 1, 6 and 9. Figure 4.6b, 4.6c and 4.6d display the optimized HC31 constellation carrying 3 bits when E_b/N_0 is equal to 1, 6 and 9 respectively. It is observed that the Hamming distances between neighbor symbols are equal to 1 in Figure 4.6b. It can be concluded that the Hamming distance plays a more important role on the calculation of BER when E_b/N_0 is small because the noise power or the variance of the additive white noise distribution becomes larger when E_b/N_0 is smaller. Among three optimized constellations, it exists different assignment patterns when $E_b/N_0 = 1$ or when $E_b/N_0 = 6$ and 9. It is noted that the Hamming distances between some neighbor symbols is greater than 2 in Figure 4.6c and 4.6d but the symbols are separated uniformly. The main reason is that the uniformly separated constellation enlarge the distance between neighbor symbols. When E_b/N_0 is larger, the variance of the additive white noise distribution becomes smaller so as to enlarge the kurtosis. Thus, the value of SER between two symbols reduces rapidly if the distance between these two symbols becomes larger.

Figure 4.7 displays the BER plot in terms of 16-QAM constellation (Gray coding), 8-PSK constellation (Gray coding) and the optimized HC31 constellations. As we can see, all three optimized HC31 constellations have a significantly smaller BER value than two Gray coding cases when E_b/N_0 is greater than 3. To be more precise, Table 4.5 demonstrates the BER value in Figure 4.7. It is noted that only the optimized HC31 constellation with $E_b/N_0 = 1$ can beat 8-PSK constellation (Gray coding) when $0 \leqslant E_b/N_0 \leqslant 2$ but it has the worst performance among three optimized HC31 constellations when E_b/N_0 is large. It is because the deployment of the simulation environment. It is evident that the optimized HC31 constellations with $E_b/N_0 = 1$

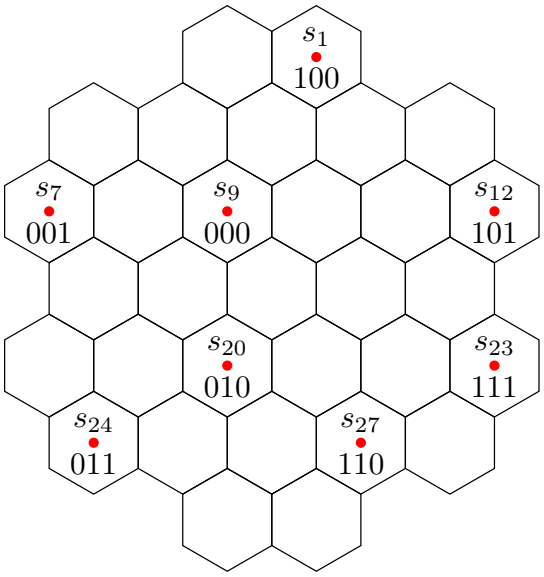
has an advantage when E_b/N_0 is merely close to 1.

	E_b/N_0	Simulation	q=64	Error	q=4096	Error	q=16384	Error	q=65536	Error
Trial 1	1	2.135E-01	2.295E-01	7.49%	2.138E-01	0.12%	2.136E-01	0.02%	2.135E-01	0.01%
	3	1.579E-01	1.974E-01	25.03%	1.583E-01	0.25%	1.580E-01	0.05%	1.579E-01	0.02%
	5	1.039E-01	1.658E-01	59.54%	1.045E-01	0.58%	1.041E-01	0.18%	1.040E-01	0.09%
	7	5.756E-02	1.426E-01	147.74%	5.830E-02	1.28%	5.779E-02	0.40%	5.767E-02	0.18%
	9	2.439E-02	1.384E-01	467.67%	2.506E-02	2.77%	2.455E-02	0.68%	2.443E-02	0.17%
Trial 2	1	2.474E-01	2.739E-01	10.69%	2.478E-01	0.13%	2.475E-01	0.03%	2.475E-01	0.03%
	3	1.786E-01	2.198E-01	23.09%	1.791E-01	0.29%	1.788E-01	0.08%	1.787E-01	0.04%
	5	1.130E-01	1.710E-01	51.33%	1.137E-01	0.58%	1.132E-01	0.13%	1.130E-01	0.01%
	7	5.930E-02	1.327E-01	123.79%	6.013E-02	1.39%	5.953E-02	0.38%	5.938E-02	0.13%
	9	2.334E-02	1.058E-01	353.46%	2.408E-02	3.17%	2.353E-02	0.83%	2.340E-02	0.25%
Trial 3	1	2.406E-01	2.499E-01	3.86%	2.407E-01	0.03%	2.407E-01	0.04%	2.406E-01	0.01%
	3	1.981E-01	2.201E-01	11.09%	1.983E-01	0.07%	1.982E-01	0.03%	1.981E-01	0.03%
	5	1.501E-01	1.803E-01	20.12%	1.505E-01	0.29%	1.503E-01	0.13%	1.502E-01	0.06%
	7	1.006E-01	1.480E-01	47.10%	1.012E-01	0.63%	1.008E-01	0.21%	1.007E-01	0.08%
	9	5.613E-02	1.207E-01	115.07%	5.696E-02	1.48%	5.639E-02	0.46%	5.624E-02	0.19%
Trial 4	1	2.195E-01	2.280E-01	3.86%	2.197E-01	0.09%	2.195E-01	0.00%	2.195E-01	0.01%
	3	1.621E-01	1.800E-01	11.07%	1.624E-01	0.17%	1.621E-01	0.03%	1.621E-01	0.03%
	5	1.077E-01	1.390E-01	29.02%	1.081E-01	0.34%	1.078E-01	0.07%	1.077E-01	0.02%
	7	6.081E-02	1.077E-01	77.10%	6.125E-02	0.74%	6.089E-02	0.13%	6.080E-02	0.02%
	9	2.655E-02	8.736E-02	229.00%	2.707E-02	1.94%	2.668E-02	0.47%	2.657E-02	0.07%

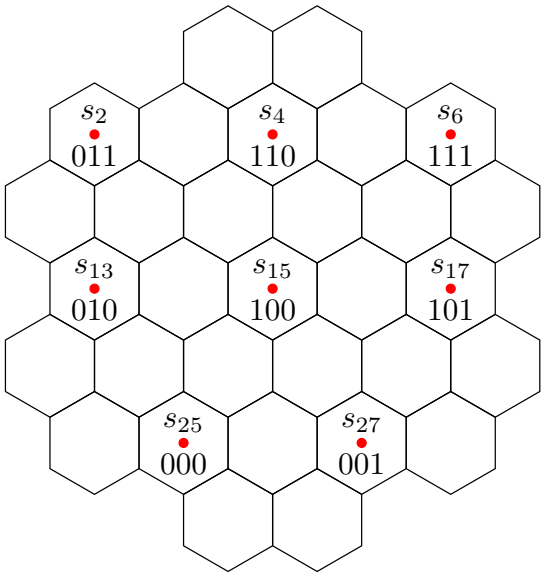
Table 4.4: BER approximation calculation in terms of 4 trials in HC31 constellation



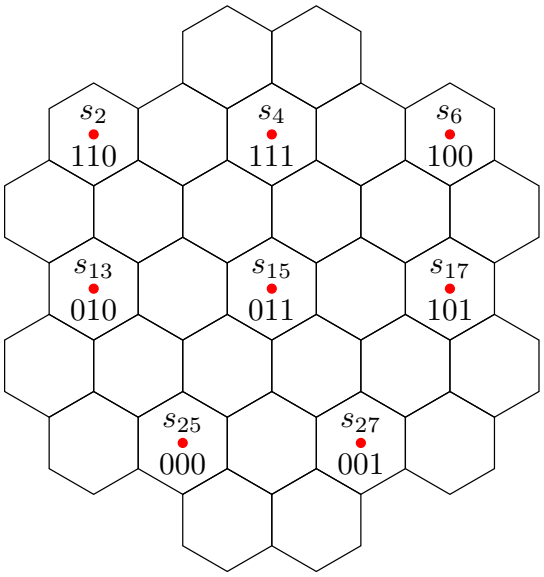
(a) Symbols in the HC31 constellation [109]



(b) Optimized HC31 with $E_b/N_0 = 1$



(c) Optimized HC31 with $E_b/N_0 = 6$



(d) Optimized HC31 with $E_b/N_0 = 9$

Figure 4.6: HC31 constellation carrying 3 bits

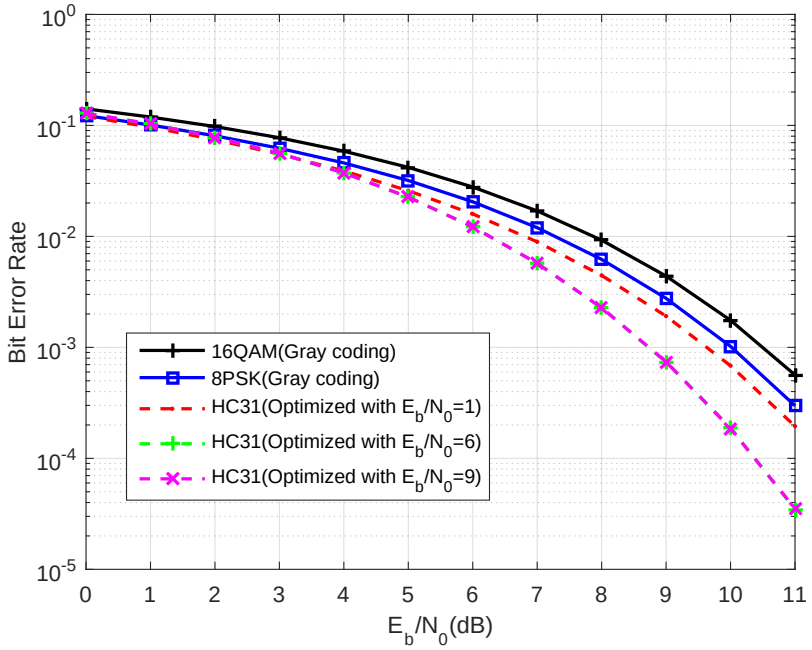


Figure 4.7: BER plot of optimized HC31 constellations

E_b/N_0	16-QAM$_{\text{Gray coding}}$	8-PSK$_{\text{Gray coding}}$	HC31$_{E_b/N_0\,=\,1}$	HC31$_{E_b/N_0\,=\,6}$	HC31$_{E_b/N_0\,=\,9}$
0	1.4098E-01	1.2271E-01	1.2065E-01	1.2898E-01	1.2890E-01
1	1.1900E-01	1.0082E-01	9.6373E-02	1.0295E-01	1.0293E-01
2	9.7742E-02	8.0636E-02	7.4414E-02	7.8329E-02	7.8306E-02
3	7.7453E-02	6.2276E-02	5.5130E-02	5.6079E-02	5.6084E-02
4	5.8624E-02	4.5928E-02	3.8890E-02	3.7277E-02	3.7301E-02
5	4.1893E-02	3.1897E-02	2.5811E-02	2.2601E-02	2.2563E-02
6	2.7871E-02	2.0516E-02	1.5934E-02	1.2184E-02	1.2193E-02
7	1.6967E-02	1.1984E-02	8.9383E-03	5.7157E-03	5.7264E-03
8	9.2472E-03	6.2058E-03	4.4499E-03	2.2660E-03	2.2753E-03
9	4.3903E-03	2.7652E-03	1.9136E-03	7.3918E-04	7.4109E-04
10	1.7542E-03	1.0212E-03	6.8376E-04	1.8881E-04	1.8568E-04
11	5.6471E-04	2.9815E-04	1.9500E-04	3.4700E-05	3.5300E-05

Table 4.5: The values of BER in Figure 4.7

4.6 Final Remark

In this chapter, we propose a novel method to optimize the constellation in digital communication system. Using the SER matrix and Hamming distance matrix, the design of the constellation is formulated as a QAP in which the value of BER is minimzed. We apply the GARSP algorithm to solve the low-dimensional optimization problem on the design of constellations. For high-dimensional constellations, we propose a tailor-made method based on SA algorithm. Simulation results show that our propose methods can truly derive the optimal assignment solution for different constellations.

4.7 Appendix

Algorithm 6 SA for global search

1: **input** Parameters: number of candidate symbols m, number of the bit-sequences n, number of small squares q, E_b/N_0, bits per symbol k, number of neighbors $\mathcal{N}_{\text{neighbor}}$, terminal temperature $T_{\min}$, initial temperature $T_{\max}$, cooling factor T_{factor}, accept factor α, iteration of every temperature L, expected lowest cost G_{value}, maximum of consecutive unaccepted times $G_{\max}$, iteration of local search $\text{Iter}_{\text{local}}$

2: Compute the SER matrix of constellation with m symbols carrying n bit-sequences and find the nearest symbols whose number is $\mathcal{N}_{\text{neighbor}}$ to each symbol to construct matrix Neighbors

3: Generate the initial assignment X: X =randperm(m, n)

4: Compute the total energy E_{total} of X and compute the corresponding d

5: Compute the Hamming distance matrix A of bit-sequences $0 : n-1$

6: Generate SER matrix of the squares B_0 with $E_b/N_0, k, d, m, q$

7: Set $B_0^{\text{total}} = B_0$, $\text{Level}_{\text{global}} = 1$ and $\text{Engergy}_{\text{store}} = [1; E_{\text{total}}]$ where $\text{Level}_{\text{global}}$ is the number of energy level in the whole optimization process

8: Compute SER matrix B with B_0 and X

Compute BER : BER $= \sum\sum(A \circ B)/(n \times k)$

Set $d_{\text{total}} = d_{\text{now}} = d$; $\text{BER}_{\text{best}} = \text{BER}_{\text{global}} = \text{BER}$; $\text{Level}_{\text{local}} = 1$

$t = T_{\max}, g = 0$

while $t > T_{\min}$ **do**

  for iteration $= 1 : L$ **do**

    $X_{\text{now}} = X$; $B_{\text{now}} = B$

    Randomly select one assigned symbol: s_1

    Randomly select one neighbor of s_1 : s_2

    if find($s_2 == X_{\text{now}}$) **then**

      Update X_{now} via the position exchange between s_1 and s_2

      Update B_{now} via the exchange of both the columns and rows related to s_1 and s_2

    else

      Update X_{now} by replacing s_1 with s_2

      Compute the new corresponding d and let $d_{\text{now}} = d$

      if find($d_{\text{now}} == d_{\text{total}}$) **then**

        $\text{Level}_{\text{local}} = \text{find}(d_{\text{now}} == d_{\text{total}})$; $B_0 = B_0^{\text{total}}(:, :, \text{Level}_{\text{local}})$

      else

        Compute the new total energy $E_{\text{total}}^{\text{new}}$ and update B_0 with d_{new} and $E_{\text{total}}^{\text{new}}$

        Update $\text{Level}_{\text{global}} = \text{Level}_{\text{global}} + 1$; $B_0^{\text{total}}(:, :, \text{Level}_{\text{global}}) = B_0$ and $d_{\text{total}} = [d_{\text{total}}; d_{\text{new}}]$

        Update $\text{Energy}_{\text{store}} = [\text{Level}_{\text{global}}; E_{\text{total}}^{\text{new}}]$; $\text{Level}_{\text{local}} = \text{Level}_{\text{global}}$ and $\text{BER}_{\text{best}} = [\text{BER}_{\text{best}}; \text{Inf}]$

      end if

      Compute B_{now} with X_{now} and B_0

    end if

    $\text{BER} = A \circ B_{\text{now}}/n/k$

    for $\text{Iter}_{\text{ls}} = 1 : \text{Iter}_{\text{local}}$ **do**

      $B_{\text{local}} = B_{\text{now}}$

Randomly select two assigned symbols s_{local1} and s_{local2}
Update B_{local} via the exchange of columns and rows related to s_{local1} and s_{local2}
Update $\text{BER}_{\text{local}}$ by $\text{BER}_{\text{local}} = \sum\sum(A \circ B_{\text{local}})/(n \times k)$
if $\text{BER} > \text{BER}_{\text{local}}$ **then**
$\text{BER} = \text{BER}_{\text{local}}$; $B_{\text{now}} = B_{\text{local}}$
Update X_{now} via the position exchange between s_{local1} and s_{local2}
end if
if $\text{BER} < \text{BER}_{\text{best}}(\text{Level}_{\text{local}})$ **then**
$\text{BER}_{\text{best}}(\text{Level}_{\text{local}}) = \text{BER}$; $X_{\text{best}}(\text{Level}_{\text{local}}\ ,:) = X_{\text{now}}$
end if
end for
if $\text{BER} < \text{BER}_{\text{global}}$ **then** $q = 1$
else
$q = \alpha \times t \times \exp(-(\text{BER} - \text{BER}_{\text{global}})/t)$
end if
if $q > \text{rand}$ **then** $\text{BER}_{\text{global}} = \text{BER}$; $X = X_{\text{now}}$; $B = B_{\text{now}}$; $g = 0$
else
$g = g + 1$. Compute the new corresponding d and let $d_{\text{now}} = d$. $\text{Level}_{\text{local}} = \text{find}(d_{\text{now}} == d_{\text{total}})$
end if
end for
$t = t \times T_{\text{factor}}$
if $\text{BER}_{\text{global}} \leqslant G_{\text{value}} | g > G_{\text{max}}$ **then** break
end if
end while
output $\text{Energy}_{\text{store}}$; BER_{best}

Chapter 5

Coordinates optimization in constellations

Signal constellation plays an important role in signal processing and digital modulation system. Good constellation design can reduce information errors during signal transmission. A common discriminant index is the bit error rate (BER) which represents the number of bit errors per unit time. In this chapter, we focus on coordinates optimization for symbols in constellations. The constellation design problem is formulated as a dynamic optimization problem where the symbol coordinates in the constellation are the decision variables. The objective function is to minimize the BER value in the range of signal-to-noise ratios (SNRs). The optimization process contains two phases: construction phase and optimization phase. In the construction phase, a rough optimization scheme is proposed. In the optimization phase, a novel approximation calculation method of BER is proposed to speed up the optimization process. Simulation results show that our proposed methods can truly improve the performance of constellations.

5.1 Research background

In communication systems, there are two common kinds of signals including the digital signal and analog signal. Digital signals are represented by the sequence of

voltage pulses, which are normally used within the circuitry of the computer system. In other words, the stored data in the computer is represented by a discrete bit-sequence, e.g., '00100010' is a bit-sequence containing 8 bits information, where '1' is represented by high voltage and '0' is denoted by low voltage. When it comes to the transmission process via physical medium, bit-sequences are normally transmitted by the analog signals. The analog signals are continuous waveform in nature. For example, conventional signal cables carry electromagnetic waves and fiber optic systems transmit light waves. In practice, raw analog signals emitted by an information source, i.e., baseband signals, always have properties such as low frequency, wide frequency band and overlap [66]. All these characteristics make baseband signals unsuitable for wireless transmission so as to limit the communication capacity. Besides that, baseband signals have a poor ability of anti-interference. Thus, it is merely used in the short-distance local area network (LAN) transmission. In order to enhance the anti-interference ability as well as reduce the transmission error, it is helpful to apply the modulation technique [3] where digital signals are transformed to the periodic electromagnetic waveform. According to different configurations on the amplitude, frequency and phase, one or more periodic waveforms can represent different digital signals. Various modulation modes have been proposed, such as quadrature amplitude modulation (QAM) [54], amplitude shift keying (ASK) [7] and phase-shift keying (PSK) [108]. In the QAM constellation, modulated signals can be written as follows,

$$s(t) = x(t)\cos 2\pi f_c t - y(t)\sin 2\pi f_c t$$

where f_c is the carrier frequency. It means all the modulated signals could be represented by a linear combination of orthogonal basis. The modulated signal $s(t)$ is one-to-one correspondence with the transmitted bit-sequence. For example, if we want to transmit 3 bits each time, all the possible sequences are '000', '001', '010',

'011', '100', '101', '110' and '111'. After the modulation phase, we will obtain 8 different $s(t)$ which could be represented by $s_0, s_1, \ldots, s_7$ with different values of x_t and y_t. In other words, the set $\{x(t), y(t)\}$ could represent the modulated signals as well as the encoded bit-sequences. Thus, encoded bit-sequences can be presented by a scatter diagram if we draw points (symbols) according to x_t and y_t. The scatter diagram is called a constellation diagram (two typical constellations are shown in Figure 5.1 and Figure 5.2) [90]. Note that since the quantity of symbols is determined by the amount of the bit-sequences, the number of symbols is usually a power of 2.

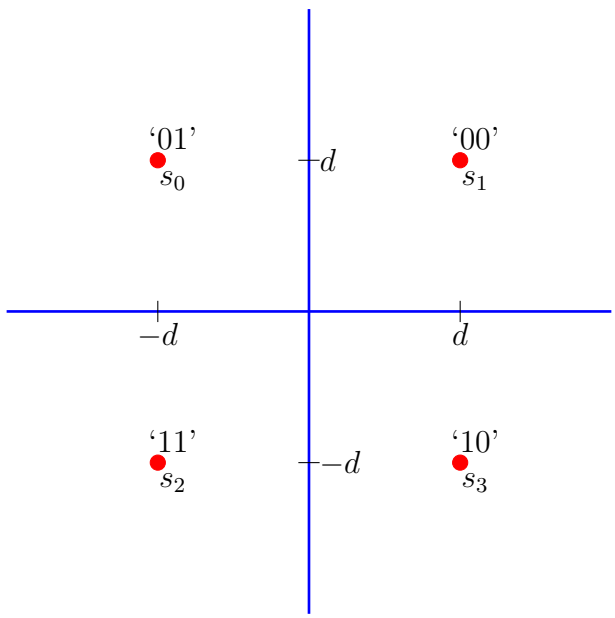


Figure 5.1: Constellation of 4-QAM: transmit 2 bits data each time; each symbol share the same amplitude while the phase shifts of the carrier sine wave are $0, \frac{\pi}{4}, \frac{\pi}{2}, \frac{3\pi}{2}$.

A constellation intuitively shows the mapping relationship between the bit-sequence and the modulated signal. Moreover, the angle of (x_t, y_t), measured counterclockwise from the x-axis represents the phase shift of the carrier wave and $||x(t), y(t)||$ is the measurement of the signal power.

On the contrary, demodulation refers to the process where the receiver recovers

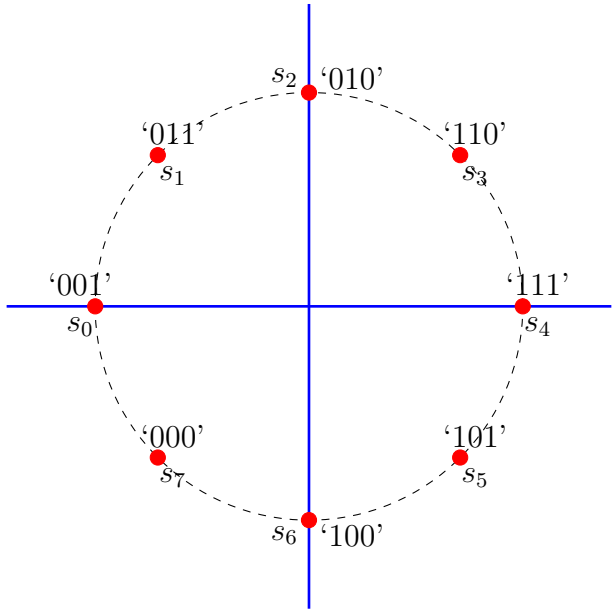


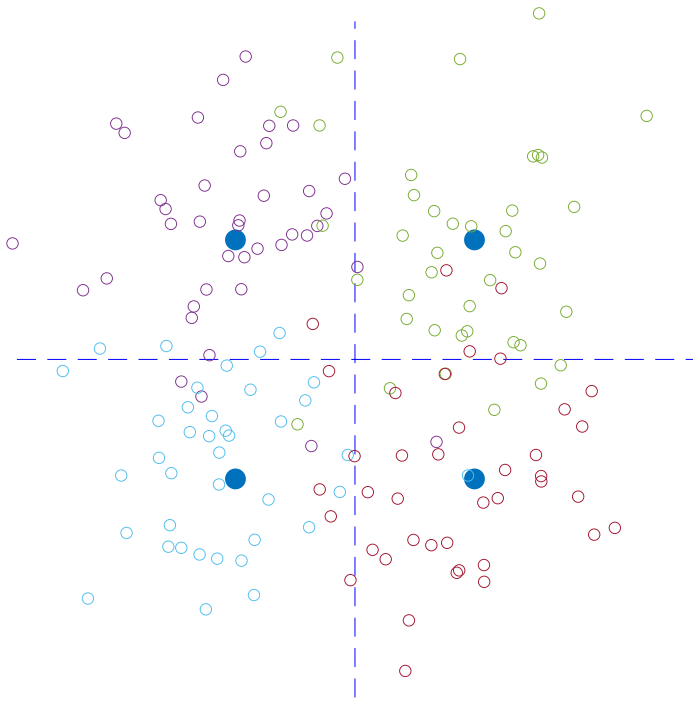
Figure 5.2: Constellation of 8-PSK: transmit 3 bits data each time; each symbol share the same amplitude while the phase shifts of the carrier sine wave are $0 : \frac{\pi}{4} : \frac{7\pi}{4}$.

the bit-sequence carried by the modulated signal. However, the constellation diagram is an ideal situation where the position of the modulated signal is precisely defined. In practice, due to the noise influence on signal transmission (i.e., attenuation, delay distortion and so forth), the received signal may not be consistent with the original modulated signal. As shown in Figure 5.3a and Figure 5.3b, 4 signals are transmitted in the ubiquitous presence of white noise. The received signal follows a bivariate Gaussian distribution where the mean is the symbol coordinates in terms of the transmitted signal. The positions of these received signals are affected by the noise power which can be measured by the signal-to-noise ratio (SNR). The smaller SNR leads to the more dispersed received signals [18].

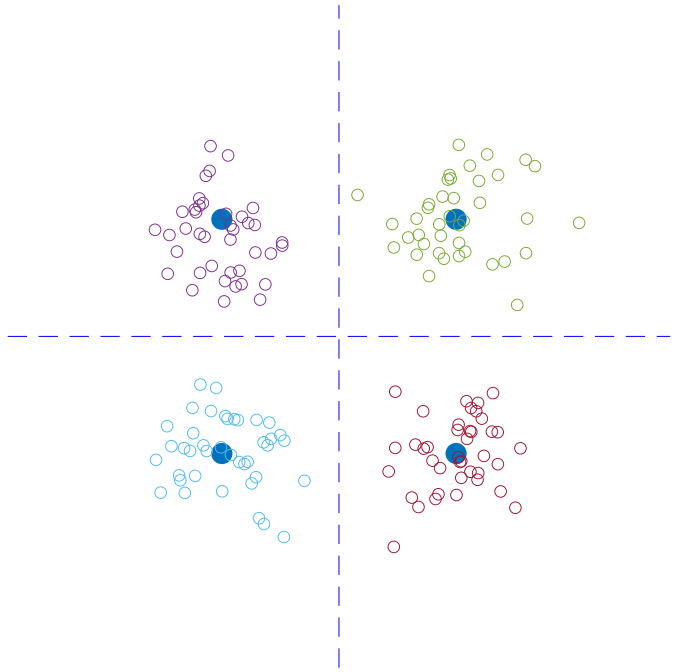
For the receiver, the discrimination of bit-sequences is realized by allocating received signals to different symbols because one symbol only represents a unique bit-sequence. The received signal is recognized as one symbol if the Euclidean distance between the received signal and this symbol is shortest. Due to the departure

from the received signal and the original symbol, it exists the transmission error especially in the presence of the large noise power. The bit-error-rate (BER) is applied to measure the degree of the transmission error, which denote the proportion of the number of error bits among the total number of transmitted bits. For example, if the transmitted bit-sequence is '00001001' but the received sequence is '00101000', there are two error bits among 8 transmitted bits. In this example, the BER value is $2/8 = 0.25$. It is noted that people focus on the expected value of BER for a designed constellation diagram rather than a BER value in terms of a short bit-sequence. Thus, the BER value is often calculated when different bit-sequences are repeatedly transmitted.

In the desired case, the BER value is expected to be as small as possible at the same level of the noise power. As mentioned in the previous sections, error bits can be caused by the incorrect allocation of the received signals. This incorrect allocation can be measured by the symbol-error-rate (SER) which can be represented by the probability of the received signal far from the original symbol because the received signals follow a Gaussian distribution. Besides that, the Hamming distances between symbols play an important role on the calculation of BER. To be more specific, the Hamming distance measures the difference between two bit-sequences with the same length. When one received signal is wrongly allocated, the number of error bits is equal to the corresponding Hamming distance. It is known that the bit-sequences with small Hamming distance should be assigned in relatively close positions while the positions of symbols should be distributed as widely as possible. For the fixed-length bit-sequences, the Hamming distance between any pair of sequences is also fixed. Moreover, The change of mapping relationship between bit-sequence and symbol can be realized by the change of symbol position in the constellation diagram. At the fixed SNR and mapping rule, BER can be solely determined by the coordinates of symbols. The optimization of symbol coordinates in the constellation is not trivial



(a) Scatter plot in terms of the received signals with small SNR



(b) Scatter plot in terms of the received signals with large SNR

Figure 5.3: Scatter plots in terms of the received signals

because the SER value of one specified symbol is varied with the change of the position of other symbols. We formulate the whole problem as a dynamic optimization problem where the coordinates in the constellation are the decision variables and the BER value is the objective function to be minimized [103], i.e.,

$$\min_{(x_i,y_i),\,1\leqslant i\leqslant N} \quad J = \mathrm{BER}((x_1,y_1),(x_2,y_2),\ldots,(x_N,y_N)), \quad (x_i,y_i)\in C, \tag{5.1}$$

where N is the number of the bit-sequences and C is the set of the candidate symbols. If the restriction of total symbol power are not imposed, $C = \mathbb{R}^2$. There are two major concerns about solving this problem:

- The optimization method for this problem;
- The accuracy and efficiency of the calculation method for the objective function.

For the first concern, we propose a two-phase optimization scheme including the rough optimization phase and the main optimization phase. Since the performance of optimization methods for the non-convex problem is heavily related to the initial guess, rough optimization aims to construct a good initial assignment solution. In the main optimization phase, the gradient-based optimization method is employed. For the second concern, we propose a novel approximation method to compute the SER matrix which can shorten the computation time and ensure the high computation accuracy. Numerical simulations verify that the proposed framework is effective and accurate.

The rest of this chapter is organized as follows. Section 5.2 shows the formulation and analysis of the optimization problem. Section 5.3 shows the proposed framework to compute the SER matrix. In Section 5.4, numerical simulations are given to verify both of the approximation computation of BER and the gradient-based optimization procedure. Final marks are presented in Section 5.5.

5.2 Problem formulation

The ultimate goal of constellation design is to achieve a better transmission performance, i.e., a lower BER value. In the previous section, we demonstrate that SER and Hamming distance are two determining factors of BER. In this section, we first illustrate the decoding (or demodulation) mechanism. Then, we introduce the optimization method in terms of symbol coordinates.

5.2.1 Hamming distance matrix and SER matrix

In the Additive White Gaussian Noise (AWGN) channel, signals are transmitted in the presence of Gaussian noise. The received signal is received in the form of

$$r(t) = s(t) + n(t),$$

where $n(t)$ is the additive white Gaussian noise with mean zero and variance $N_0/2$, and $n(t)$ is independent of $s(t)$. Given the signal $s(t)$, the received signal $r(t)$ follows a conditional Gaussian distribution. Mathematically, it is denoted by

$$r(t)|s(t) \backsim G(s(t), N_0/2) \tag{5.2}$$

where $G(s(t), N_0/2)$ denotes the Gaussian distribution with mean $s(t)$ and variance $N_0/2$. Without loss of generalization or optimality, this continuous-time AWGN channel can be replaced by an equivalent discrete-time channel model in which the received signal is denoted as

$$r = s + n, \tag{5.3}$$

where s is the random input signal sequence and n is an i.i.d. Gaussian noise sequence. The demodulation is achieved by measuring the Euclidean distance between the received signal and symbols in the constellation diagram. The received signal is categorized as the symbol s_i which attains the smallest Euclidean distance. To be

more clear, r is accepted as the symbol s_i if it satisfies

$$\arg\min_{s_i \in S} \|r - s_i\|_2, \tag{5.4}$$

where S is the set of all the symbols in the constellation. It is straightforward to define a threshold line, based on Set (5.4), to discriminate the region in which the received signal belongs to symbol s_i.

Assume that s_j is the transmitted signal but the received signal is s_i. SER represents the possibility that the signal s_j is wrongly received as s_i. Since the received signal can be represented as a conditional Gaussian distribution in (5.2), SER can be measured by a cumulative probability which is denoted by

$$P(s_i|s_j) = \iint\limits_{(x,y)\in D_{s_i}} f\Big((x,y)|(x(s_j), y(s_j))\Big)\mathrm{d}x\mathrm{d}y \tag{5.5}$$

where $(x(s_j), y(s_j))$ is the coordinates of symbol s_j; D_{s_i} denote the received region for symbol s_i; and $f\Big((x,y)|(x(s_j), y(s_j))\Big)$ is the probability density function of the bivariate Gaussian distribution with mean vector $(x(s_{i_0}), y(s_{i_0}))^\top$ and variance matrix $\Sigma = \mathrm{diag}(N_0/2, N_0/2)$.

The Hamming distance measures the difference between two bit-sequences with equal length. Mathematically, a Hamming distance d_{B_1,B_2} between bit-sequences B_1 and B_2 is defined by

$$d_{B_1,B_2} = \sum_{i=1}^{M} |B_1(i) - B_2(i)|, \tag{5.6}$$

where $B_1(i)$ and $B_2(i)$ are the ith binary variable in sequence B_1 and B_2, respectively; M is the total length of sequence B_1 or B_2. For example, given $B_1 =' 1010'$ and $B_2 =' 0111'$, d_{B_1,B_2} is equal to 3 based on Equation (5.6). If we assign bit-sequences B_1 and B_2 to s_1 and s_2 respectively, the Hamming distance between s_1 and s_2, d_{s_1,s_2}, is equal to d_{B_1,B_2}.

By doing so, a local BER between these two signals can be calculated by

$$\mathrm{BER}(s_2|s_1) = \frac{1}{M} P(s_2|s_1) d_{s_1,s_2},$$

where M is the length of each bit-sequence. All the local BER values can be computed similarly. The BER in terms of the transmission of symbol s_1 is $\sum_{i=0}^{N-1} \mathrm{BER}(s_i|s_1)$ where N is the number of symbols. The total BER could be given by

$$\mathrm{BER} = \frac{1}{NM} \sum_{j=0}^{N-1} \sum_{i=0}^{N-1} \mathrm{BER}(s_i|s_j) = \frac{1}{NM} \sum_{j=0}^{N-1} \sum_{i=0}^{N-1} P(s_i|s_j) d_{s_i,s_j}. \tag{5.7}$$

For convenience, we denote the Hamming distance matrix as A and the SER matrix as B, i.e.,

$$A = \begin{bmatrix} d_{s_0,s_0} & d_{s_1,s_0} & \cdots & d_{s_{N-1},s_0} \\ d_{s_0,s_1} & d_{s_1,s_1} & \cdots & d_{s_{N-1},s_1} \\ d_{s_0,s_2} & d_{s_1,s_2} & \cdots & d_{s_{N-1},s_2} \\ \vdots & \vdots & \ddots & \vdots \\ d_{s_0,s_{N-1}} & d_{s_1,s_{N-1}} & \cdots & d_{s_{N-1},s_{N-1}} \end{bmatrix}, \tag{5.8}$$

$$B = \begin{bmatrix} P(s_0|s_0) & P(s_1|s_0) & \cdots & P(s_{N-1}|s_0) \\ P(s_0|s_1) & P(s_1|s_1) & \cdots & P(s_{N-1}|s_1) \\ P(s_0|s_2) & P(s_1|s_2) & \cdots & P(s_{N-1}|s_2) \\ \vdots & \vdots & \ddots & \vdots \\ P(s_0|s_{N-1}) & P(s_1|s_{N-1}) & \cdots & P(s_{N-1}|s_{N-1}) \end{bmatrix} \tag{5.9}$$

The calculation of BER can be denoted by

$$\mathrm{BER} = \frac{1}{NM} \langle B, A \rangle$$

where $\langle \cdot, \cdot \rangle$ denotes the inner product of two matrices.

5.2.2 Gradient-based optimization

It is noted that the elements of Matrix A can be assumed to be constants. The BER value is only affected by the symbol coordinates in the constellation. We rewrite the optimization problem (5.1) as

$$\min_{(x_i,y_i),\,1\leqslant i\leqslant N} \quad J = \langle B((x_1,y_1),(x_2,y_2),\ldots,(x_N,y_N)), A\rangle, \quad (x_i,y_i)\in\mathbb{R}^2. \tag{5.10}$$

Since the symbol coordinates can be changed in the whole constellation, the problem becomes a continuous optimization problem where the coordinates are the decision variables. To simplify the expression,let θ denote the set of decision variables in the rest of this chapter, i.e.,

$$\theta = \{(x_1,y_1),(x_2,y_2),\ldots,(x_N,y_N)\}.$$

In the continuous optimization problem, the first step is to find the stationary point where the first-order gradient is equal to zero, i.e., $\nabla_\theta J = 0$. After that, we have to evaluate the second-order gradient, $\nabla_\theta^2 J$, at the stationary point. If $\nabla_\theta^2 J > 0$ holds, the stationary point is recognized as a local minimum of the objective function. Since it is not easy to derive the analytical solution of stationary points in Problem 5.10, we adopt an iterative approach to solve it. The strategy is that we search for a better solution from the initial guess along the *descent direction*. One direction s is recognized as the descent direction if s satisfies

$$\nabla J(\theta)^\top s < 0.$$

It is because ,for sufficiently small α, we can obtain the following equation from Taylor expansion

$$J(\theta+\alpha s) = J(\theta) + \alpha\nabla J(\theta+\eta\alpha s)^\top s < J(\theta)$$

which suggests that J will be smaller along direction s. The gradient-based algorithm is summarized as Algorithm 7.

Algorithm 7 Gradient-based optimization

Step 1. Choose an initial guess for θ.
Step 2. Calculate $B(\theta)$
Step 3. Compute the values of $J(\theta)$ and $\nabla_\theta J(\theta)$.
Step 4. Use the gradient information to calculate a search direction, and then update θ.
Step 5. If θ is optimal, then stop. Otherwise, return to Step 2.

There are three major concerns when Algorithm 7 is employed.

- The computation of the SER matrix, B, is time-consuming so as to slow down the whole optimization process.
- It is impossible to derive the theoretical expression of the gradient $\nabla_\theta J(\theta)$.
- It is better to derive a good initial guess for θ.

For the first concern, we propose a novel method to approximately compute the SER matrix. The details of our proposed method is illustrated in Section 3. For the second concern, we can only use the numerical differentiation methods to compute the gradients at each optimization step. Some softwares have the toolbox in which the gradients can be computed automatically following the numerical differentiation methods. In addition, several classical optimization methods can be applied to this problem such as BFGS method and SQP method [82]. For the third concern, we propose a rough optimization method.

5.2.3 Rough optimization

In order to kick off the optimization process in Problem (5.10), one can randomly generate a set of coordinates as the initial assignment solution. Alternatively, experienced one can manually construct an initial assignment solution according to the criteria that bit-sequences with small Hamming distance should be assigned to close symbols. However, the manual construction work becomes impossible if the number of symbols is large.

We propose a strategy to construct a good initial assignment where no preliminary knowledge or experience is required. The value of the probability density function is used to replace the value of the cumulative probability function in Problem (5.10). For two small identical areas of the constellation, given a specified Gaussian distribution, the value of probability density function is bigger when the center of this area is close to the mean of the Gaussian distribution. In other words, if the probability at the symbol point becomes larger, the probabilities at the points within its nearest region become larger too. Therefore, the precise probability in matrix B is replaced by the probability density value. To be more specific, the element (i, j) in matrix B is replaced by

$$\frac{1}{2\pi\sigma^2}\exp\left(-\frac{1}{2}\left(\frac{(x_i - x_j)^2}{\sigma^2} + \frac{(y_i - y_j)^2}{\sigma^2}\right)\right) \tag{5.11}$$

where (x_i, y_i) denote the coordinates of symbols.

5.3 Computation of the SER matrix

In this section, we discuss the approximate computation of the SER matrix. According to the definition of SER in Equation (5.5), the computation of SER is essentially a double integral. Among them, the form of integrand is easy to be determined because the received signal follows the gaussian distribution, i.e.,

$$p\big((x, y)|(x(s_{n_0}), y(s_{n_0}))\big) = \frac{1}{2\pi\sigma^2}\exp\left\{-\frac{(x - x(s_{n_0}))^2 + (y - y(s_{n_0}))^2}{2\sigma^2}\right\}. \tag{5.12}$$

5.3.1 The integration regions

For classical constellation as shown in Figure 5.1, the region of integration is clear and regular. It is obvious that all the first quadrant is the region of s_1, as shown in Figure 5.4. The region of integration is denoted by $(x, y) \in ([0, \infty), [0, \infty))$ and the integral value could be computed by the function $\mathrm{erfc}(\cdot)$. However, when it comes

to other complex cases, as shown in Figure 5.5, the division of integral region is no longer a trivial task. Although the boundary of integral region is not constant, we can calculate the double integral by analyzing the geometric properties.

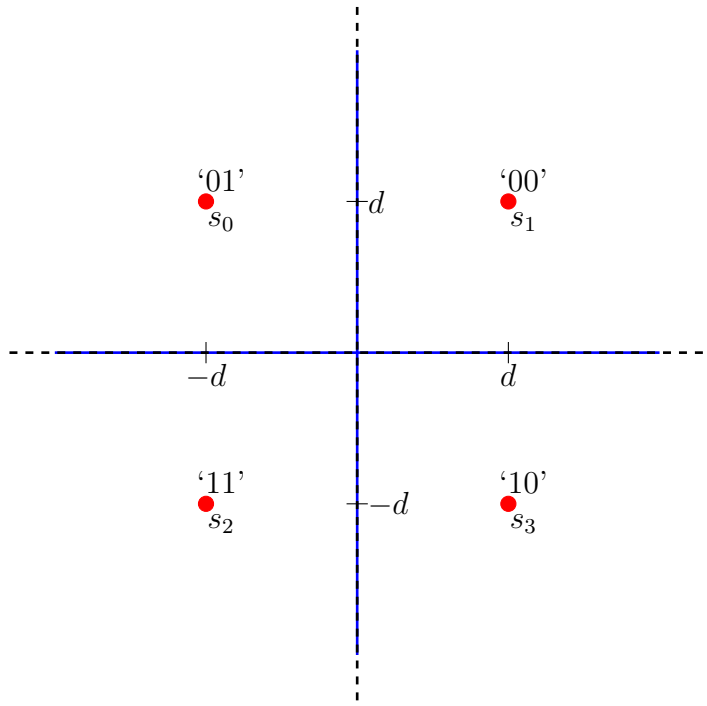


Figure 5.4: Integral region of 4-QAM.

Unfortunately, it becomes infeasible in our optimization problem since we need to compute the BER value several times in each iteration. To calculate each element of matrix B, we have to calculate the integral of a probability distribution function (pdf) in a relative area. However, the region of integration could be extremely irregular since the coordinates of the symbols are changed continuously. It is difficult to divide the region and numerical integration can cost much more time. Moreover, the partition structure could differ from time to time in the optimization process.

5.3.2 Approximation method for the SER matrix

To overcome such difficulties, we propose a new approach to approximate the SER matrix. Our ultimate goal is to figure out a way to approximate the SER value

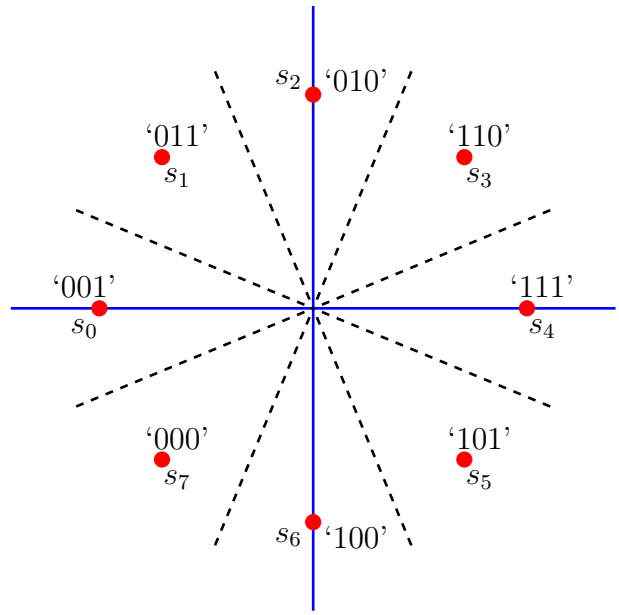


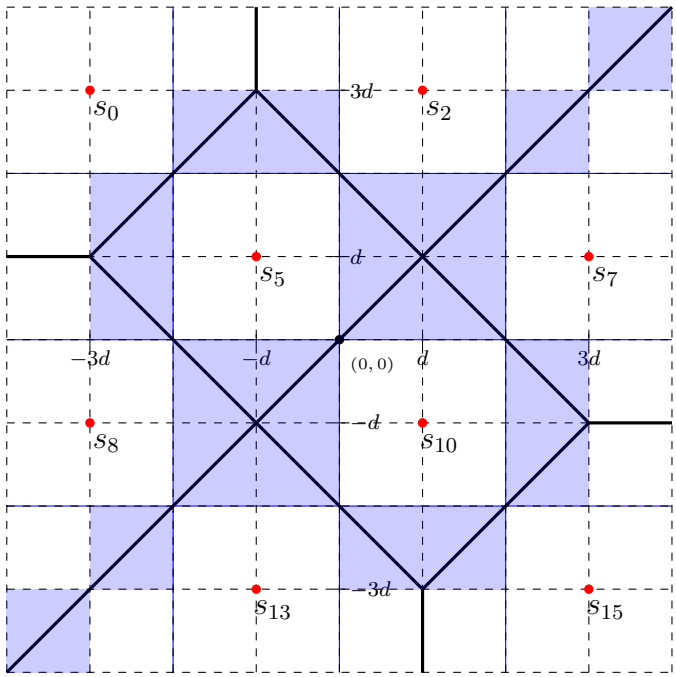
Figure 5.5: Integral region of 8-PSK

without the identification of the exact threshold lines between adjoint regions. In our method, the whole constellation plane is divided into small square regions. The probabilities that each transmitted signal falls into each of the small square regions is computed. This step is the same as the calculations of the probabilities where erfc$(\cdot)$ function is adopted. Denote the small square regions as $c_0, c_1, c_2, \ldots, c_{m-1}$ andall the candidate symbols as $s_0, s_1, s_2, \ldots, s_{N-1}$ where m is the number of square regions and N is the number of the candidate symbols. Normally, $N \ll m$ holds to ensure the accuracy of the estimated SER value. The resulting matrix of probability values is denoted by

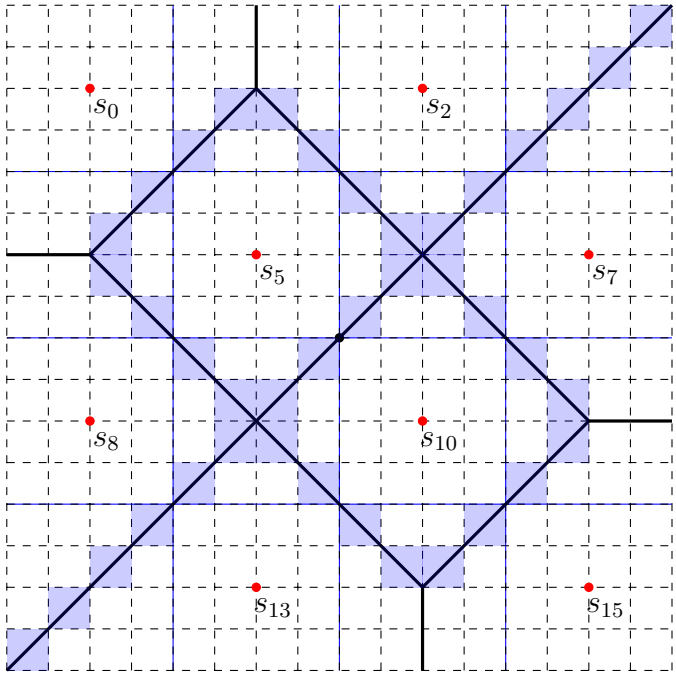
$$B_0 = \begin{bmatrix} p(c_0|s_0) & p(c_1|s_0) & p(c_2|s_0) & \cdots & p(c_{m-1}|s_0) \\ p(c_0|s_1) & p(c_1|s_1) & p(c_2|s_1) & \cdots & p(c_{m-1}|s_1) \\ p(c_0|s_2) & p(c_1|s_2) & p(c_2|s_2) & \cdots & p(c_{m-1}|s_2) \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ p(c_0|s_{N-1}) & p(c_1|s_{N-1}) & p(c_2|s_{N-1}) & \cdots & p(c_{m-1}|s_{N-1}) \end{bmatrix}$$

which is a N-by-m matrix. Figure 5.6

Next, we have to determine which symbol is closest to each of these small square



(a) Coarse grid block



(b) Fine grid block

Figure 5.6: The division of the whole constellation plane

regions. Roughly speaking, this step could be achieved by finding the largest value of each column of B_0. For example, if $p(c_0|s_i)$ is the largest one of Column 1, we then know that s_i is one of the nearest symbols to square c_0. The values in Column 1 of B_0 should be added into Column s_i in Matrix B. By the combination of columns in matrix B_0, matrix B could be denoted by:

$$B = \begin{bmatrix} p(s_0|s_0) & p(s_1|s_0) & p(s_2|s_0) & \cdots & p(s_{N-1}|s_0) \\ p(s_0|s_1) & p(s_1|s_1) & p(s_2|s_1) & \cdots & p(s_{N-1}|s_1) \\ p(s_0|s_2) & p(s_1|s_2) & p(s_2|s_2) & \cdots & p(s_{N-1}|s_2) \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ p(s_0|s_{N-1}) & p(s_1|s_{N-1}) & p(s_2|s_{N-1}) & \cdots & p(s_{N-1}|s_{N-1}) \end{bmatrix}$$

Problems may occur when square regions are allocated to symbols. Some square regions cannot be assigned strictly to a single symbol. Actually, only if all the points in one small square region are closest to the same symbol, we can allocate the square region to that symbol. For the small square regions that cannot be completely subdivided to a single symbol, i.e., different parts of such square regions are close to different symbols (blue squares in Figure 5.6), we need further analysis.

As shown in Figure 5.7, there are only two symbols. All of the squares should be allocated to s_1 and s_2 based on the distances. There are 4 square regions denoted by A, B, C and D which contain the threshold of the integral region. Assume A and C is allocated to s_2 while B and D belong to s_1. The SER with respect to square A is $P(A|s_1) = P(A_1|s_1) + P(A_2|s_1)$ but the true value is $P(A_1|s_2) + P(A_2|s_1)$. Since $P(A_1|s_2) < P(A_1|s_1)$ and the Hamming distance is the same, we overestimate the SER. This holds for all of A, B, C and D. If the square region becomes smaller, the blue area becomes smaller. Thus, one possible way is that we first find the small square regions that cannot be divided and then divide these regions into smaller squares. As long as we make the square small enough, the error is negligible.

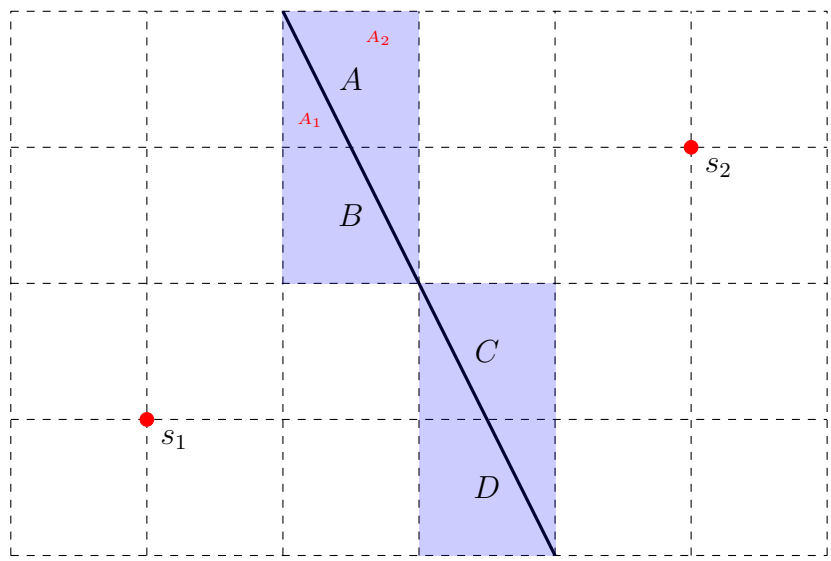


Figure 5.7: Sketch of a square region partitioned by two symbols

5.3.3 Comparison between simulation and proposed approximation

The simulation method is also known as Monte Carlo method. In the simulation method, one should first generate a large sum of samples with respect to each symbol. Then, we should decide which symbol the samples should belong to. Since all the samples are generated according to the probability density function, all the samples have the same weight. The proposed approximation method is also a sampling method where the samples are evenly distributed and the weights are the probabilities that each symbol falls into the region. Table 5.1 shows the comparison of these two methods. Note that the computation amount for each sqaure/sample is similar except that in the first and last step the computation for the square is slightly complex. However, in the proposed approximation method, we only need to deal with m_1 squares while in the simulation program we have to deal with $m_2 \times N$ samples. In general, $m_1 < m_2$ so that $m_2 \times N \gg m_1$. Therefore, the computation amount of the proposed approximation method is much less than the simulation method.

Approximation method	Simulation method
Divide the whole domain into m_1 squares	Generate m_2 samples for all N symbols ($m_2 \times m$ samples in total)
For each square, compute the probabilities that all the symbols falling into the square	For each sample, compute the distances between the sample and all the symbols
For each square, find the maximum one of all the probabilities with respect to the square	For each sample, find the minimum one of all the distances with respect to the sample
For all the squares, update a column of Matrix B by $p(s_i\|\cdot) = p(s_i\|\cdot) + p(c_i\|\cdot)$, where c_i is the current square which has been divided to s_i	For all the samples, update the count by $\text{count}(s_i\|s_j) = \text{count}(s_i\|s_j)+1$ where s_i is the symbol to which the current sample is divided and s_j is the symbol used to generate the current sample. $p(s_i\|s_j) = \text{count}(s_i\|s_j)/m_2$

Table 5.1: Comparison of the two methods (nearest symbol decoding strategy)

5.4 Simulation

In this section, we use two examples to illustrate how the proposed framework works and verify the effectiveness of our algorithm.

5.4.1 Example 1: 8-PSK Constellation

Firstly, we study the 8-PSK constellation in which information is transmitted as one of eight symbols and each symbol represents 3 bits of data. The Gray coded 8-PSK constellation can be found in [3]. First of all, rough optimization is applied to construct the initial assignment solution for the optimization process when E_b/N_0 is equal to 1. Figure 5.8 displays the positions of symbols in both the 8-PSK constellation and roughly optimized constellation. In the roughly optimized constellation, one symbol is close to the original and others are uniformly separated. Compared to the 8-PSK constellation, the roughly optimized constellation has a lower average symbol power.

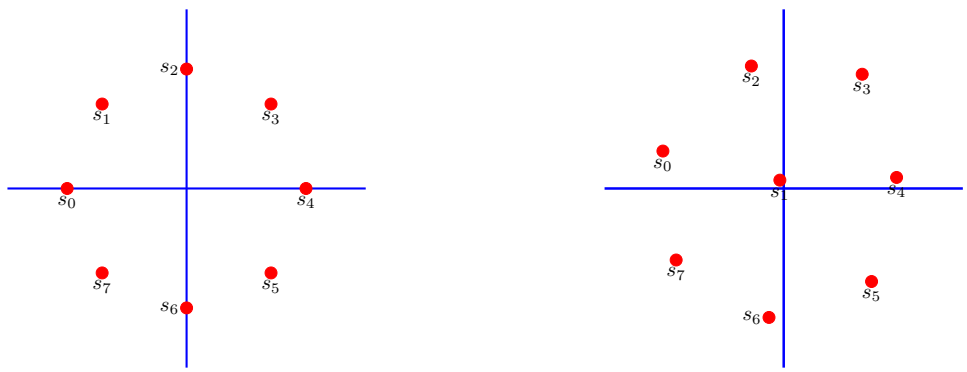


Figure 5.8: Coordinates in terms of the rough optimization

After that, Algorithm 7 is applied to continue optimizing the constellation with different E_b/N_0 values. In this example, E_b/N_0 is equal to 1,6 and 9, respectively. Since the proposed BER approximation method is significantly faster than the classical sampling method, the searching phase can be finished in a short time. Taking this advantage, it is able to implement the searching phase with different initial assignment solution such that the local optima can be more easily avoided. Figure 5.9 shows the optimized coordinates in comparison with coordinates of the 8-PSK constellation. It is evident that the optimized constellation with $E_b/N_0 = 1$ has a different pattern among the three cases. It might be caused by a large noise power when E_b/N_0 is small. For all three cases, we can observe that the average length of adjacent symbols in the optimized constellation becomes larger than the counterpart in the 8-PSK constellation. For the optimized constellations, the increase in the average length of adjacent symbols strengthens the discrimination ability of received symbols in the demodulation phase. Table 5.2 shows the coordinate value in Figure 5.9. Figure 5.10 displays the BER plot in terms of optimized constellations and the Gray coded 8-PSK constellation. It is observed that all three optimized constellations outperform the Gray coded 8-PSK constellation. When E_b/N_0 is large, the optimization case with $E_b/N_0 = 9$ has the best performance; The optimization case

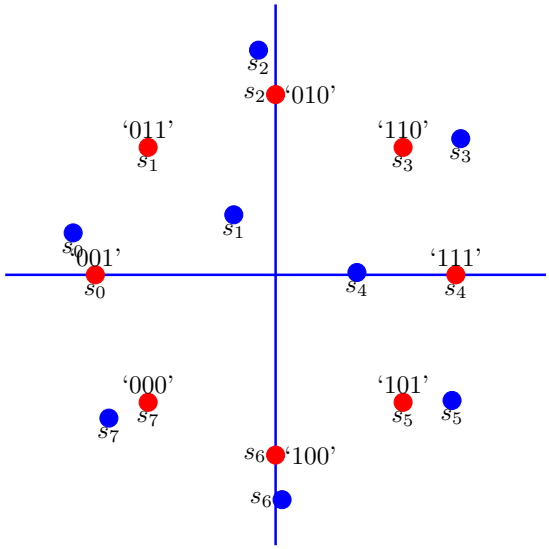
with $E_b/N_0 = 1$ can beat the 8-PSK constellation but has a bad performance among three optimization cases. Table 5.3 demonstrates the BER value in Figure 5.9. It is noted that the optimization case has the best performance if E_b/N_0 is close to the value of E_b/N_0 which is used in the optimization process. To be specific, the optimization case $E_b/N_0 = 1$ has the smallest BER value when E_b/N_0 is chosen as 0,1,2,3 and 4; The optimization case $E_b/N_0 = 6$ has the smallest BER value when E_b/N_0 is chosen as 5,6 and 7; The optimization case $E_b/N_0 = 9$ has the smallest BER value when E_b/N_0 is chosen as 8,9,10 and 11.

Symbol	8-PSK Gray coding	Optimized $E_b/N_0 = 1$	Optimized $E_b/N_0 = 6$	Optimized $E_b/N_0 = 9$
s_0	(-1 , 0)	(-1.12310 , 0.23265)	(-1.01165 , 0.31236)	(-1.01347 , 0.33996)
s_1	(-0.70711 , 0.70711)	(-0.23197 , 0.33298)	(-0.03005 , 0.07094)	(-0.01376 , 0.03233)
s_2	(0 , 1)	(-0.09473 , 1.24648)	(-0.26951 , 1.02581)	(-0.29442 , 1.00870)
s_3	(0.70711 , 0.70711)	(1.02706 , 0.75537)	(0.65612 , 0.95568)	(0.62903 , 0.94888)
s_4	(1 , 0)	(0.45114 , 0.01305)	(0.94556 , 0.09317)	(0.98916 , 0.10125)
s_5	(0.70711 , -0.70711)	(0.97876 , -0.69683)	(0.73703 , -0.77846)	(0.74832 , -0.78421)
s_6	(0 , -1)	(0.03586 , -1.24728)	(-0.12371 , -1.08277)	(-0.12870 , -1.07147)
s_7	(-0.70711 , -0.70711)	(-0.92474 , -0.79491)	(-0.90231 , -0.59865)	(-0.90785 , -0.57677)

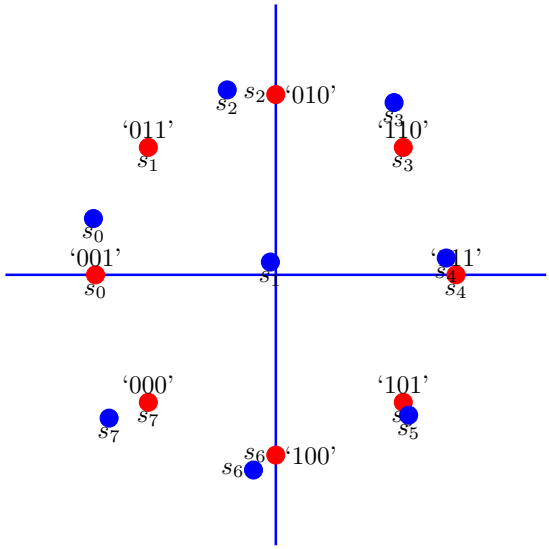
Table 5.2: Optimized coordinates in terms of 8-PSK constellation

E_b/N_0	8-PSK Gray coding	Optimized $E_b/N_0 = 1$	Optimized $E_b/N_0 = 6$	Optimized $E_b/N_0 = 9$
0	1.227125E-01	**1.186506E-01**	1.282775E-01	1.308743E-01
1	1.008216E-01	**9.329659E-02**	1.014227E-01	1.037112E-01
2	8.063599E-02	**7.043466E-02**	7.614939E-02	7.800683E-02
3	6.227582E-02	**5.067719E-02**	5.359220E-02	5.495221E-02
4	4.592821E-02	**3.444873E-02**	3.480939E-02	3.567957E-02
5	3.189651E-02	2.190113E-02	**2.048511E-02**	2.094589E-02
6	2.051646E-02	1.286765E-02	**1.068664E-02**	1.086622E-02
7	1.198388E-02	6.887590E-03	**4.815863E-03**	4.847723E-03
8	6.205802E-03	3.301728E-03	1.817744E-03	**1.800602E-03**
9	2.765191E-03	1.389000E-03	5.534628E-04	**5.355624E-04**
10	1.021158E-03	5.006663E-04	1.296658E-04	**1.215752E-04**
11	2.981514E-04	1.502283E-04	2.199100E-05	**1.980937E-05**

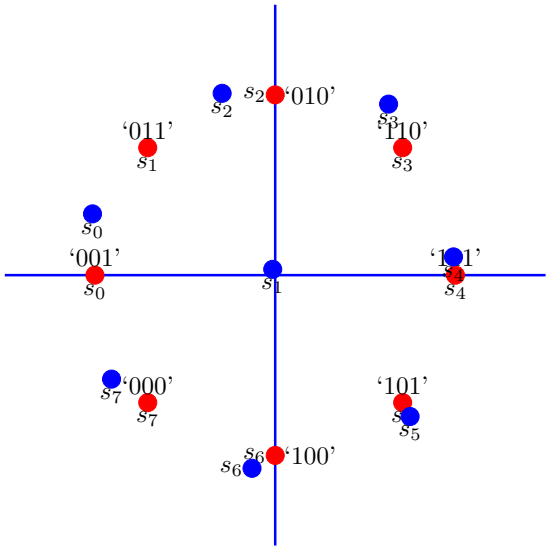
Table 5.3: BER performance of constellations with 8 symbols



(a) Optimization with $E_b/N_0 = 1$



(b) Optimization with $E_b/N_0 = 6$



(c) Optimization with $E_b/N_0 = 9$

Figure 5.9: Optimized coordinates in terms of 8-PSK constellation where red nodes denote Gray coded 8-PSK constellation [3] and blue nodes denote the optimized constellations

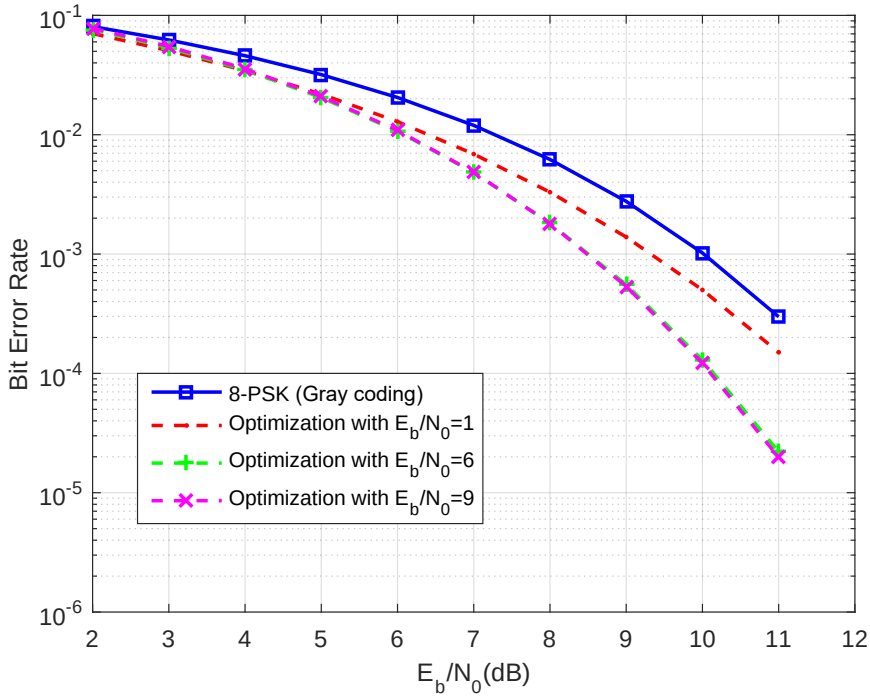


Figure 5.10: BER polt of constellations with 8 symbols

5.4.2 Example 2: 16-QAM Constellation

In this subsection, we consider the 16-QAM constellation in which information is transimitted as one of 16 symbols and each symbol represents 4 bits of data. Apart from the optimization of the symbol coordinates, in this subsection, we also discuss the effectiveness of the approximation calculation method of BER. Before approximately calculating the BER value, the whole constellation plane has to be divided into small square regions (cells). For each cell, it is allocated to the selected symbol which is nearest to its central point. The allocation process can be achieved by finding the largest element in each column of B_0. The error brought by this approximation has been analyzed in Figure 5.7. Therefore, we will overestimate the SER value in general. In order to find out the influence of the number of square regions on BER accuracy, we divide the whole constellation plane into different numbers of square regions. To be precise, we use q to represent the total number of cells in the

constellation plane. We consider 4 constellations whose symbols are all located at random points. The computation result of BER is shown in Table 5.4. One can figure out that the results of the proposed approximation method can match the simulation results with a large q. When $q \geqslant 65536$, the approximation results are accurate enough for searching procedure.

As illustrated before, we divided the whole plane into small square regions and then decide which selected symbol each square should belong to. Firstly, we directly divide each square to the symbol which is nearest to the central point of the square. This is achieved by finding the largest element in each column of B_0. The error brought by this approximation has been analyzed in Figure 5.7. Therefore, we will overestimate the SER value in general. In order to find out the influence of the number of square regions on BER accuracy, we divide the whole domain into different numbers of square regions. To be precise, we use q to represent the number of squares we applied. We consider 4 constellations whose symbols are all located at random points. The computation result of BER is shown in Table 5.4. One can figure out that the results of the proposed approximation method can match the simulation results. When $q \geqslant 65536$, the error caused by overestimation is less than 0.09% such that the approximation method is accurate enough for searching procedure.

Normally, the initial guess can be generated by editing the Gray Coding assignment of 16-QAM constellation. Meanwhile, we can also apply our method to construct the initial guess. We start with several different sets of initial values, some of which were very poorly aligned. The result is shown in Figure 5.11 where the solid point represents the initial solution and the hollow point represents the construction result. It is noted that we can get a relatively good construction result although the initial guess is very poor.

Based on the permutations of Gray Coding, we construct the initial guess by adding random perturbations to the coordinates of symbols in 16-QAM constellation.

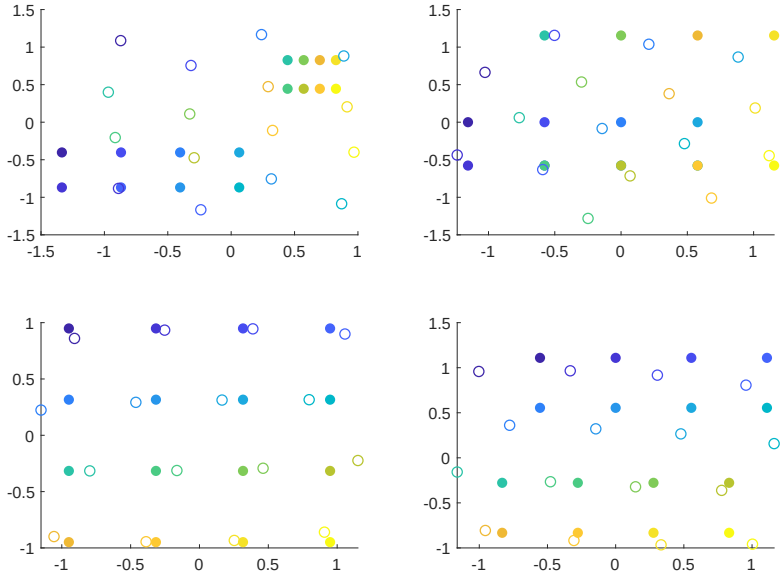


Figure 5.11: The construction phase of the initial guess

The value of E_b/N_0 is chosen as 1,6 and 9, respectively. The optimized coordinates are presented in Table 5.6. Figure 5.12 shows the BER values with the change of $E_b/N_0(dB)$. As we can see, the BER value in terms of the optimization case with $E_b/N_0 = 9$ is greater than the counterpart of Gray coding case when E_b/N_0 is smaller. With the increase of E_b/N_0 value, the BER value in terms of the optimization case with $E_b/N_0 = 9$ drops down faster than the counterpart of Gray coding. When E_b/N_0 is equal to 11, it can be observed that the optimization case with $E_b/N_0 = 9$ outperforms the Gray coding. On the contrary, the optimization case with $E_b/N_0 = 1$ has a bad performance when E_b/N_0 is large but it is a superior case when E_b/N_0 is small. The optimization case with $E_b/N_0 = 6$ has the moderate performance when E_b/N_0 is increased from 0 to 11. Table 5.5 presents the BER value in Figure 5.12. As we can see, the optimization case with $E_b/N_0 = 1$ has the smallest BER when E_b/N_0 is between 0 and 3. The optimization case with $E_b/N_0 = 6$ has the smallest BER when E_b/N_0 is between 4 and 7. The optimization case with $E_b/N_0 = 9$ has the smallest BER when E_b/N_0 is between 8 and 9. It is evident that our optimization

method can achieve the best performance if E_b/N_0 is close to the value of E_b/N_0 using in the optimization process.

	E_b/N_0	Simulation	q=64	Error	q=576	Error	q=1024	Error	q=4096	Error	q=65536	Error
X_1	1	2.329E-01	2.681E-01	15.09%	2.412E-01	3.58%	2.336E-01	0.30%	2.331E-01	0.08%	2.329E-01	0.01%
	3	2.111E-01	2.515E-01	19.14%	2.062E-01	2.30%	2.116E-01	0.25%	2.112E-01	0.07%	2.111E-01	0.01%
	5	1.969E-01	2.410E-01	22.35%	2.071E-01	5.18%	1.972E-01	0.12%	1.970E-01	0.03%	1.970E-01	0.01%
	7	1.872E-01	2.360E-01	26.04%	1.976E-01	5.56%	1.873E-01	0.03%	1.872E-01	0.01%	1.872E-01	0.00%
	9	1.799E-01	2.346E-01	30.41%	1.780E-01	1.01%	1.799E-01	0.01%	1.799E-01	0.01%	1.799E-01	0.01%
X_2	1	1.949E-01	2.071E-01	6.26 %	1.963E-01	0.72%	1.957E-01	0.38%	1.951E-01	0.12%	1.949E-01	0.00%
	3	1.469E-01	1.654E-01	12.58%	1.491E-01	1.48%	1.481E-01	0.77%	1.473E-01	0.25%	1.470E-01	0.04%
	5	1.041E-01	1.285E-01	23.50%	1.070E-01	2.79%	1.055E-01	1.44%	1.045E-01	0.42%	1.041E-01	0.03%
	7	6.884E-02	9.763E-02	41.81%	7.242E-02	5.19%	7.074E-02	2.76%	6.939E-02	0.80%	6.889E-02	0.07%
	9	4.268E-02	7.259E-02	70.06%	4.664E-02	9.27%	4.483E-02	5.03%	4.327E-02	1.38%	4.272E-02	0.08%
X_3	1	2.255E-01	2.350E-01	4.22 %	2.261E-01	0.29%	2.267E-01	0.57%	2.253E-01	0.07%	2.254E-01	0.01%
	3	1.822E-01	2.002E-01	9.90 %	1.839E-01	0.96%	1.839E-01	0.94%	1.820E-01	0.08%	1.822E-01	0.01%
	5	1.421E-01	1.715E-01	20.67%	1.452E-01	2.16%	1.441E-01	1.43%	1.420E-01	0.09%	1.421E-01	0.00%
	7	1.084E-01	1.502E-01	38.58%	1.127E-01	3.97%	1.106E-01	2.03%	1.083E-01	0.02%	1.083E-01	0.01%
	9	8.252E-02	1.357E-01	64.42%	8.751E-02	6.04%	8.475E-02	2.70%	8.265E-02	0.16%	8.249E-02	0.04%
X_4	1	2.134E-01	2.270E-01	6.35 %	2.150E-01	0.75%	2.142E-01	0.38%	2.133E-01	0.06%	2.134E-01	0.02%
	3	1.703E-01	1.894E-01	11.21%	1.726E-01	1.35%	1.715E-01	0.72%	1.704E-01	0.05%	1.704E-01	0.04%
	5	1.319E-01	1.568E-01	18.86%	1.351E-01	2.36%	1.334E-01	1.09%	1.322E-01	0.21%	1.320E-01	0.06%
	7	1.014E-01	1.300E-01	28.28%	1.053E-01	3.87%	1.026E-01	1.23%	1.018E-01	0.41%	1.014E-01	0.00%
	9	7.921E-02	1.087E-01	37.25%	8.409E-02	6.17%	8.010E-02	1.12%	7.985E-02	0.81%	7.925E-02	0.05%

Table 5.4: BER approximation of constellation with 16 symbols of arbitrary coordinates

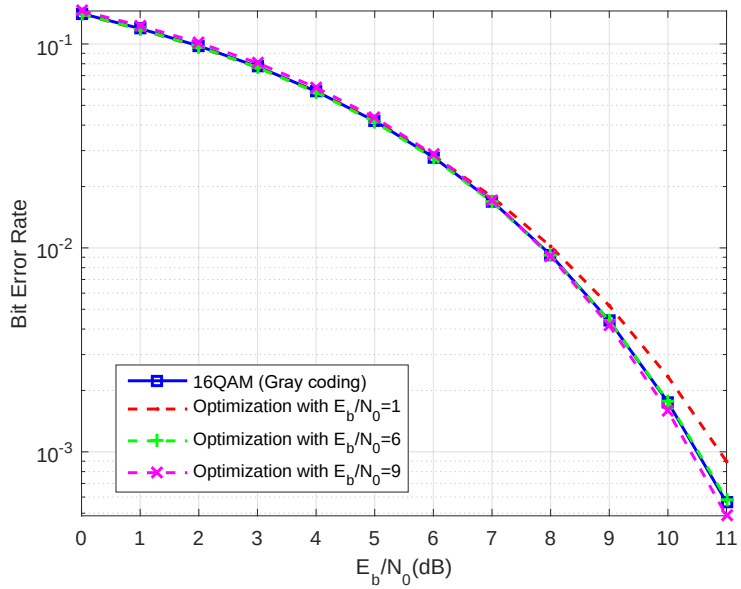


Figure 5.12: BER plot in terms of 16QAM constellation with optimized coordinates

5.5 Final remark

In this chapter, the constellation design problem is formulated as a dynamic optimization problem in which symbol coordinates are the decision variables. A novel approximation calculation method of BER is proposed. Taking advantage of this fast approximation method, symbol coordinates can be optimized by the gradient-based method. Since non-convex optimization problems can be trapped by the local optima, the initial solution plays an important role in the optimization process. We propose a rough optimization method to construct a good initial assignment for our constellation design problem. Two examples have been demonstrated. According to the simulation results, it is evident that our proposed method can truly reduce the BER value such that the performance of constellations is improved.

E_b/N_0	16-QAM Gray coding	Optimized $E_b/N_0 = 1$	Optimized $E_b/N_0 = 6$	Optimized $E_b/N_0 = 9$
0	1.409816E-01	**1.398864E-01**	1.402395E-01	1.451466E-01
1	1.189974E-01	**1.178028E-01**	1.181982E-01	1.229909E-01
2	9.774185E-02	**9.668147E-02**	9.695337E-02	1.014540E-01
3	7.745306E-02	**7.671716E-02**	7.673380E-02	8.069035E-02
4	5.862374E-02	5.832728E-02	**5.801774E-02**	6.115932E-02
5	4.189276E-02	4.206408E-02	**4.142781E-02**	4.357385E-02
6	2.787133E-02	2.845124E-02	**2.755677E-02**	2.871377E-02
7	1.696673E-02	1.781618E-02	**1.679100E-02**	1.716774E-02
8	9.247214E-03	1.017490E-02	9.179086E-03	**9.101465E-03**
9	4.390336E-03	5.207286E-03	4.387219E-03	**4.161947E-03**
10	1.754151E-03	2.338293E-03	1.775785E-03	**1.587952E-03**
11	5.647061E-04	8.972542E-04	5.852527E-04	**4.855900E-04**

Table 5.5: BER performance of constellations with 16 symbols

Symbol	16-QAM Gray coding	Optimized $E_b/N_0 = 1$	Optimized $E_b/N_0 = 6$	Optimized $E_b/N_0 = 9$
s_0	(-3 , 3)	(-2.84871 , 3.09416)	(-3.01322 , 3.01322)	(-3.24214 ,2.98474)
s_1	(-1 , 3)	(-0.78004 , 2.93509)	(-0.94678 , 3.01322)	(-1.13740 ,3.04719)
s_2	(1 , 3)	(0.88392 , 2.82993)	(0.94678 , 3.01322)	(0.81241 , 2.87437)
s_3	(3 , 3)	(2.89689 , 3.02334)	(3.01322 , 3.01322)	(2.84824 , 3.05089)
s_4	(-3 , 1)	(-3.12430 , 0.89205)	(-3.01322 , 0.94678)	(-2.53005 ,1.04395)
s_5	(-1 , 1)	(-0.85518 , 0.85497)	(-0.94678 , 0.94678)	(-0.54158 ,0.92441)
s_6	(1 , 1)	(0.85113 , 0.82136)	(0.94678 , 0.94678)	(1.44189 , 0.90814)
s_7	(3 , 1)	(3.22890 , 0.75927)	(3.01322 , 0.94678)	(3.69928 , 1.02225)
s_8	(-3 , -1)	(-2.89333 , -0.81177)	(-3.01322 , -0.94678)	(-3.71150 ,-0.64790)
s_9	(-1 , -1)	(-0.88566 , -0.84277)	(-0.94678 , -0.94678)	(-1.38727 ,-0.95861)
s_{10}	(1 , -1)	(0.79637 , -0.85399)	(0.94678 , -0.94678)	(0.69724 , -1.09450)
s_{11}	(3 , -1)	(2.72213 , -0.90816)	(3.01322 , -0.94678)	(2.81108 , -0.90670)
s_{12}	(-3 , -3)	(-3.09461 , -2.89724)	(-3.01322 , -3.01322)	(-3.07787 ,-2.74717)
s_{13}	(-1 , -3)	(-0.93265 , -3.03113)	(-0.94678 , -3.01322)	(-1.02318 ,-3.06491)
s_{14}	(1 , -3)	(0.79092 , -3.05484)	(0.94678 , -3.01322)	(0.93607 , -3.28671)
s_{15}	(3 , -3)	(2.97662 , -2.95908)	(3.01322 , -3.01322)	(2.98461 , -2.94398)

Table 5.6: Optimized coordinates of constellations with 16 symbols

Chapter 6

Conclusions and future work

This thesis is concerned with the speech enhancement system in the presence of noisy data. It contains two major parts. The first one is concerned with the dealing of the environment noise. The second one is concerned with the dealing of the channel noise.

6.1 Conclusions

The main conclusions of this thesis are listed below.

- we propose a novel distributed MMSE beamformer using gossip algorithms. We prove the convergence of gossip algorithms when WASN is a complete graph. Our proposed distributed MMSE beamformers outperform the distributed delay-and-sum beamformer and MVDR beamformer in a simulated reverberation environment. Based on the blockchain technique, a data protection scheme is also proposed to avoid faulty data transmissions. To illustrate the effectiveness of the proposed beamformer with blockchain protection, we simulate a typical scenario in a square office room with reverberation. The experimental results show that blockchain data protection is able to secure the quality of the output signals from a distributed beamformer in a relatively poor network environment. In addition, we showed that greedy algorithm can

perform better than the randomized gossip algorithm if additional sources are used to receive information from all neighbors.

- We propose an enhancement method for the BSS system via the optimization of microphone locations when the nonlinear mixing problem exists. We fill the research gap by studying the effect of sensor placement on the performance of BSS. Spatial information can be fully exploited via an optimized array geometry. An optimization strategy is proposed to optimize the performance of BSS for the best sensor placement on the allowable placement space as the number of sensors and reverberation time change. In order to solve the nonlinear mixing problem, we introduce a novel hybrid descent method. Results show that the optimized array consistently yields a greater suppression across the different reverberation times compared to an unoptimized linear configuration.

- We investigate the optimization methods in terms of the M-QAM constellation in additive white noise channel. A novel method is proposed to optimize the constellation in the digital communication system. Using the SER matrix and Hamming distance matrix, the design of the constellation is formulated as a QAP. We apply the GARSP algorithm to solve the low-dimensional optimization problem on the design of constellations. Since the assignment problem in constellations can be an NP-hard problem when a large number of symbols is inside the constellation, we propose a tailor-made method based on SA algorithm. Simulation results show that the Gray coding in terms of 16-QAM constellation can be easily found. Besides that, our tailor-made method can derive the optimal assignment solutions for the Honeycomb-structured constellation. Compared to the Gray coding of 8-PSK and 16-QAM constellation, BER can be significantly reduced in the optimized HC31 constellation.

- The constellation design problem is solved via the optimization of symbol coordinates. In particular, we have formulated the original assignment problem to a dynamic optimization problem where the decision variables are the coordinates of the symbols. In order to solve this dynamic optimization problem, the optimization process is divided into two phases: the construction phase and the optimization phase. In the construction phase, We propose a rough optimization method to construct a good initial assignment for our constellation design problem. Simulation results show that our proposed methods can truly improve the performance of constellations.

6.2 Future works

Related topics for future work are listed below.

- In the convergence analysis of gossip algorithms, WASN is assumed to be a complete graph. However, the connection in the wireless network may not stable such that the assumption of the complete graph may fail. It is of interest to investigate the robustness of the gossip algorithm in a dynamic wireless network.

- It is of interest to study and optimize the proposed beamformer or BSS system with voice control accuracy as the performance criterion in smart systems.

Bibliography

[1] E. Aarts and J. Korst. Simulated annealing and boltzmann machines. 1988.

[2] K. Akdagli. Null steering of linear antenna arrays using a modified tabu search algorithm. *Progress In Electromagnetics Research*, 33:167–182, 2001.

[3] J. B. Anderson, T. Aulin, and C.-E. Sundberg. *Digital phase modulation*. Springer Science & Business Media, 2013.

[4] W. N. Anderson Jr and T. D. Morley. Eigenvalues of the laplacian of a graph. *Linear and multilinear algebra*, 18(2):141–145, 1985.

[5] S. Araki, S. Makino, Y. Hinamoto, R. Mukai, T. Nishikawa, and H. Saruwatari. Equivalence between frequency-domain blind source separation and frequency-domain adaptive beamforming for convolutive mixtures. *EURASIP Journal on Applied Signal Processing*, 2003:1157–1166, 2003.

[6] S. Araki, R. Mukai, S. Makino, T. Nishikawa, and H. Saruwatari. The fundamental limitation of frequency domain blind source separation for convolutive mixtures of speech. *IEEE Transactions on Speech and Audio Processing*, 11(2):109–116, 2003.

[7] N. Avlonitis, E. Yeatman, M. Jones, and A. Hadjifotiou. Multilevel amplitude shift keying in dispersion uncompensated optical systems. *IEE Proceedings-Optoelectronics*, 153(3):101–108, 2006.

[8] J. K. Baksalary and O. M. Baksalary. Idempotency of linear combinations of two idempotent matrices. *Linear Algebra and its Applications*, 321(1-3):3–7, 2000.

[9] N. V. Banichuk. *Introduction to optimization of structures*. Springer Science & Business Media, 2013.

[10] M. S. Bartlett. An inverse matrix adjustment arising in discriminant analysis. *The Annals of Mathematical Statistics*, 22(1):107–111, 1951.

[11] A. J. Bell and T. J. Sejnowski. An information-maximization approach to blind separation and blind deconvolution. *Neural computation*, 7(6):1129–1159, 1995.

[12] J. Benesty, S. Makino, and J. Chen. *Speech enhancement*. Springer Science & Business Media, 2005.

[13] A. Bertrand. Applications and trends in wireless acoustic sensor networks: A signal processing perspective. In *2011 18th IEEE symposium on communications and vehicular technology in the Benelux (SCVT)*, pages 1–6. IEEE, 2011.

[14] A. Bertrand and M. Moonen. Distributed node-specific lcmv beamforming in wireless sensor networks. *IEEE Transactions on Signal Processing*, 60(1):233–246, 2011.

[15] F. Black. Noise. *The journal of finance*, 41(3):528–543, 1986.

[16] S. Boyd, A. Ghosh, B. Prabhakar, and D. Shah. Analysis and optimization of randomized gossip algorithms. In *2004 43rd IEEE Conference on Decision and Control (CDC)(IEEE Cat. No. 04CH37601)*, volume 5, pages 5310–5315. IEEE, 2004.

[17] S. Boyd, A. Ghosh, B. Prabhakar, and D. Shah. Randomized gossip algorithms. *IEEE/ACM Transactions on Networking (TON)*, 14(SI):2508–2530, 2006.

[18] J. H. Braslavsky, R. H. Middleton, and J. S. Freudenberg. Feedback stabilization over signal-to-noise ratio constrained channels. *IEEE Transactions on Automatic Control*, 52(8):1391–1403, 2007.

[19] T. Braun, T. Voigt, and A. Dunkels. Tcp support for sensor networks. In *2007 Fourth Annual Conference on Wireless on Demand Network Systems and Services*, pages 162–169. IEEE, 2007.

[20] C. Buratti, A. Conti, D. Dardari, and R. Verdone. An overview on wireless sensor networks technology and evolution. *Sensors*, 9(9):6869–6896, 2009.

[21] R. Burkard, M. Dell'Amico, and S. Martello. *Assignment Problems*. Society for Industrial and Applied Mathematics, 2012.

[22] R. E. Burkard, E. Cela, P. M. Pardalos, and L. S. Pitsoulis. The quadratic assignment problem. In *Handbook of combinatorial optimization*, pages 1713–1809. Springer, 1998.

[23] R. E. Burkard, S. E. Karisch, and F. Rendl. Qaplib–a quadratic assignment problem library. *Journal of Global optimization*, 10(4):391–403, 1997.

[24] R. E. Burkard and J. Offermann. Entwurf von schreibmaschinentastaturen mittels quadratischer zuordnungsprobleme. *Zeitschrift für Operations Research*, 21(4):B121–B132, 1977.

[25] G. Caire, G. Taricco, and E. Biglieri. Bit-interleaved coded modulation. *IEEE transactions on information theory*, 44(3):927–946, 1998.

[26] E. Cela. *The quadratic assignment problem: theory and algorithms*, volume 1. Springer Science & Business Media, 2013.

[27] H. Chan, A. Perrig, B. Przydatek, and D. Song. Sia: Secure information aggregation in sensor networks. *Journal of Computer Security*, 15(1):69–102, 2007.

[28] K. Y. Chan, C. K. F. Yiu, and S. Nordholm. Microphone configuration for beamformer design using the taguchi method. *Measurement*, 96:58–66, 2017.

[29] J. Chen. *Carrier recovery in burst-mode 16-qam*. PhD thesis, 2004.

[30] K. Chen, X. Yun, Z. He, and C. Han. Synthesis of sparse planar arrays using modified real genetic algorithm. *IEEE Transactions on Antennas and Propagation*, 55(4):1067–1073, 2007.

[31] E. C. Cherry. Some experiments on the recognition of speech, with one and with two ears. *The Journal of the acoustical society of America*, 25(5):975–979, 1953.

[32] J. Cho and P. J. Winzer. Probabilistic constellation shaping for optical fiber communications. *Journal of Lightwave Technology*, 37(6):1590–1607, 2019.

[33] A. Cichocki and S.-i. Amari. *Adaptive blind signal and image processing: learning algorithms and applications*. John Wiley & Sons, 2002.

[34] P. Comon, C. Jutten, and J. Herault. Blind separation of sources, part ii: Problems statement. *Signal processing*, 24(1):11–20, 1991.

[35] R. V. Cox, C. A. Kamm, L. R. Rabiner, J. Schroeter, and J. G. Wilpon. Speech and language processing for next-millennium communications services. *Proceedings of the IEEE*, 88(8):1314–1337, 2000.

[36] M. Crosby, P. Pattanayak, S. Verma, and V. Kalyanaraman. Blockchain technology: Beyond bitcoin. *APPLIED INNOVATION*, page 6, 2016.

[37] H. H. Dam, S. Nordholm, S. Y. Low, and A. Cantoni. Blind signal separation using steepest descent method. *IEEE Transactions on Signal Processing*, 55(8):4198–4207, 2007.

[38] G. M. Davis. *Noise reduction in speech applications*, volume 7. CRC press, 2002.

[39] J. Dickey and J. Hopkins. Campus building arrangement using topaz. *Transportation Research*, 6(1):59–68, 1972.

[40] A. G. Dimakis, S. Kar, J. M. Moura, M. G. Rabbat, and A. Scaglione. Gossip algorithms for distributed signal processing. *Proceedings of the IEEE*, 98(11):1847–1864, 2010.

[41] G. Doblinger. Optimized design of interpolated array and sparse array wideband beamformers. In *2008 16th European Signal Processing Conference*, pages 1–5. IEEE, 2008.

[42] S. Doclo, M. Moonen, T. Van den Bogaert, and J. Wouters. Reduced-bandwidth and distributed mwf-based noise reduction algorithms for binaural hearing aids. *IEEE Transactions on Audio, Speech, and Language Processing*, 17(1):38–51, 2009.

[43] A. Dorri, S. S. Kanhere, and R. Jurdak. Towards an optimized blockchain for iot. In *Proceedings of the Second International Conference on Internet-of-Things Design and Implementation*, pages 173–178. ACM, 2017.

[44] A. N. Elshafei. Hospital layout as a quadratic assignment problem. *Journal of the Operational Research Society*, 28(1):167–179, 1977.

[45] T. A. Feo and M. G. Resende. Greedy randomized adaptive search procedures. *Journal of global optimization*, 6(2):109–133, 1995.

[46] T. M. Fernández-Caramés and P. Fraga-Lamas. A review on the use of blockchain for the internet of things. *IEEE Access*, 6:32979–33001, 2018.

[47] W. Ford. *Numerical linear algebra with applications: Using MATLAB*. Academic Press, 2014.

[48] M. Ga-rey and D. Johnson. Computers and intractability: A guide to the theory of {NP}-completeness. 1979.

[49] S. Gannot, E. Vincent, S. Markovich-Golan, and A. Ozerov. A consolidated perspective on multimicrophone speech enhancement and source separation. *IEEE/ACM Transactions on Audio, Speech and Language Processing (TASLP)*, 25(4):692–730, 2017.

[50] B. L. Golden and C. C. Skiscim. Using simulated annealing to solve routing and location problems. *Naval Research Logistics Quarterly*, 33(2):261–279, 1986.

[51] N. Grbic, S. Nordholm, J. Nordberg, and I. Claesson. A new pilot-signal based space-time adaptive algorithm. In *In Proc. of IEEE International Conference on Communications, ICT 2001*, 2001.

[52] S. Haller, S. Karnouskos, and C. Schroth. The internet of things in an enterprise context. In *Future Internet Symposium*, pages 14–28. Springer, 2008.

[53] X. Han and M. Rashid. Gesture and voice control of internet of things. In *2016 IEEE 11th Conference on Industrial Electronics and Applications (ICIEA)*, pages 1791–1795. IEEE, 2016.

[54] L. Hanzo, W. Webb, and T. Keller. *Single-and Multi-carrier Quadrature Amplitude Modulation: Principles and Applications for Personal Communications, WATM and Broadcasting: 2nd*. IEEE Press-John Wiley, 2000.

[55] R. Heusdens, G. Zhang, R. C. Hendriks, Y. Zeng, and W. B. Kleijn. Distributed mvdr beamforming for (wireless) microphone networks using message passing. In *IWAENC 2012; International Workshop on Acoustic Signal Enhancement*, pages 1–4. VDE, 2012.

[56] A. Hyvärinen and E. Oja. A fast fixed-point algorithm for independent component analysis. *Neural computation*, 9(7):1483–1492, 1997.

[57] A. Hyvärinen and E. Oja. Independent component analysis: algorithms and applications. *Neural networks*, 13(4-5):411–430, 2000.

[58] R. T. Jones, T. A. Eriksson, M. P. Yankov, B. J. Puttnam, G. Rademacher, R. S. Luis, and D. Zibar. Geometric constellation shaping for fiber optic communication systems via end-to-end learning. *arXiv preprint arXiv:1810.00774*, 2018.

[59] I. Kamel and H. Juma. A lightweight data integrity scheme for sensor networks. *Sensors*, 11(4):4118–4136, 2011.

[60] D. Kempe, A. Dobra, and J. Gehrke. Gossip-based computation of aggregate information. In *44th Annual IEEE Symposium on Foundations of Computer Science, 2003. Proceedings.*, pages 482–491. IEEE, 2003.

[61] P. Koopman and T. Chakravarty. Cyclic redundancy code (crc) polynomial selection for embedded networks. In *International Conference on Dependable Systems and Networks, 2004*, pages 145–154. IEEE, 2004.

[62] T. C. Koopmans and M. Beckmann. Assignment problems and the location of economic activities. *Econometrica: journal of the Econometric Society*, pages 53–76, 1957.

[63] E. J. Kostelich and J. A. Yorke. Noise reduction: Finding the simplest dynamical system consistent with the data. *Physica D: Nonlinear Phenomena*, 41(2):183–196, 1990.

[64] C. Kyriakakis, P. Tsakalides, and T. Holman. Surrounded by sound. *IEEE Signal processing magazine*, 16(1):55–66, 1999.

[65] F. Lalem, M. Alshaikh, A. Bounceur, R. Euler, L. Laouamer, L. Nana, and A. Pascu. Data authenticity and integrity in wireless sensor networks based on a watermarking approach. In *The Twenty-Ninth International Flairs Conference*, 2016.

[66] B. P. Lathi. *Modern Digital and Analog Communication Systems 3e Osece*. Oxford University Press, Inc., 1998.

[67] E. L. Lawler. The quadratic assignment problem. *Management science*, 9(4):586–599, 1963.

[68] E. A. Lehmann, A. M. Johansson, and S. Nordholm. Reverberation-time prediction method for room impulse responses simulated with the image-source model. In *2007 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, pages 159–162. IEEE, 2007.

[69] J. Leng. Modelling and analysis on noisy financial time series. *Journal of Computer and Communications*, 2:64–69, 2014.

[70] A. Letchford, J. Gao, and L. Zheng. Filtering financial time series by least squares. *International Journal of Machine Learning and Cybernetics*, 4(2):149–154, 2013.

[71] Z. Li and K.-F. C. Yiu. Beamformer configuration design in reverberant environments. *Engineering Applications of Artificial Intelligence*, 47:81–87, 2016.

[72] Z. Li, K. F. C. Yiu, and S. Nordholm. On the indoor beamformer design with reverberation. *IEEE/ACM Transactions on Audio, Speech and Language Processing (TASLP)*, 22(8):1225–1235, 2014.

[73] U. A. Lindgren and H. Broman. Source separation using a criterion based on second-order statistics. *IEEE Transactions on Signal Processing*, 46(7):1837–1850, 1998.

[74] E. M. Loiola, N. M. M. de Abreu, P. O. Boaventura-Netto, P. Hahn, and T. Querido. A survey for the quadratic assignment problem. *European journal of operational research*, 176(2):657–690, 2007.

[75] S. Y. Low, S. Nordholm, and R. Togneri. Convolutive blind signal separation with post-processing. *IEEE Transactions on Speech and Audio Processing*, 12(5):539–548, 2004.

[76] S. Y. Low, C. K.-F. Yiu, and S. Nordholm. Second-order-based blind signal separation in reverberant environments. *International Journal of Electronics*, 102(9):1583–1593, 2015.

[77] S. Y. Low, K.-F. C. Yiu, and S. Nordholm. Beamspace blind signal separation for speech enhancement. *Optimization and engineering*, 10(2):313–330, 2009.

[78] S. Markovich-Golan, A. Bertrand, M. Moonen, and S. Gannot. Optimal distributed minimum-variance beamforming approaches for speech enhancement in wireless acoustic sensor networks. *Signal Processing*, 107:4–20, 2015.

[79] A. Masnadi-Shirazi and B. Rao. Independent vector analysis incorporating active and inactive states. In *2009 IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 1837–1840. IEEE, 2009.

[80] P. Meer, D. Mintz, A. Rosenfeld, and D. Y. Kim. Robust regression methods for computer vision: A review. *International journal of computer vision*, 6(1):59–70, 1991.

[81] Y. Meng, Z. Wang, W. Zhang, P. Wu, H. Zhu, X. Liang, and Y. Liu. Wivo: Enhancing the security of voice control system via wireless signal in iot environment. In *Proceedings of the Eighteenth ACM International Symposium on Mobile Ad Hoc Networking and Computing*, pages 81–90. ACM, 2018.

[82] J. Nocedal and S. Wright. *Numerical optimization*. Springer Science & Business Media, 2006.

[83] S. Nordholm, I. Claesson, and M. Dahl. Adaptive microphone array employing calibration signals: an analytical evaluation. *IEEE Transactions on Speech and Audio Processing*, 7(3):241–252, 1999.

[84] S. Nordholm, I. Claesson, and N. Grbić. Optimal and adaptive microphone arrays for speech input in automobiles. In *Microphone Arrays*, pages 307–329. Springer, 2001.

[85] L. Pardalos and M. Resende. A greedy randomized adaptive search procedure for the quadratic assignment problem. *Quadratic Assignment and Related Problems, DIMACS Series on Discrete Mathematics and Theoretical Computer Science*, 16:237–261, 1994.

[86] L. Parra and C. Spence. Convolutive blind separation of non-stationary sources. *IEEE transactions on Speech and Audio Processing*, 8(3):320–327, 2000.

[87] M. S. Pedersen, J. Larsen, U. Kjems, and L. C. Parra. Convolutive blind source separation methods. In *Springer handbook of speech processing*, pages 1065–1094. Springer, 2008.

[88] A. Perrig, R. Szewczyk, J. D. Tygar, V. Wen, and D. E. Culler. Spins: Security protocols for sensor networks. *Wireless networks*, 8(5):521–534, 2002.

[89] M. Pilkington. 11 blockchain technology: principles and applications. *Research handbook on digital transformations*, 225, 2016.

[90] G. Poltyrev. On coding without restrictions for the awgn channel. *IEEE Transactions on Information Theory*, 40(2):409–417, 1994.

[91] J. Postel. User datagram protocol. *Request for Comments, RFC 768, ISI*, 1980.

[92] H. Sawada, H. Kameoka, S. Araki, and N. Ueda. Multichannel extensions of non-negative matrix factorization with complex-valued data. *IEEE Transactions on Audio, Speech, and Language Processing*, 21(5):971–982, 2013.

[93] T. Schmidt. Shot-noise processes in finance. In *From statistics to mathematical finance*, pages 367–385. Springer, 2017.

[94] S. C. Shah. *Mining noisy data: A prediction quality perspective*. The University of Iowa, 2005.

[95] E. A. Silver. An overview of heuristic solution methods. *Journal of the operational research society*, 55(9):936–956, 2004.

[96] J. O. Smith. *Introduction to digital filters: with audio applications*, volume 2. Julius Smith, 2007.

[97] A. S. Soofi and L. Cao. *Modelling and forecasting financial data: techniques of nonlinear dynamics*, volume 2. Springer Science & Business Media, 2012.

[98] M. Souden, S. Araki, K. Kinoshita, T. Nakatani, and H. Sawada. A multichannel mmse-based framework for speech source separation and noise reduction. *IEEE Transactions on Audio, Speech, and Language Processing*, 21(9):1913–1928, 2013.

[99] A. Souloumiac. Blind source detection and separation using second order nonstationarity. In *1995 International Conference on Acoustics, Speech, and Signal Processing*, volume 3, pages 1912–1915. IEEE, 1995.

[100] L. Steinberg. The backboard wiring problem: A placement algorithm. *Siam Review*, 3(1):37–50, 1961.

[101] R. Storn and K. Price. Differential evolution–a simple and efficient heuristic for global optimization over continuous spaces. *Journal of global optimization*, 11(4):341–359, 1997.

[102] M. Strumik, W. M. Macek, and S. Redaelli. Discriminating additive from dynamical noise for chaotic time series. *Physical Review E*, 72(3):036219, 2005.

[103] K. L. Teo, C. Goh, and K. Wong. *A unified computational approach to optimal control problems*. 1991.

[104] G. Ungerboeck. Channel coding with multilevel/phase signals. *IEEE transactions on Information Theory*, 28(1):55–67, 1982.

[105] D. Ustebay, B. N. Oreshkin, M. J. Coates, and M. G. Rabbat. Greedy gossip with eavesdropping. *IEEE Transactions on Signal Processing*, 58(7):3765–3776, 2010.

[106] M. Vacher, B. Lecouteux, and F. Portet. Recognition of voice commands by multisource asr and noise cancellation in a smart home environment. In *2012 Proceedings of the 20th European Signal Processing Conference (EUSIPCO)*, pages 1663–1667. IEEE, 2012.

[107] Q. Wang, S. Guo, and K.-F. C. Yiu. Distributed acoustic beamforming with blockchain protection. *IEEE Transactions on Industrial Informatics*, 2020.

[108] W. Weber. Differential encoding for multiple amplitude and phase shift keying systems. *IEEE Transactions on Communications*, 26(3):385–391, 1978.

[109] X. Wu, B. Liu, L. Zhang, Y. Mao, X. Xu, J. Ren, Y. Zhang, L. Jiang, and X. Xin. Probabilistic shaping design based on reduced-exponentiation subset indexing and honeycomb-structured constellation optimization for 5g fronthaul network. *IEEE Access*, 7:141395–141403, 2019.

[110] L. Xiao and S. Boyd. Fast linear iterations for distributed averaging. *Systems & Control Letters*, 53(1):65–78, 2004.

[111] F. Xiong. *Digital Modulation Techniques, (Artech House Telecommunications Library)*. Artech House, Inc., 2006.

[112] K. F. C. Yiu, N. Grbic, K.-L. Teo, and S. Nordholm. A new design method for broadband microphone arrays for speech input in automobiles. *IEEE Signal Processing Letters*, 9(7):222–224, 2002.

[113] K. F. C. Yiu, Y. Liu, and K. L. Teo. A hybrid descent method for global optimization. *Journal of Global Optimization*, 28(2):229–238, 2004.

[114] K. F. C. Yiu, X. Yang, S. Nordholm, and K. L. Teo. Near-field broadband beamformer design via multidimensional semi-infinite-linear programming techniques. *IEEE Transactions on Speech and Audio processing*, 11(6):725–732, 2003.

[115] X. Yu, D. Hu, and J. Xu. *Blind source separation: theory and applications*. John Wiley & Sons, 2013.

[116] K. Zeger and A. Gersho. Pseudo-gray coding. *IEEE Transactions on communications*, 38(12):2147–2158, 1990.

[117] E. Zehavi. 8-psk trellis codes for a rayleigh channel. *IEEE Transactions on Communications*, 40(5):873–884, 1992.

[118] Y. Zeng and R. C. Hendriks. Distributed estimation of the inverse of the correlation matrix for privacy preserving beamforming. *Signal Processing*, 107:109–122, 2015.

[119] Y. Zeng, R. C. Hendriks, and R. Heusdens. Clique-based distributed beamforming for speech enhancement in wireless sensor networks. In *21st European Signal Processing Conference (EUSIPCO 2013)*, pages 1–5. IEEE, 2013.

[120] J. Zhang, S. P. Chepuri, R. C. Hendriks, and R. Heusdens. Microphone subset selection for mvdr beamformer based noise reduction. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 26(3):550–563, 2017.

[121] H. Zheng and J. Boyce. An improved udp protocol for video transmission over internet-to-wireless networks. *IEEE Transactions on Multimedia*, 3(3):356–365, 2001.