# USC-SIPI REPORT #366

## Three-Dimensional Detection and Equalization for Page-Oriented Optical Data Storage Systems

by

**Tawei Ho**

**August 2005**

Signal and Image Processing Institute
**UNIVERSITY OF SOUTHERN CALIFORNIA**
Viterbi School of Engineering
Department of Electrical Engineering-Systems
3740 McClintock Avenue, Suite 400
Los Angeles, CA 90089-2564 U.S.A.

# Acknowledgements

First of all, I would like to thank Prof. Alexander A. Sawchuk for being my dissertation advisor. I might not be able to finish this if I were not under his professional guidance. I have learned not only from his academic expertise, but also from his lifelong knowledge and discipline. His gentleness is a reflection of his great wisdom. These invaluable experiences have profoundly affected my life, which I didn't expect in the beginning of my Ph. D career.

I would like to extend my grateful thanks to Prof. B. Keith Jenkins and Prof. Roger Zimmermann for serving on both my dissertation and qualifying examination committee. I have made several essential improvements in this research based on their pinpoint suggestions and advice. I owe a debt of thanks to Prof. Charles L. Weber and Prof. Wlodek Proskurowski, who are on my qualifying examination committee. Their extensive knowledge guides me through the early stage of dissertation.

I also appreciate the friendship from my colleagues: Nopparit Intharasombat, Zahir Y. Alpaslan, Chiao Wang and Yunsong Huang. I gratefully thank Gloria Halfacre, Gerrielyn Ramos, and Diane Demetras for their administrative help.

Finally, I would express my sincere gratitude to my significant other, Linda Chu for her endless love and support. Because of that, I have the source to keep working. Her accurate comments help me to achieve, both academically and socially.

# Table of Contents

# List of Tables

# List of Figures

# Abstract

In this thesis, we describe the effects of both inter-page interference (IPI) and inter-symbol interference (ISI) in volumetric (3D) page-oriented optical data storage (PODS) systems, and present a detailed mathematical model for 3D PODS systems. Based on this model, we describe the equalization and detection techniques that overcome both IPI and ISI in the readout system. This detection scheme is extremely useful for future high data capacity page-oriented storage systems, where the IPI effect is severe. For different application requirements, our proposed methods can be operated flexibly in both *batch* and *recursive* modes. In our simulation results, our detection method shows consistent improvements over conventional methods.

In addition, we propose multi-level data signaling on 3D PODS to enhance the data density further. Our detection algorithm also yields significant improvements on multi-level 3D PODS systems. Therefore, it is advantageous to apply our detection method for increasing the storage density in future storage systems. We also propose a variable thresholding detection method with improved performance. It overcomes the ISI and IPI effects simultaneously by using knowledge of the interference from surrounding neighbors and changing the threshold locally. Hence, it provides a low complexity detection method for the readout system.

To further reduce the complexity, we examine the implementation of least square equalization and variable threshold methods using finite precision arithmetic. In a

bit-error-rate comparison, we show that the fixed-point variable threshold method achieves similar results as the floating-point version but only using one seventh of the bits. Therefore, a potential saving of the storage and complexity is achieved by using finite precision arithmetic.

Last, we investigate 3D PODS systems with pixel misregistration. Using our least square equalization and variable threshold methods, we have significant improvement when the amount of misalignment is 10% of the pixel pitch or less. These simulation results show the robustness and feasibility of our detection methods.

# Chapter 1 Introduction

In this chapter, we discuss the limitations of current data storage systems and the objective of the work in this thesis. A summary of the contributions of this thesis is also provided here.

## 1.1 Motivation and Objective

In modern multimedia applications, the amount of data generated grows dramatically. Since this enormous amount of data needs to be stored and transmitted, there are many challenges for both storage and transmission media. Conventional two dimensional (planar) data storage systems, such as magnetic hard drive, compact disc (CD), digital versatile disc (DVD), and blu-ray disc (BD), manage to satisfy this requirement. However, the evidence shows that the storage densities of these planar data storage systems (particularly magnetic storage) are approaching the theoretical limit [31]. As demands on data storage systems keep increasing, volumetric (3D) page-oriented optical data storage (PODS) systems offer potentially larger capacity, faster access time, and higher data transfer rates. In general, the 3D PODS can provide extremely high data transfer rate (over 1 Gbits/sec) and massive data storage capacity (over 1 Tbits/cm3) [14][44]. Therefore, 3D PODS systems are a promising candidate for next generation data storage systems.

One advantage of the 3D PODS over conventional planar data storage systems is the high data transfer rate that is achieved by the parallel readout mechanism. In the 3D

PODS, the data is recorded/read out as a sequence of 2D data pages as compared to the one dimensional sequential mechanism in planar systems. Although researchers are working on mutli-channel readout for conventional data storage systems (such as the 9 track DVD-ROM system developed by Philips [9]), the number of readout channels is still much less than that of the 3D PODS.

Another advantage of 3D PODS over planar data storage systems is the data storage capacity. Because the third dimension is utilized, 3D PODS can maximize the data packing density within a storage volume. Therefore, 3D PODS can also be treated as a multi-layered extension of conventional planar storage systems. As the bit density of multi-layered data storage increases, both inter-symbol interference (ISI) and inter-page interference (IPI) increase as well. To maintain the reliability of 3D PODS, it is a challenging issue to mitigate the ISI and IPI effects.

The goal of this research is to overcome the effects of ISI and IPI in 3D PODS systems. To achieve this goal, we develop a detailed optical and electronic model of 3D PODS systems. With the model, we can determine effects of ISI and IPI. Finally, we apply 3D estimation and detection based on least square equalization and variable threshold methods to overcome effects of ISI and IPI simultaneously. We analyze the performance of our proposed methods in terms of BER and other system parameters. For a complete analysis, we consider systems using rectangular coordinates as well as hexagonal coordinates.

To further increase the system capacity, we investigate multi-level 3D PODS systems and describe how our methods produce satisfactory results in the presence of ISI and IPI. To reduce overall system complexity, we apply finite precision arithmetic in our detection methods, and demonstrate that they have performance comparable to the floating point algorithm.

In actual optical data storage systems, mechanical misalignment error is an influential parameter in system performance. We describe methods to overcome this problem using digital signal processing and an extension of an algorithms.

### 1.2 Organization

The organization of this thesis is shown in Figure 1.1. The relevant background knowledge is introduced in Chapter 2. Here we present a concise review of general data storage systems. Several important related works are also discussed in this chapter.

In Chapter 3, page-oriented optical data storage (PODS) systems are introduced. The readout subsystem of PODS is of special importance and analyzed in detail. In addition, the three-dimensional page-oriented data storage (3D PODS) system model is examined. We focus on 3D PODS systems that use two-photon absorption technology and an incoherent imaging recording and readout model. In particular, we scrutinize the effect of focusing error on the readout optics. This forms the basis for inter-page interference analysis. The readout light intensity distribution is also

included in our system model for completeness. All these analyses are derived from mathematical models and they provide the foundation for the detection algorithms in the remaining chapters.

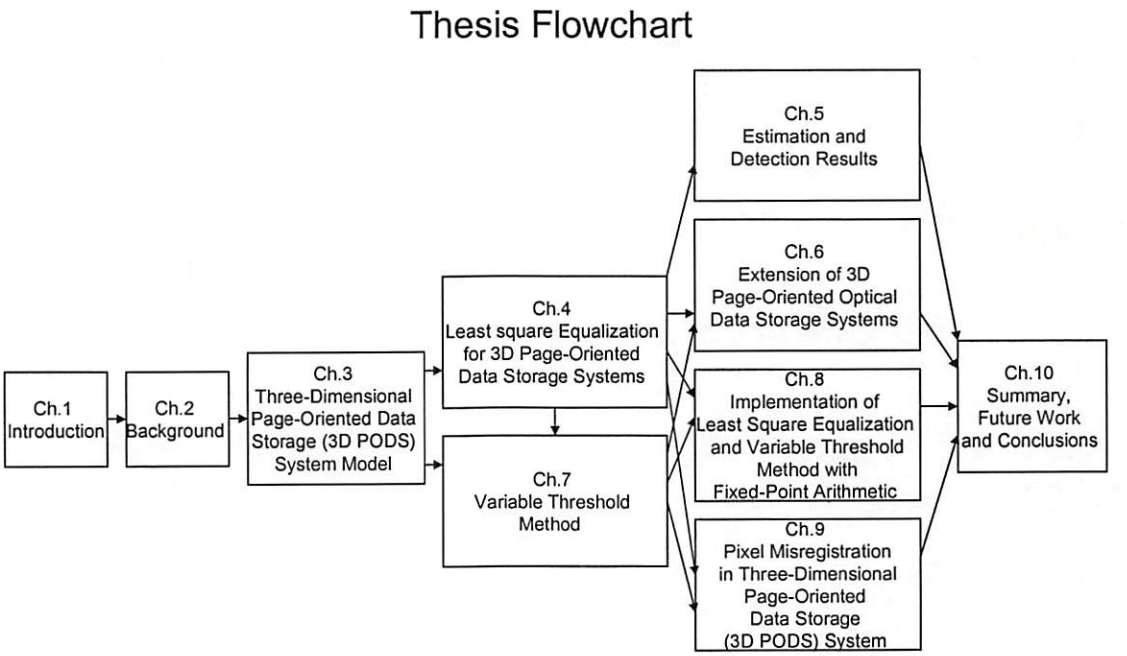



Figure 1.1. Thesis organization.

The least square equalization algorithm for 3D PODS is introduced in Chapter 4. There are two modes of operation associated with it, i.e. the *Batch* mode and the *Recursive* mode. Both of these two modes are described in detail. In addition, the time complexity of the algorithm is also investigated in this chapter.

In Chapter 5, the estimation and detection results are presented. The simulation parameters from actual experimental systems are listed for reference. The simulations are executed under different situations and the results are shown as a

function of different variables. In these simulations, our detection algorithm shows consistently better results than conventional algorithms.

Some extensions of 3D PODS are discussed in Chapter 6. We explore multi-level data storage techniques for 3D PODS. This method potentially enhances the data packing density. We compare two different methods to achieve multi-level data storage and give simulation results.

In Chapter 7, we propose a novel detection algorithm called the variable threshold method. By evaluating the interference from surrounding neighbors, this method varies the threshold for each pixel. It is an extension of the conventional fixed threshold method and can reduce the effects of both inter-symbol interference and inter-page interference simultaneously.

The least square equalization and variable threshold methods using finite precision arithmetic are discussed in Chapter 8. From the simulation, the fixed-point arithmetic achieves comparable performance to the floating-point arithmetic. Therefore, our methods are capable of operating with low complexity and limited local processor storage.

In Chapter 9, we consider 3D PODS systems with pixel misregistration problems. Using the least square equalization and variable threshold methods, the errors resulting from misalignment are reduced significantly when the misregistration is 10% or less of the detector pixel pitch.

Conclusions are given in Chapter 10. We also discuss several possible future studies including multi-wavelength readout, which adds another dimension of freedom to increase the data capacity.

### 1.3 Contribution

The contributions of this thesis to general optical data storage systems are summarized in the following topics.

(1) Mathematical Models for Three-Dimensional Page-Oriented Data Storage Systems. We investigate inter-page interference in 3D PODS and formulate the mathematical model. This model can be applied to either coherent or incoherent data storage systems. We derive this model to have a better understanding of inter-page interference effects.

(2) Least Square Equalization Detection for Three-Dimensional Page-Oriented Data Storage Systems[15][18]. Based on our 3D PODS model, we propose the least square equalization algorithm. In our simulation results, it achieves significant improvement over the conventional thresholding method. It has two modes of operation and is suitable for different readout formats and complexity requirements.

(3) Multi-Level Data Storage System for 3D PODS. To further increase the data packing density, we employ the third spatial dimension in 3D PODS systems and

analyze possible ways to achieve multi-level data storage. The simulation results suggest that one method is preferable over the other under certain situations.

(4) Variable Threshold Detection for Three-Dimensional Page-Oriented Data Storage Systems. By using the knowledge of the readout point-spread function (PSF), we compute the interference from surrounding neighbors and modify the threshold at each pixel. It shows superior results than the conventional fixed threshold method.

(5) Implementation of Least Square Equalization and Variable Threshold Method with Fixed-Point Arithmetic. In order to reduce the complexity, we develop methods for performing the signal processing using finite precision arithmetic. The simulation results suggest that potential saving of bits is attainable with only minor effects on the system performance.

(6) Improvement of 3D PODS System with Pixel Misregistration. The detected signals are degraded when mechanical misalignment occurs. With small amounts of misregistration, our method can provide a performance improvement. It also shows the robustness of the least square equalization and variable threshold methods in the presence of pixel misregistration phenomena.

(7) Complete and Detailed Simulations Based on 3D PODS System with Rectangular and Hexagonal Coordinates. In the 3D PODS system, the coordinate system is used to describe the distribution of data pixels. Two most common choices are rectangular and hexagonal coordinates. We perform complete simulations of our

algorithms on 3D PODS systems using both these coordinates and discuss their relative advantages and disadvantages.

# Chapter 2 Background

In this chapter, a brief introduction of general data storage systems is presented. It provides the background knowledge for this thesis. In addition, several important works related to page-oriented data storage (PODS) systems are also discussed.

## 2.1 Data Storage Systems

There are three major parts to the general data storage system; namely the channel en/decoder, the modulation en/decoder, and the recording/readout subsystem. The following figure shows these three components and the data flow.

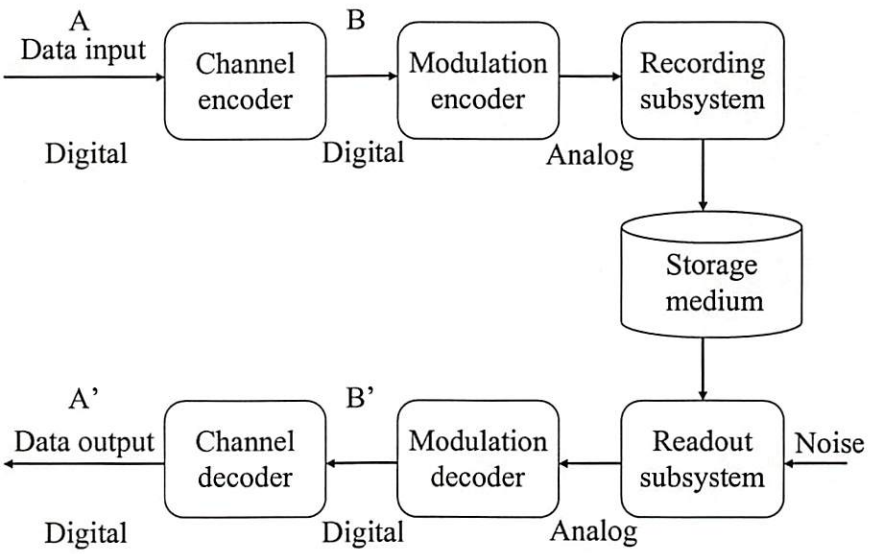



Figure 2.1. Block diagram of the general data storage system.

The above block diagram describes all kinds of digital data storage systems in which information is stored and retrieved in a digital format. Such systems include

magnetic tape, magnetic floppy and hard drives, and optical systems such as CD-ROM, DVD, etc. The upper half of the diagram is the data recording part. The raw digital data is input sequentially into the channel encoder, which inserts extra bits into the raw data stream. These redundant bits are used for error detection and error correction when retrieving the data. Therefore, the channel encoder gives the ability to detect and correct some errors that occur elsewhere in the system.

The recording/readout subsystem provides the analog input and output to/from the analog storage medium. Noise (measurement error) is an inherent part of the recording and readout process. Hence, the modulation encoder is applied to process the data to match the physical characteristics of the recording/readout subsystem. To maximize the storage capacity, the physical bit packing density of the data storage system is generally made as large as possible. In this situation, the inter-symbol interference (ISI) becomes a limiting factor for the reliability of data retrieving. To compensate for ISI, the modulation encoder maps the input digital data with algorithms that attempt to minimize the effective ISI. With proper design, the modulation codes can improve system robustness without compromising too much storage capacity.

The recording subsystem writes the digital data from the modulation encoder output onto the storage medium in analog form. Therefore, it acts as a digital-to-analog converter. There are several types of storage media, such as magnetic, mechanical and optical media [45]. The data might be recorded onto a 1D, 2D or 3D medium in a

serial stream (1 dimension) or in parallel with several streams. The data retrieving process is in the bottom part of the block diagram. The stored data is retrieved through the readout subsystem and transformed into digital form. Hence, the function of the readout subsystem is just opposite to the recording subsystem, and as an analog-to-digital converter. Besides the data format conversion, the readout subsystem must reliably recover the data in the presence of noise and ISI. A good readout subsystem will give reliable output with moderate complexity. The output from readout subsystem goes into the modulation decoder. The decoder uses the same mapping table as the encoder side with opposite direction of mapping. Hence, it converts the data back to its original form with minimum error.

The data input to the channel decoder typically has a bit-error-rate (BER) of $10^{-3}$ to $10^{-4}$ and the channel decoder further reduces the BER to $10^{-12}$ or better by correcting errors. There are many error control coding algorithms that have this capability [24]. Some of them use off-the-shelf chips, and implement codes such as the Reed-Solomon code, that is widely used in many data storage systems. These error control coding algorithms can decrease the BER from $10^{-3}$ to $10^{-20}$. Some examples are shown in Figure 2.2 [37]. This figure shows the output BER (vertical axis) as a function of the input BER (horizontal axis) for Reed-Solomon codes with different parameters.

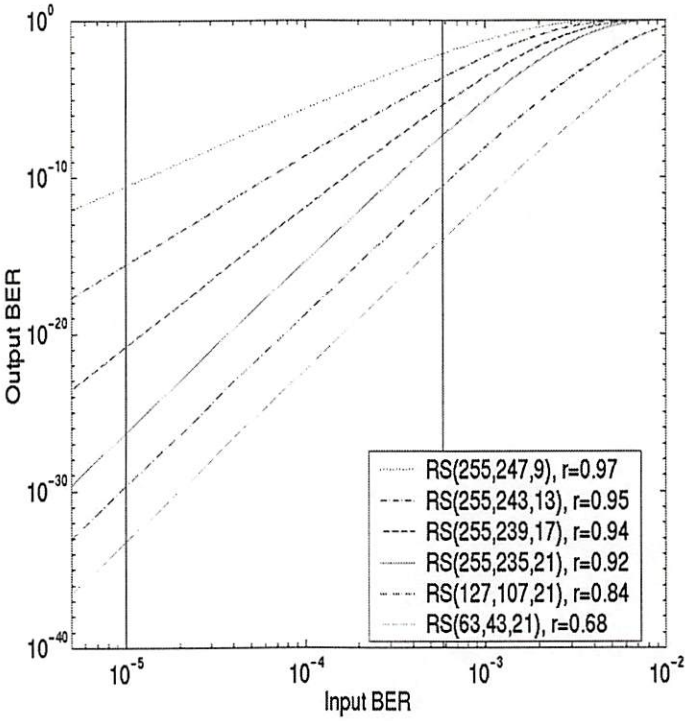



Figure 2.2. Capabilities of Reed-Solomon code.

The BER between the channel encoder and decoder is the actual BER that the end user will encounter. This error rate after the channel en/decoder is called residual error rate while the raw error rate refers to the error rate from the storage medium.

In addition to the BER, the transfer rate is another important parameter for any data storage system. Both the recording and retrieving transfer rates are usually specified. Depending on different applications, the data transfer rate must satisfy the application requirements. For example, real-time applications need to have a constant average transfer rate. To increase the data transfer rate, parallel readout is preferable over sequential readout. In a parallel readout scheme, the data is recorded or retrieved

through more than one channel. As an example, Philips has described the design of an experimental DVD-ROM drive with parallel readout of up to 9 tracks simultaneously [9]. On the other hand, page-oriented data storage (PODS) systems use two dimensional data pages for parallel recording/retrieval, therefore, they can dramatically increase the data transfer rate.

Another important characteristic of data storage systems is the data capacity. Since the storage system includes modulation coding and channel coding, the data capacity refers to the net capacity which can be used by the user. These coding algorithms will put some overhead or redundancy on the data and increase the overall data volume. The code rate is defined as the ratio of the useful data to the overall data volume after applying the coding algorithms. Therefore, the net capacity of data storage system depends on the code rates of the modulation code and channel code.

With advances in technology, the basic physical storage capacity has been increasing. To measure it, one often uses areal density, also called bit density, which is the amount of data that can be stored in a given area of a two dimensional disk. It is usually expressed in bits per square inch (BPSI). New data storage systems have been doubling this value every nine months [41]. In the magnetic data storage, this density almost approaches the theoretical limit [31]. For this reason, two-photon PODS and holographic data storage systems are attractive techniques for achieving higher data densities, because they potentially utilize all three dimensions (the volume) of the available recording material for storage.

## 2.2 Other Related Work

There are several different technologies used in page oriented data storage (PODS) systems, including holographic storage systems [14][16][23][33][31][31][34], spectral hole burning storage systems [8][22][25][35][38], and two-photon data storage systems [10][17][26][29][44]. The advantage of page oriented data storage (PODS) systems over traditional data storage systems is the utilization of three dimensional volumetric media with parallel recording/retrieving channels. Therefore, it potentially increases the data capacity but also increases the data transfer rate. However, as the data packing density increases, the PODS system also suffer from noise, inter-symbol interference (ISI), and inter-page interference (IPI).

The simple binary threshold mechanism is typically used in PODS systems as shown in the previous section. Some researchers have proposed other detection and equalization techniques to alleviate those unwanted effects. For 2D ISI and noise effects, a linear equalizer structure was suggested to approximate an inverse filter to the ISI characteristics [5][6]. It makes use of the low pass nature of the 2D ISI channel and eliminates it with inverse filtering. Another linear minimum mean-squared-error equalization method is introduced for the holographic data storage system [19][42]. It uses equalization followed by the threshold decision to retrieve the data within one data page. In the experiment results, they show that the error rate is higher on the periphery than that on the center of the data page. In stead of doing equalization on the data retrieving side, a recording side inverse filtering is proposed

[2]. It exploits a *priori* knowledge of the 2D ISI and pre-distorts the data before recording. Meanwhile, other researchers focus on modulation coding schemes to improve the BER performance of holographic data storage system [3][13][21][43].

There are still others engaged in enhancing other parameters of the PODS. For increasing the capacity, a multilevel two dimensional optical storage system is present, which uses different pit sizes within the hexagonal coordinate [11].

Although there are many studies on mitigating 2D ISI effects, none of them considers 3D IPI. Since the demand for capacity keeps increasing, 3D IPI will be an important factor when packing data closely into three dimensions. Therefore, this work addresses the ISI and IPI issues simultaneously. This gives the readout subsystem the ability to handle the applications with extremely high packing density. In addition, we also consider the possibility of multilevel signaling in PODS. This can be considered as another degree of freedom that may further improve the bit density and storage capacity in the future.

# Chapter 3 Three-Dimensional Page-Oriented Data Storage (3D PODS) System Model

We describe a detailed 3D PODS system model in this chapter. In this novel model, we include both the inter-symbol interference (ISI) and inter-page interference (IPI) effects. A special emphasis is on the readout system of 3D PODS. Based on its system configuration, we derived the three dimensional incoherent point spread function (PSF) for the readout system model. It contains the 2D PSF with and without focusing error, which are expressed mathematically according to optics theory.

## 3.1 Page-Oriented Optical Data Storage (PODS) Systems

Page-Oriented Optical Data Storage (PODS) Systems consists of two parts, the recording subsystem and the readout subsystem. Figure 3.1 shows the recording subsystem.

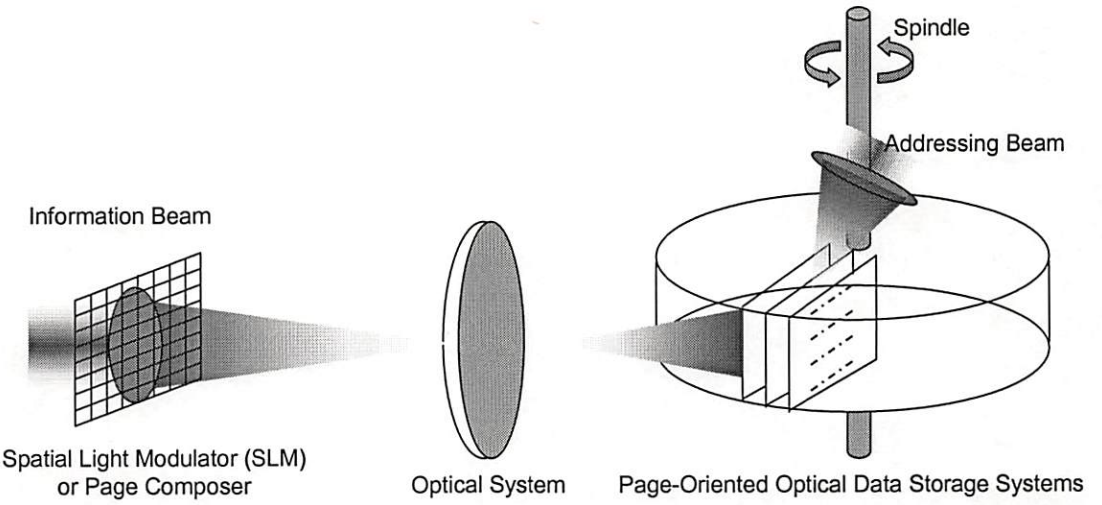



Figure 3.1. Recording subsystem of PODS.

In the recording subsystem, a spatial light modulator (SLM) or so-called page composer is used to convert electrical data signals into spatially modulated coherent or incoherent optical signals [12]. Essentially, the SLM modulates the spatial light distribution of absorption or phase shift according to the electrical driving signal that represents the data. Because of the fast temporal response, SLM devices are widely used in optical signal processing field. After processing with the SLM, the desired image pattern is written onto a photo-sensitive material which records the pattern through the optical lens system. The recorded data forms a two dimensional array of data marks called data pages on these photo-sensitive materials. These data pages are stored sequentially on a rotating disk for access in the PODS system. During the recording process, only the data page to be stored is excited by the addressing beam [10][26].

The readout subsystem is shown in Figure 3.2.

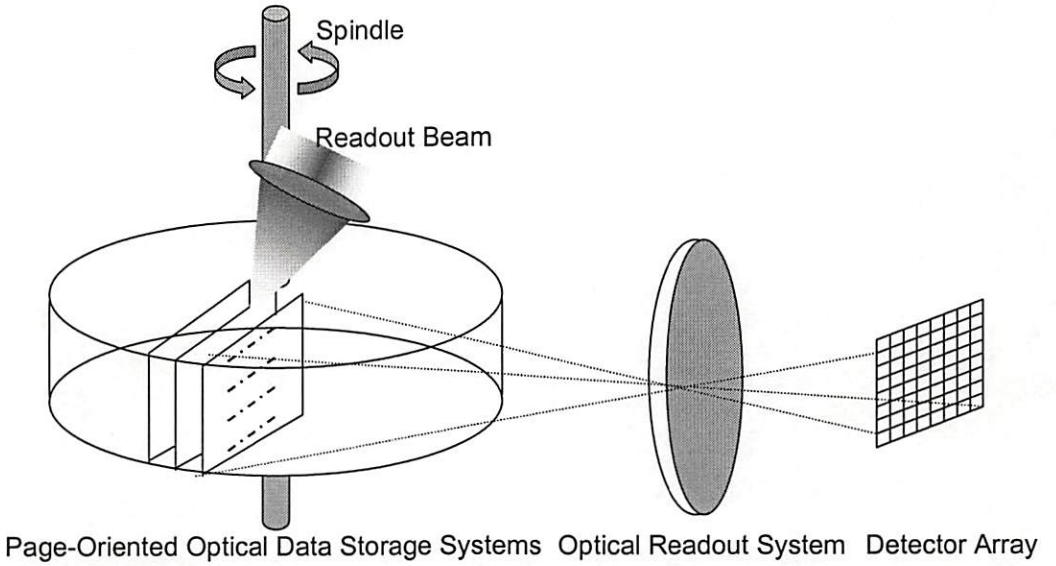



Figure 3.2. Readout subsystem of PODS.

Once the data is stored in the 3D PODS, the readout subsystem retrieves the data. The readout subsystem consists of an optical readout lens and the detector array. To retrieve the data, the data page is accessed from the rotating disk and illuminated by the readout beam. For systems using two-photon materials and technology [26], this illuminated data page generates a fluorescence pattern that is imaged by the optical readout lens onto the two dimensional detector arrays. Since the non-illuminated data pages are effectively transparent, only the fluorescent mark pixels (1 bit) on the illuminated data page are sensed by the detector cells. The data page and the 2D detector array are aligned such that each detector cell corresponds to an individual data mark. Hence, the data is retrieved in parallel with high throughput.

### 3.2 New Readout System Model

To analyze the readout subsystem, we represent it by the block diagram of Figure 3.3. For simplicity, we assume that the inter-page interference is due to one neighbor page on both sides of a given page being read. If the inter-page interference is severe, i.e. more pages interfere with the data page in focus, we can include them in a similar way. Hence, the 3D mathematical model of the readout subsystem is shown in Figure 3.3.

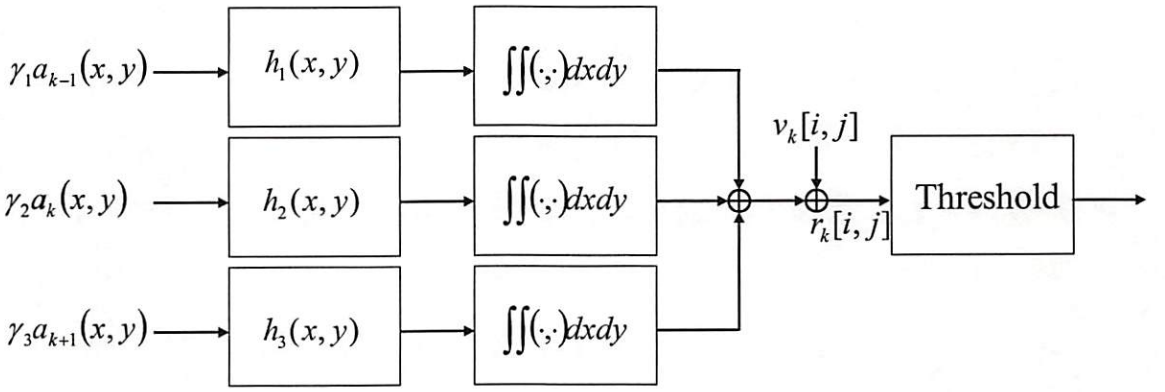



Figure 3.3. The 3D mathematical model for the readout subsystem of PODS.

In the above mathematical model, $(x,y)$ denotes the 2D continuous signal whereas $[i,j]$ denotes the 2D discrete signal, i.e. $x$ and $y$ are continuous values and $i$ and $j$ are discrete values. Here $a_k(x,y)$ represents the data on the data page that is in focus. The interference terms are shown as $a_{k-1}(x,y)$ and $a_{k+1}(x,y)$, which represent the data page in front and in the back of the page in focus, respectively. The weights $\gamma_1, \gamma_2$, and $\gamma_3$ denote the relative light intensity from the corresponding data pages.

These weights are nonnegative and are normalized such that $\gamma_1 + \gamma_2 + \gamma_3 = 1$. Here $h_1(x, y)$, $h_2(x, y)$ and $h_3(x, y)$ are the point spread functions (PSFs) corresponding to different data pages in the readout subsystem.

These point spread functions are derived in Sections 3.5.1 and 3.5.2, and are functions of the object distance in front of the lens [12]. For the same object distance, we can use the same point spread function to represent it. Since we assume equal spacing between data pages in PODS, we use these three point spread functions for the entire readout process. At the detector plane, each detector collects (integrates) the incoming light and generates corresponding intensity levels.

While we assume an incoherent system here, such as two-photon system, we can easily apply it to coherent systems (holographic systems etc.) as well. The ideal point spread function of the *diffraction-limited* image system is modeled mathematically in Sections 3.5.1. When there is a focusing error, the point spread function will vary correspondingly. It is mathematically equivalent to convolving an out-of-focus term with the *diffraction-limited* point spread function. The details of the out-of-focus PSF are discussed in Section 3.4. For simplicity, we assume that the magnification factor of the optical readout system is one. Since the out-of-focus distance is relatively small compared to the focal distance, it is a reasonable assumption. In addition, when the focusing error is large, the inter-page interference becomes negligible and will be ignored.

In the readout system, the detector collects the incident photons and produces a signal proportional to the intensity and the detector area. For an incoherent system, we use integration to model it mathematically. The integration area is the active field of the detector. Assuming a detector pixel having unit $x$ and $y$ dimensions, the detector integration in one dimension extends from $-\delta/2$ to $\delta/2$, where $\delta$ is a linear fill factor of a detector ranging from 0 to 1 [28]. The fill factor defines the light-gathering area of a detector. For instance, a detector with fill factor of 0.8 has an imaging array in which 20% of each pixel's area is insensitive to incident light. The detector also serves as a spatial sampling device, hence, its output signal with discrete index values is

$$\begin{aligned} r_k[i,j] = & \int_{j-\delta/2}^{j+\delta/2}\int_{-\delta/2}^{i+\delta/2}\gamma_1 a_{k-1}(x,y)\otimes h_1(x,y)dxdy + \int_{j-\delta/2}^{j+\delta/2}\int_{-\delta/2}^{i+\delta/2}\gamma_2 a_k(x,y)\otimes h_2(x,y)dxdy \\ & + \int_{j-\delta/2}^{j+\delta/2}\int_{-\delta/2}^{i+\delta/2}\gamma_3 a_{k+1}(x,y)\otimes h_3(x,y)dxdy + v_k[i,j] \end{aligned} \quad ,(3.1)$$

where $\delta$ is the linear fill factor, $\otimes$ denotes the 2D convolution operator and $v_k[i,j]$ is 2D additive noise. Because the integration and convolution are linear operators, we can further combine the terms in the received signal once we have the linear relationship between those point-spread functions. In reality, the data recorded on the storage medium is in analog form. Ideally, the recorded data mark $a_k(x,y)$ is a delta function with a weight of either zero or one. If the size of the data mark is small compared to the extent of the point spread function, we can model the data marks as

delta functions, and combine them with the point-spread functions to carry out the 2D discrete convolution in the above equation giving

$$\begin{aligned} r_k[i,j] = \gamma_1 \sum_{m=-M}^{M} \sum_{n=-N}^{N} a_{k-1}[i-m, j-n] \int_{n-\delta/2}^{n+\delta/2} \int_{m-\delta/2}^{m+\delta/2} h_1(x,y)\,dxdy \\ + \gamma_2 \sum_{m=-M}^{M} \sum_{n=-N}^{N} a_k[i-m, j-n] \int_{n-\delta/2}^{n+\delta/2} \int_{m-\delta/2}^{m+\delta/2} h_2(x,y)\,dxdy \\ + \gamma_3 \sum_{m=-M}^{M} \sum_{n=-N}^{N} a_{k+1}[i-m, j-n] \int_{n-\delta/2}^{n+\delta/2} \int_{m-\delta/2}^{m+\delta/2} h_3(x,y)\,dxdy + v_k[i,j] \end{aligned} \quad , \quad (3.2)$$

where $a_{k-1}[i,j]$, $a_k[i,j]$ and $a_{k+1}[i,j]$ are binary-valued discrete index versions of their counterparts and $M$ and $N$ are the horizontal and vertical extent of the point spread function.

The two dimensional noise in the readout system is represented as $v_k[i,j]$. This noise represents photon (Poisson) noise, electronic (thermal) noise and other noise sources such as background light. In this study, we use additive white Gaussian noise (AWGN) to model them. More accurate and detailed noise models can also be used for particular types of storage systems [40].

The simplest readout detection system uses a binary threshold decision mechanism. The threshold value $T$ is pre-selected and fixed for the whole readout process. Given the detector output signal $r_k[i,j]$, the output of the threshold device is

$$\hat{a}_k[i,j] = \begin{cases} 0, & if\ r_k[i,j] < T \\ 1, & if\ r_k[i,j] > T \end{cases}. \quad (3.3)$$

We want the output of the readout subsystem $\hat{a}_k[i,j]$ to be as close to the data stored on the medium $a_k[i,j]$ as possible. The discrepancy between them is the error. For the data stored in a binary format, this error is calculated as bit error. Therefore, the performance of the readout subsystem can be judged by the bit error rate (BER) it generates. We propose a novel readout subsystem to combat both the inter-symbol interference and inter-page interference simultaneously and provide a significant BER improvement over traditional readout systems.

### 3.3 Optical System Configuration

Our three-dimensional page-oriented data storage (3D PODS) system model is shown in Figure 3.4. The 2D pages of stored binary data are accessed sequentially from a rotating disk by means of a narrow readout beam that ideally illuminates only one page (Data Page 2) as shown. The fluorescence pattern from Data Page 2 located in the object plane at $z_o$ in front of the lens is imaged to a diffraction-limited spot array at the in-focus image plane located at $z_i$ behind the lens. Here a 2D detector array measures the output for further processing by a modulation decoder.

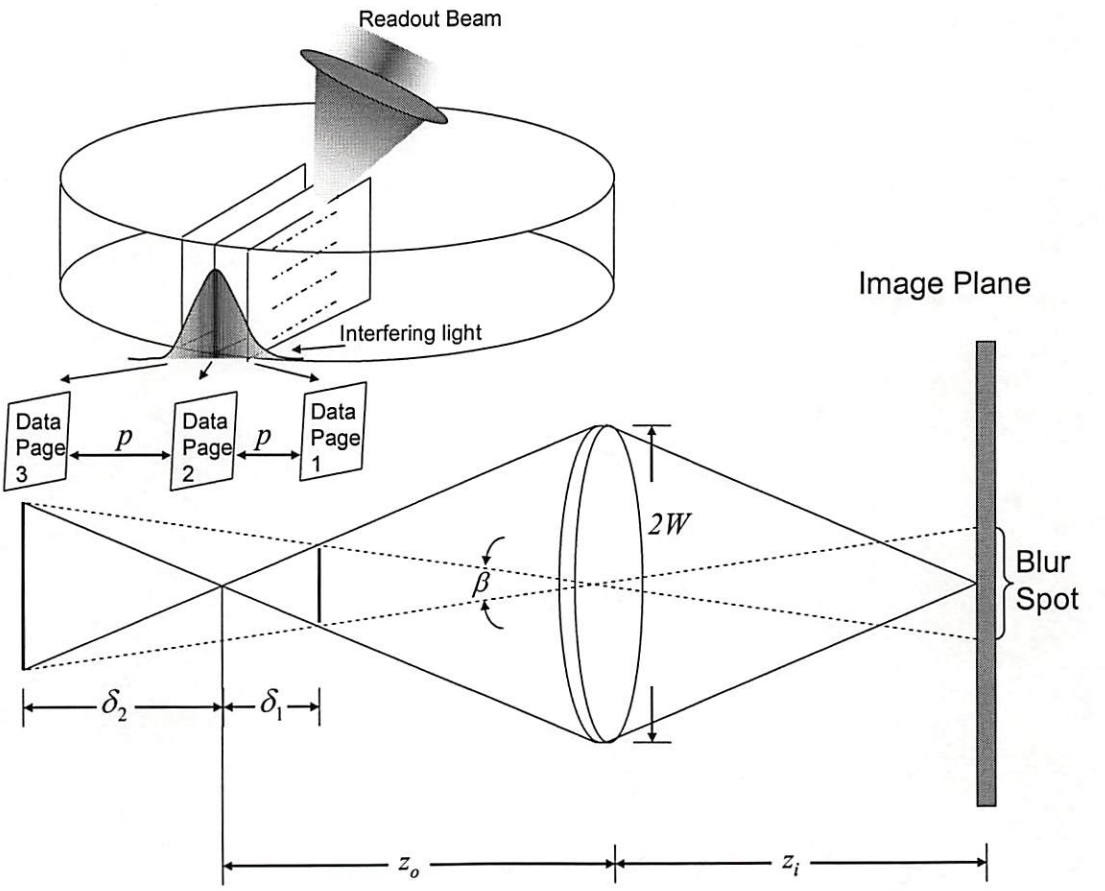



Figure 3.4. Optical system configuration for 3D PODS.

If an object plane is placed at a distance $z_a \neq z_o$, the image of a point object is out of focus. If $z_a > z_o$, we write $z_a = z_o + \delta_2$, while if $z_a < z_o$, we write $z_a = z_o - \delta_1$. Note that $p$ is the fixed inter-page spacing in the page-oriented optical memory. We note that a point object in an out of focus plane produces a blur spot of diameter $d$ at the image plane. From geometrical optics [39], we have the system setup shown in Figure 3.5.

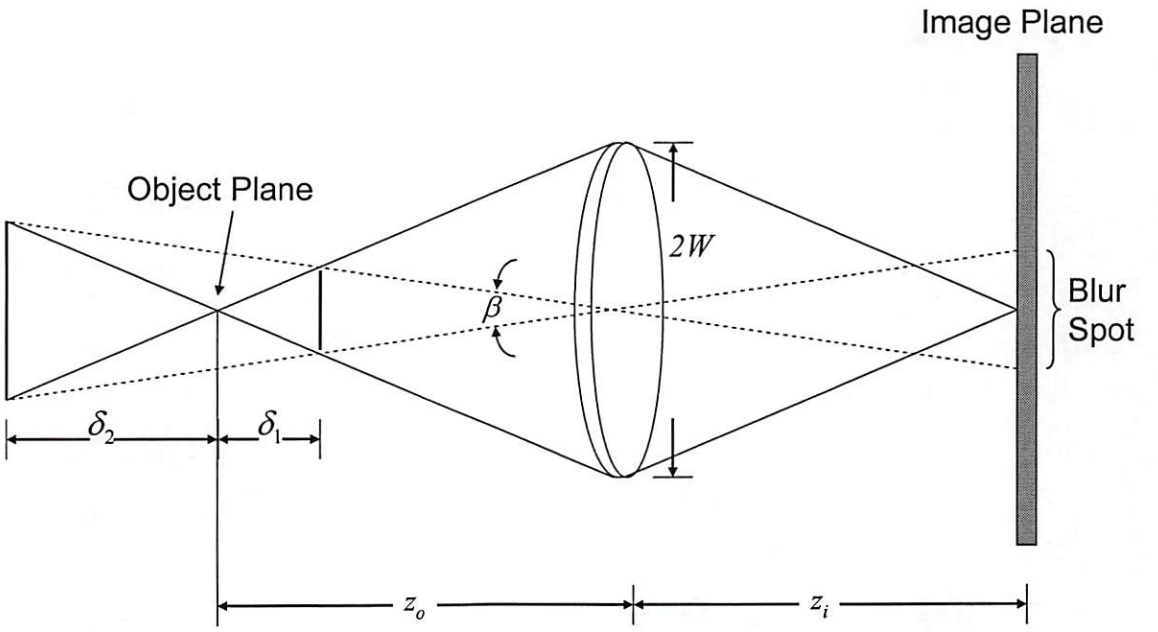



Figure 3.5. Optical system setup for determining blur spots in the image plane.

where $W$ is the lens diameter

$\beta$ is the angle subtended by the blur spot

$f$ is the focal length of the lens

The planes located at distance $\delta_1$ and $\delta_2$ in front and in back of the focus point respectively are chosen to produce the same size of blur spot on the image plane. The distances $z_i$ and $z_o$ satisfy the lens law, i.e.

$$\frac{1}{z_i}+\frac{1}{z_o}=\frac{1}{f} \tag{3.4}$$

and from geometrical relationships we have

$$\frac{\delta_2}{(z_o+\delta_2)\beta}=\frac{\delta_1}{(z_o-\delta_1)\beta}=\frac{z_o}{2W} \tag{3.5}$$

which we solve for

$$\delta_1 = \frac{z_o^2 \beta}{2W + z_o \beta} \tag{3.6}$$

and

$$\delta_2 = \frac{z_o^2 \beta}{2W - z_o \beta} \tag{3.7}$$

Hence, $\delta_1$ is smaller than $\delta_2$. When the lens diameter $2W$ is large compared to $z_0\beta$, where $\beta$ is the angle subtended by the blur spot, the values of $\delta_1$, $\delta_2$ and $p$ are approximately the same and the object plane at $z_o$ falls approximately halfway between two out-of-focus planes located as shown to the left and right of $z_o$. We assume that the system parameters and the diameter $d$ of the blur spot satisfy these conditions.

### 3.4 Coordinate Systems

There are many different coordinate systems that describe the distribution of data points (pixels) in optical data storage systems. The most common is the rectangular (Cartesian) coordinate system. We also consider the hexagonal coordinate system and analyze and perform simulations based on both. Other coordinate systems are possible but may be more difficult to implement because of their more irregular nature.

The most conventional coordinate system is the rectangular grid system, consisting of aligned vertical and horizontal grids. The data pixels and the detector arrays are lined up on these grids to facilitate the data readout process. In this work, we assume that the coordinates are normalized such that the center of each detector is located at integer coordinates and the origin of the coordinates is positioned on the center detector as shown in Figure 3.6.

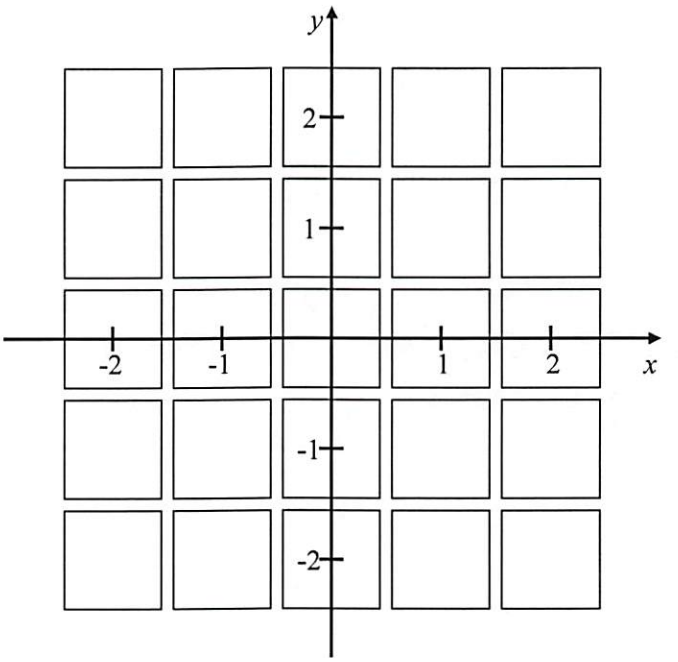



Figure 3.6. Rectangular coordinate system of detectors.

Because most image detectors are made in a rectangular coordinate grid, this is the first system to be analyzed.

We also examine the hexagonal coordinate system in detail as shown in Figure 3.7. Similar to the rectangular coordinate system, the hexagonal coordinate system also

consists of aligned vertical and horizontal grids. However, there is a relative displacement between even numbered rows and odd numbered rows.

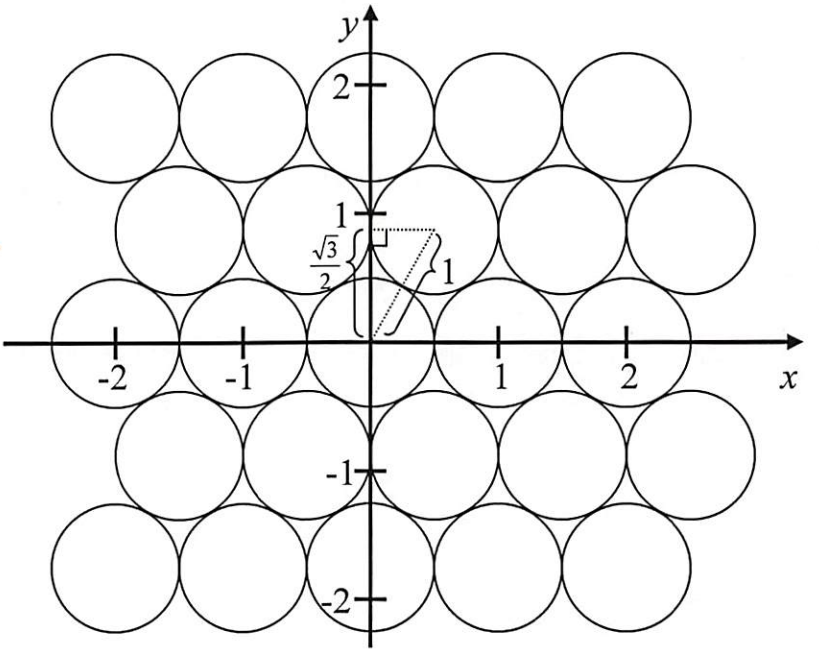



Figure 3.7. Hexagonal coordinate system of detectors.

Although we use circular detectors with a unit diameter in this example, the detector shape can be a circle or hexagon. The advantage of the hexagonal coordinate system over the rectangular coordinate system is the aerial density. From the above figure, the pitch between the centers of two detector rows is $\sqrt{3}/2$ compared to the unit detector pitch of the rectangular coordinate system. Therefore, the aerial density is $2/\sqrt{3} = 1.1547$ times that of the aerial density of the rectangular coordinate system. Thus, using the hexagonal coordinate system provides about 15% or more capacity

over its rectangular counterpart. In addition, the six closest surrounding neighbors are all equidistant in the hexagonal coordinate system. Therefore, the hexagonal coordinate system may be more suitable for optical systems with circular symmetry, such as readout systems with a circular aperture.

### 3.5 Three Dimensional Incoherent Point Spread Function

We model the effects of ISI and IPI as a set of discrete weights located in a three-dimensional block surrounding the center data point (pixel) to be estimated. As an example, we assume the ISI and IPI are negligible at distances far from the center and limit the effects of the PSF to a finite $3 \times 3 \times 3$ volume as shown in the figure below.

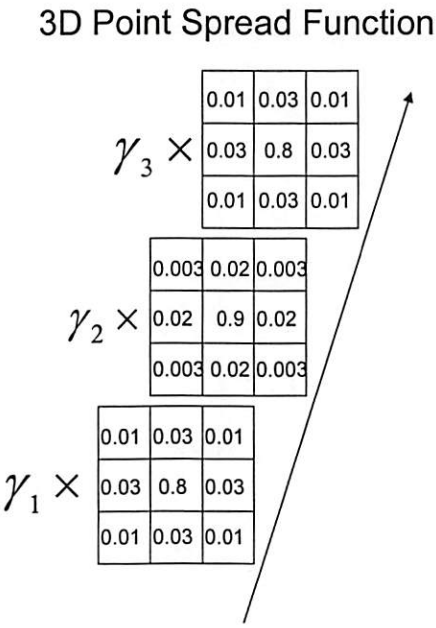



Figure 3.8. Example of space-limited 3D point spread function.

Therefore, for any mark pixel, it will generate limited range ISI to the neighbors. For the above example, the ISI effect is shown in the following figure.

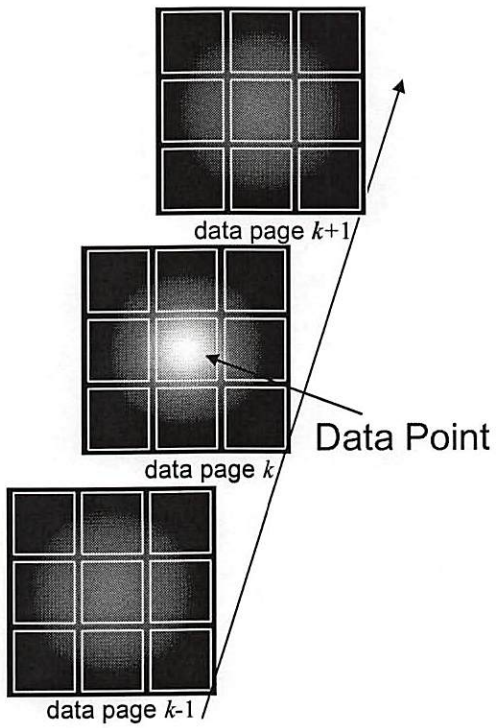



Figure 3.9. Example of ISI effect.

### 3.5.1 Incoherent Point Spread Function Without Focusing Error

From Section 3.1, we know that the detector readout intensity is the sum of light from many statistically independent fluorescent elements in the 3D PODS, and can be modeled as an incoherent imaging system. The incoherent point spread function (PSF) is determined by the aperture of the imaging system. Some examples of the *diffraction-limited* incoherent PSF are shown below, while the PSF with focusing

error is introduced in the next section. For an imaging system with square aperture, the incoherent PSF is

$$h_{dl}(x,y) = \frac{1}{b_s^2}\,\mathrm{sinc}^2\left(\frac{x}{b_s}, \frac{y}{b_s}\right), \tag{3.8}$$

where $b_s$ is inversely proportional to the width of the aperture.

The factor $1/b_s^2$ in front of the PSF is used to normalize the integral of the PSF over all values of $(x,y)$ of the PSF to one. In an incoherent imaging system, this integral is proportional to the total energy in the PSF for a $\delta$ function input. Several PSFs with different $b_s$ values are shown in Figure 3.10. As the $b_s$ value increases, the width of the main lobe also increases. Therefore, more energy is extended to the neighbors and the ISI effect worsens.

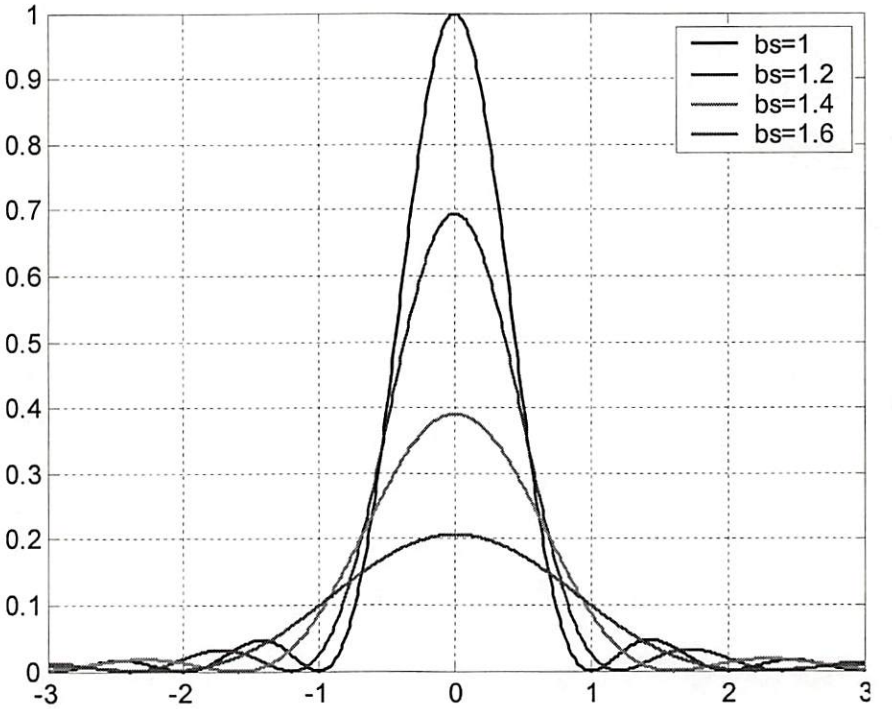



Figure 3.10. The $\text{sinc}^2$ PSF along the $x$-axis with various values of $b_s$.

Using the Rayleigh resolution criterion, two points are said to be "just resolved" when the minimum of one PSF is located at the first maximum of its closest neighbor PSF [12]. Hence, for the incoherent $\text{sinc}^2$ PSF used in Eq. (3.8), the Rayleigh resolution criterion is equivalent to $b_s$=1. For $b_s$ values larger than 1, the two PSFs are closer than the Rayleigh resolution criterion. The Rayleigh resolution limit is shown in Figure 3.11.

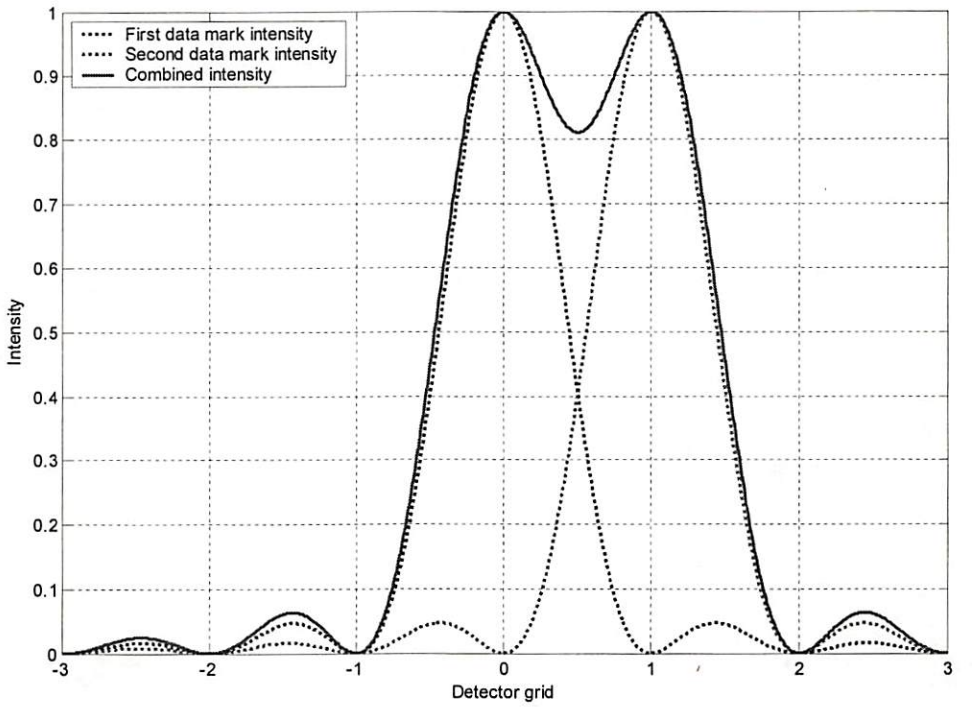



Figure 3.11. The $\text{sinc}^2$ PSF corresponding to the Rayleigh resolution criterion.

For a readout system with a circular aperture of the, the incoherent PSF is

$$h_{dl}(x,y) = \frac{1}{\pi b_c^2}\left[\frac{J_1(2\pi\sqrt{x^2+y^2}/b_c}{\sqrt{x^2+y^2}/b_c}\right]^2 \tag{3.9}$$

Equation (3.9) is also called the jinc function for its resemblance to the sinc function. Similar to the rectangular case, the factor $1/\pi b_c^2$ is used to normalize the volume of the PSF to one. Because this PSF is circularly symmetric, the hexagonal coordinate detector system is usually used with it. Several PSFs with different $b_c$ values are shown in Figure 3.12.

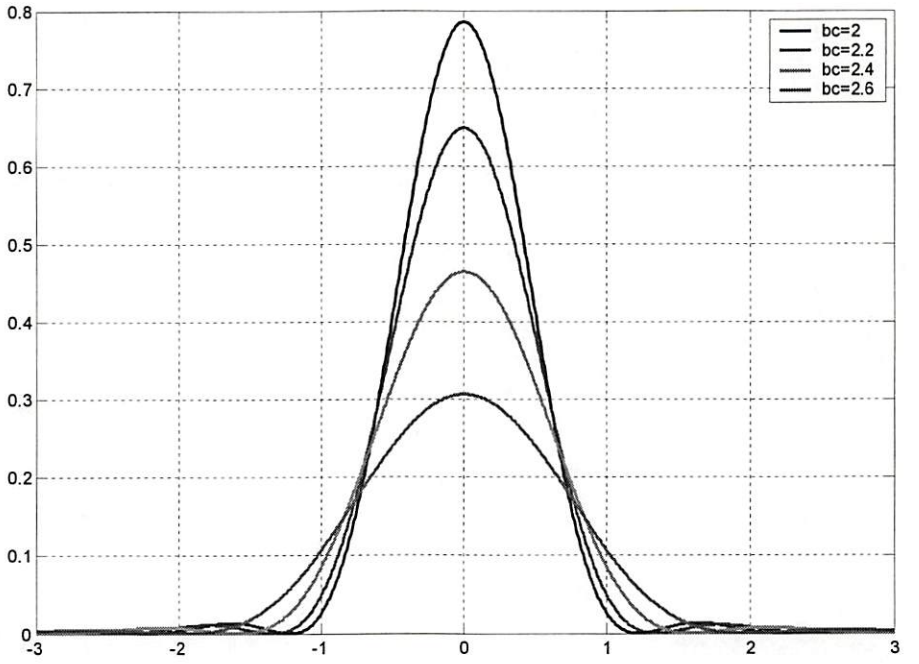



Figure 3.12. The $J_1^2$ PSF along the $x$-axis for various values of $b_c$.

As before, the Rayleigh criterion is defined as the first zero of PSF falling on the maximum of the neighbor's PSF. In this case, the corresponding $b_c$ value of Rayleigh criterion is 1.6398, which is calculated by using a numerical solution. An example of Rayleigh resolution limit is shown in Figure 3.13.

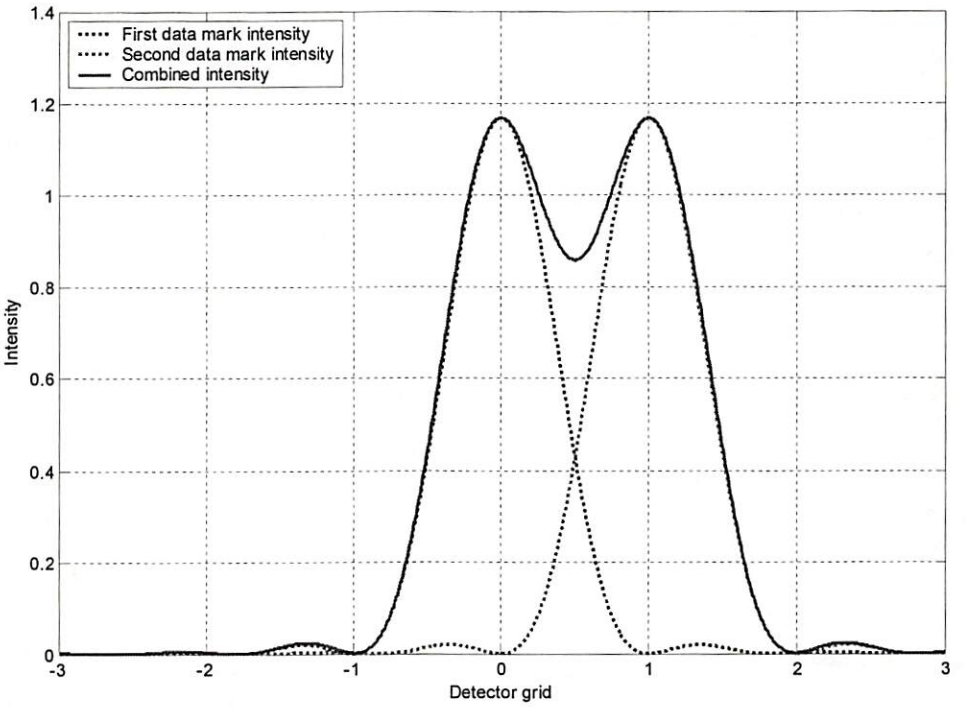



Figure 3.13. The $J_1^2$ PSF corresponding to the Rayleigh resolution criterion.

Comparing the Rayleigh resolution examples for $\text{sinc}^2$ and $J_1^2$ PSF shows that the relative height of the sidelobe of $J_1^2$ PSF is smaller than that of the $\text{sinc}^2$ PSF. Based on numerical integration, the percentage of main lobe energy to the overall energy for the $\text{sinc}^2$ PSF is 93.42%, while this percentage for the $J_1^2$ PSF is 97.93% in the Rayleigh resolution limit. Therefore, the energy is slightly more concentrated for the $J_1^2$ PSF than that for the $\text{sinc}^2$ PSF. In other words, the ISI effect of the $J_1^2$ PSF may be slightly less, and may be is advantageous when choosing the aperture of the readout system.

### 3.5.2 Incoherent Point Spread Function With Focusing Error

For a simple focusing error, the OTF of the system in Figure 3.4 with a square aperture of size $2w \times 2w$ in the pupil plane is

$$\mathcal{H}(f_X, f_Y) = \Lambda\left(\frac{f_X}{2f_o}\right)\Lambda\left(\frac{f_Y}{2f_o}\right) \times \text{sinc}\left[\frac{8W_m}{\lambda}\left(\frac{f_X}{2f_o}\right)\left(1-\frac{|f_X|}{2f_o}\right)\right]\text{sinc}\left[\frac{8W_m}{\lambda}\left(\frac{f_Y}{2f_o}\right)\left(1-\frac{|f_Y|}{2f_o}\right)\right] \tag{3.10}$$

where

$$W_m = \frac{-1}{2}\left(\frac{1}{z_a} - \frac{1}{z_o}\right)(2W)^2, \tag{3.11}$$

$z_i$ is the distance from the aperture to the image plane, $z_a$ is the distance from the aperture to the object plane, and $f_o = \frac{w}{\lambda z_i}$. By varying the object plane distance $z_a$, we get the cross section graph of the PSF as a function of $W_m$ shown in Figure 3.14.

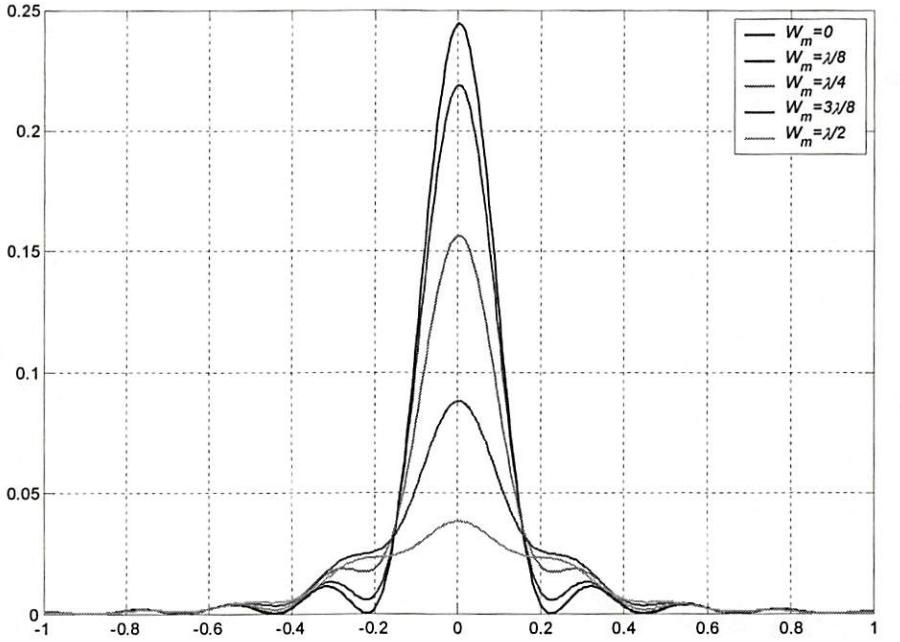



Figure 3.14. Cross section of different out-of-focus PSF as a function of $W_m$.

In addition, we can see the out-of-focus effects from the optical transfer function (OTF), which is the Fourier transform of the PSF. For different $W_m$ values, we plot the cross section of OTF in Figure 3.15.

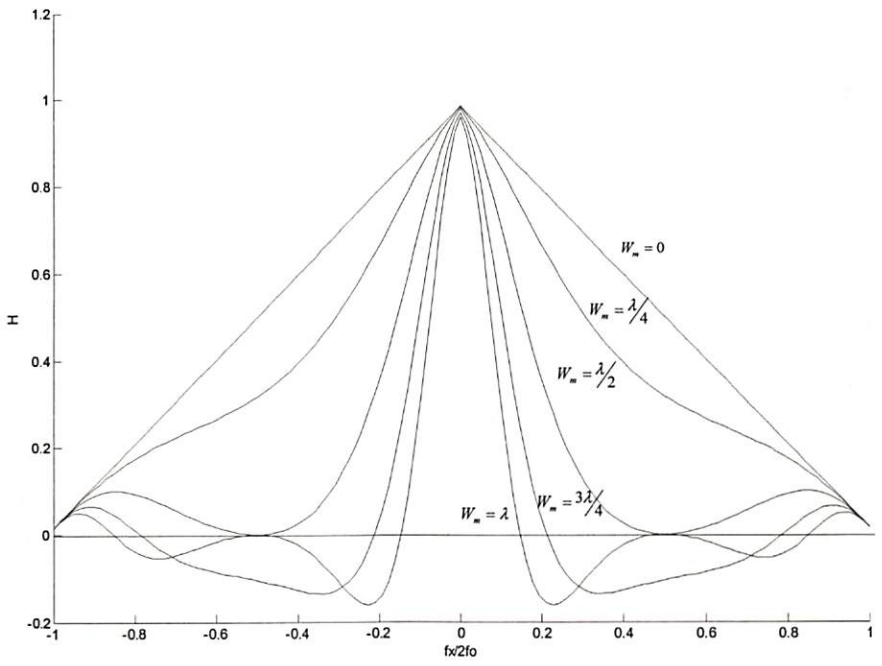



Figure 3.15. Cross section of different out-of-focus OTF as a function of $W_m$.

In general, the incoherent point spread function (PSF) of the system with focusing error is given by

$$h_{defocus}(x,y) = h_{dl}(x,y) \otimes F^{-1}\left\{ \operatorname{sinc}\left[ \frac{8W_m}{\lambda}\left(\frac{f_X}{2f_o}\right)\left(1-\frac{|f_X|}{2f_o}\right)\right] \operatorname{sinc}\left[ \frac{8W_m}{\lambda}\left(\frac{f_Y}{2f_o}\right)\left(1-\frac{|f_Y|}{2f_o}\right)\right]\right\} \tag{3.12}$$

where $h_{dl}(x,y)$ is the diffraction-limited incoherent PSF of the system, $\otimes$ denotes convolution, and the other parameters are the same as described above [12]. Although Eqs. (3.10) and (3.12) are derived assuming a square pupil aperture, similar results are obtained for apertures with different shapes. Thus Eqs. (3.10) and (3.12) are effectively linear filters applied to the aberration-free PSF. Representing Eq. (3.12) in matrix form, it becomes

$$\mathbf{H}_{defocus} = \mathbf{P}\mathbf{H}_{dl} \tag{3.13}$$

where the matrix **P** represents the second term on the right hand side of Eq. (3.12).

### 3.6 Readout Light Intensity Distribution

As shown in Figure 3.4, the finite width of the readout beam is intended to illuminate only Data Page 2, but may partially excite a response in adjacent pages 1 and 3, producing unwanted out-of-focus interfering signals at the image plane. Again, we set the $p$ as the fixed inter-page spacing in the page-oriented optical memory. We assume the intensity of the readout light beam has a Gaussian distribution profile as shown in Figure 3.16, and note that the Gaussian readout light distribution may excite the adjacent Data Pages 1 and 3. We call the excitation of these two unwanted data pages interfering light. Here, we express the proportion of the interfering light intensity from Data Pages 1 and 3 to the overall incident light intensity by the inter-page interference coefficients as $\gamma_1$ and $\gamma_3$, respectively, whose value between 0 and 1 models the degree of IPI.

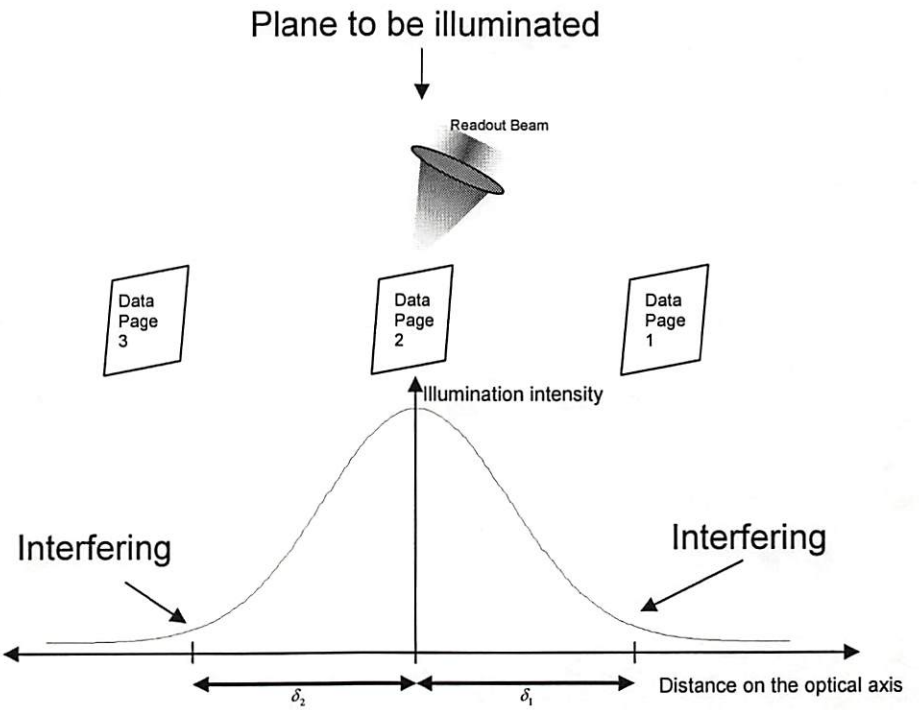



Figure 3.16. Readout light intensity distribution profile.

The measured signal at the image plane is

$$\mathbf{r} = \gamma_1 \mathbf{D}_{defocus}\mathbf{a}_1 + \gamma_2 \mathbf{D}_{dl}\mathbf{a}_2 + \gamma_3 \mathbf{D}_{defocus}\mathbf{a}_3 + \mathbf{v}, \tag{3.14}$$

where $\mathbf{a}_1$, $\mathbf{a}_2$ and $\mathbf{a}_3$ are lexicographically ordered vectors representing the 2D binary data on the three data pages, $\mathbf{v}$ is a 2D measurement noise vector, and $\mathbf{D}_{defocus}$ and $\mathbf{D}_{dl}$ are lexicographically ordered measurement matrices derived from impulse response matrices $\mathbf{H}_{dl}$ and $\mathbf{H}_{defocus}$. Here we use the matrix-vector multiplication to represent the convolution. Considering a linear time-invariant system with impulse response *h(n)*, input *x(n)*, and output *y(n)* as shown in Figure 3.17.

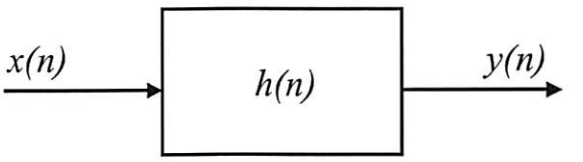



Figure 3.17. Linear time-invariant system.

In the convolution notation, the input-output relationship is

$$y(n) = h(n) \otimes x(n). \tag{3.15}$$

The matrix-vector representation of the above equation is

$$\mathbf{y} = \mathbf{H}\mathbf{x}, \tag{3.16}$$

where

$$\mathbf{X} = \begin{pmatrix} x(0) \\ x(1) \\ x(2) \\ x(3) \\ \vdots \end{pmatrix}, \tag{3.17}$$

$$\mathbf{Y} = \begin{pmatrix} y(0) \\ y(1) \\ y(2) \\ y(3) \\ \vdots \end{pmatrix}, \tag{3.18}$$

and

$$\mathbf{H} = \begin{pmatrix} h(0) & 0 & 0 & 0 & \cdots \\ h(1) & h(0) & 0 & 0 & \cdots \\ h(2) & h(1) & h(0) & 0 & \cdots \\ h(3) & h(2) & h(1) & h(0) & \cdots \\ \vdots & h(3) & h(2) & h(1) & \cdots \end{pmatrix}. \tag{3.19}$$

The matrix **H** is called the convolution matrix and has the structure of a Toeplitz matrix. In each column of the matrix **H**, the entries are the sampled impulse response of the system separated by one sample delay. Similarly, we represent the two dimensional convolution in matrix-vector multiplication. First, we write the two dimensional $n \times n$ data page into one dimensional vector $\mathbf{a}_i$ with lexicographically order of size $n^2 \times 1$ as

$$\mathbf{a}_i = \left\{ \begin{array}{|c|} \hline 1^{st} \text{ column of Data Page} \\ \hline 2^{nd} \text{ column of Data Page} \\ \hline 3^{rd} \text{ column of Data Page} \\ \hline \vdots \\ \end{array} \right\}. \tag{3.20}$$

Assuming the two dimensional impulse responses $\mathbf{H}_{dl}$ and $\mathbf{H}_{defocus}$ are of size $m \times m$, we rewrite them into vector forms as

$$\mathbf{d} = \left\{ \begin{array}{|c|c|c|c|} \hline \text{1st column of H} & \text{2nd column of H} & \text{3rd column of H} & \\ \hline \end{array} \right\}, \tag{3.21}$$

where each block is zero padded to the size of $(n+m-1)\times 1$ and vector $\mathbf{d}$ is of size $(n+m-1)^2 \times 1$. Similar to the convolution matrix in Eq. (3.19), we use the vector $\mathbf{d}$ to form the lexicographically ordered measurement matrix as

$$\mathbf{D} = \begin{pmatrix} \mathbf{d} & 0 & 0 & \cdots \\ 0 & \mathbf{d} & 0 & \cdots \\ 0 & 0 & \mathbf{d} & \cdots \\ \vdots & \vdots & \vdots & \end{pmatrix}, \tag{3.22}$$

where the size of matrix $\mathbf{D}$ is $(n+m-1)^2 \times n^2$. In each column of matrix $\mathbf{D}$, it is one sample delay to the column on the left. Therefore, it is also a Toeplitz matrix. The receiving signal vector $\mathbf{r}$ and the measurement noise vector $\mathbf{v}$ in Eq. (3.19) are similar to the one dimensional data page vector $\mathbf{a}_i$ as

$$\mathbf{r} = \begin{Bmatrix} \boxed{\begin{matrix}\text{1st column}\\ \text{of Received}\\ \text{Data Page}\end{matrix}} \\ \boxed{\begin{matrix}\text{2nd column}\\ \text{of Received}\\ \text{Data Page}\end{matrix}} \\ \boxed{\begin{matrix}\text{3rd column}\\ \text{of Received}\\ \text{Data Page}\end{matrix}} \\ \bullet \\ \bullet \\ \bullet \end{Bmatrix}, \tag{3.23}$$

and

$$\mathbf{v} = \begin{Bmatrix} \boxed{\begin{matrix}\text{1st column}\\ \text{of 2D noise}\end{matrix}} \\ \boxed{\begin{matrix}\text{2nd column}\\ \text{of 2D noise}\end{matrix}} \\ \boxed{\begin{matrix}\text{3rd column}\\ \text{of 2D noise}\end{matrix}} \\ \bullet \\ \bullet \\ \bullet \end{Bmatrix}, \tag{3.24}$$

both with size of $(n+m-1)^2 \times 1$.

We normalize the received signal versus the incident light intensity. Therefore, we have

$$\gamma_1 + \gamma_2 + \gamma_3 = 1. \tag{3.25}$$

From Figure 3.16, assuming that the illumination beam has a symmetrical profile, we assume that $\gamma_1 = \gamma_3$ and use a single variable $\gamma$ to represent them. Therefore, we have $\gamma_2 = 1 - 2\gamma$. Then,

$$\mathbf{r} = \gamma \mathbf{D}_{defocus}\left(\mathbf{a}_1 + \mathbf{a}_3\right) + (1 - 2\gamma)\mathbf{D}_{dl}\mathbf{a}_2 + \mathbf{v}, \tag{3.26}$$

and Eq. (3.14) are our basic measurement models in this work.

# Chapter 4 Least Square Equalization for 3D Page-Oriented Data Storage Systems

Here we describe a general least square equalization method based on the imaging model of Eqs. (3.14) and (3.26). The method is an equalization procedure that uses information about the inter-page interference coefficient and threshold detection following estimation. From Eq. (3.26), each measured page contains data from one directly illuminated page and two partially illuminated pages. We make use of all these data to get the estimates.

We modify Eq. (3.26) to include a general page index $k$ in the received signal to obtain

$$\mathbf{r}_k = \gamma \mathbf{D}_{defocus}\left(\mathbf{a}_{k-1} + \mathbf{a}_{k+1}\right) + (1-2\gamma)\mathbf{D}_{dl}\mathbf{a}_k + \mathbf{v}_k . \tag{4.1}$$

We write three consecutive received data pages in a matrix form as

$$\begin{bmatrix} \mathbf{r}_{k+1} \\ \mathbf{r}_k \\ \mathbf{r}_{k-1} \end{bmatrix} = \begin{bmatrix} \gamma_3\mathbf{D}_{defocus} & \gamma_2\mathbf{D}_{dl} & \gamma_1\mathbf{D}_{defocus} & 0 & 0 \\ 0 & \gamma_3\mathbf{D}_{defocus} & \gamma_2\mathbf{D}_{dl} & \gamma_1\mathbf{D}_{defocus} & 0 \\ 0 & 0 & \gamma_3\mathbf{D}_{defocus} & \gamma_2\mathbf{D}_{dl} & \gamma_1\mathbf{D}_{defocus} \end{bmatrix} \begin{bmatrix} \mathbf{a}_{k+2} \\ \mathbf{a}_{k+1} \\ \mathbf{a}_k \\ \mathbf{a}_{k-1} \\ \mathbf{a}_{k-2} \end{bmatrix} + \begin{bmatrix} \mathbf{v}_{k+1} \\ \mathbf{v}_k \\ \mathbf{v}_{k-1} \end{bmatrix}, \tag{4.2}$$

where $\mathbf{r}_{k+1}$, $\mathbf{r}_k$, $\mathbf{r}_{k-1}$ are the received signals at clock times $k$+1, $k$, and $k$-1 containing measurements of data page $\mathbf{a}_k$. Here $\mathbf{a}_{k-2}$ is the data page just before data page $\mathbf{a}_{k-1}$,

and $\mathbf{a}_{k+2}$ is the data page just after data page $\mathbf{a}_{k+1}$. We summarize Eq. (4.2) as

$$\underline{\mathbf{r}} = \underline{\underline{\mathbf{D}}}\underline{\mathbf{a}} + \underline{\mathbf{v}}, \tag{4.3}$$

where $\underline{\mathbf{r}} = [\mathbf{r}_{k+1} \quad \mathbf{r}_k \quad \mathbf{r}_{k-1}]^T$, $\underline{\mathbf{a}} = [\mathbf{a}_{k+2} \quad \mathbf{a}_{k+1} \quad \mathbf{a}_k \quad \mathbf{a}_{k-1} \quad \mathbf{a}_{k-2}]^T$, $\underline{\mathbf{v}} = [\mathbf{v}_{k+1} \quad \mathbf{v}_k \quad \mathbf{v}_{k-1}]^T$ and $\mathbf{D}$ is the matrix in Eq. (4.2). Here and in the following sections, bold lowercase letters with an underbar denote a vector which is a combination of other vectors.

The least square equalization [27] algorithm operates in a ***batch*** or ***recursive*** mode. Both of these two different modes will lead to the same results given the same set of measurements.

### 4.1 *Batch* Mode Operation

We have the least square estimator of the pages as

$$\hat{\underline{\mathbf{a}}} = \underline{\underline{\mathbf{D}}}^{+}\underline{\mathbf{r}}, \tag{4.4}$$

where

$$\hat{\underline{\mathbf{a}}} = [\hat{\mathbf{a}}_{k+2} \quad \hat{\mathbf{a}}_{k+1} \quad \hat{\mathbf{a}}_k \quad \hat{\mathbf{a}}_{k-1} \quad \hat{\mathbf{a}}_{k-2}]^T \tag{4.5}$$

and $\underline{\underline{\mathbf{D}}}^{+}$ is the pseudoinverse of matrix $\underline{\underline{\mathbf{D}}}$. We only need $\hat{\mathbf{a}}_k$ as the estimate of Data Page $\mathbf{a}_k$. In this model, the matrix $\underline{\underline{\mathbf{D}}}$ is fixed for every data page and every clock time. Hence we only need to calculate it once for the entire readout process and save

computation. Because it is a linear operation on the received signals, we represent the least square estimator as

$$\hat{\mathbf{a}}_k = \mathbf{C}_1\mathbf{r}_{k+1} + \mathbf{C}_2\mathbf{r}_k + \mathbf{C}_3\mathbf{r}_{k-1}, \tag{4.6}$$

where matrices $\mathbf{C}_1$, $\mathbf{C}_2$ and $\mathbf{C}_3$ are derived from $\underline{\mathbf{D}}^+$. For the underdetermined matrix $\underline{\mathbf{D}}$, we can write the pseudoinverse matrix $\underline{\mathbf{D}}^+$ as

$$\underline{\mathbf{D}}^+ = \underline{\mathbf{D}}^T\left(\underline{\mathbf{D}}\underline{\mathbf{D}}^T\right)^{-1}. \tag{4.7}$$

Since we only need part of the estimate vector $\hat{\mathbf{a}}$ in Eq. (4.4) to get the estimate $\hat{\mathbf{a}}_k$, we do not need to calculate the whole pseudoinverse matrix $\underline{\mathbf{D}}^+$. The complete pseudoinverse matrix is

$$\underline{\mathbf{D}}^+ = \frac{1}{\Delta}\begin{bmatrix} \gamma_3\mathbf{D}_{defocus}\Psi & \gamma_3\mathbf{D}_{defocus}\Phi \\ \gamma_3\mathbf{D}_{defocus}\Phi + \gamma_2\mathbf{D}_{dl}\Psi & \gamma_3\mathbf{D}_{defocus}\Lambda + \gamma_2\mathbf{D}_{dl}\Phi \\ \gamma_3\mathbf{D}_{defocus}\Gamma + \gamma_2\mathbf{D}_{dl}\Phi + \gamma_1\mathbf{D}_{defocus}\Psi & \gamma_3\mathbf{D}_{defocus}\Phi + \gamma_2\mathbf{D}_{dl}\Lambda + \gamma_1\mathbf{D}_{defocus}\Phi \\ \gamma_2\mathbf{D}_{dl}\Gamma + \gamma_1\mathbf{D}_{defocus}\Phi & \gamma_2\mathbf{D}_{dl}\Phi + \gamma_1\mathbf{D}_{defocus}\Lambda \\ \gamma_1\mathbf{D}_{defocus}\Gamma & \gamma_1\mathbf{D}_{defocus}\Phi \end{bmatrix} \tag{4.8}$$

$$\begin{bmatrix} \gamma_3\mathbf{D}_{defocus}\Gamma \\ \gamma_3\mathbf{D}_{defocus}\Phi + \gamma_2\mathbf{D}_{dl}\Gamma \\ \gamma_3\mathbf{D}_{defocus}\Psi + \gamma_2\mathbf{D}_{dl}\Phi + \gamma_1\mathbf{D}_{defocus}\Gamma \\ \gamma_2\mathbf{D}_{dl}\Psi + \gamma_1\mathbf{D}_{defocus}\Phi \\ \gamma_1\mathbf{D}_{defocus}\Psi \end{bmatrix}$$

where

$$\begin{aligned}\Delta = &\left(\gamma_2 \mathbf{D}_{dl}\right)^6 + \left(\gamma_1 \mathbf{D}_{defocus}\right)^2 \left(\gamma_2 \mathbf{D}_{dl}\right)^4 + \left(\gamma_1 \mathbf{D}_{defocus}\right)^4 \left(\gamma_2 \mathbf{D}_{dl}\right)^2 + \left(\gamma_1 \mathbf{D}_{defocus}\right)^6 \\ &- 4\gamma_1\gamma_3 \mathbf{D}^2_{defocus} \left(\gamma_2 \mathbf{D}_{dl}\right)^4 - 2\gamma_1^3\gamma_3 \mathbf{D}^4_{defocus} \left(\gamma_2 \mathbf{D}_{dl}\right)^2 + \left(\gamma_2 \mathbf{D}_{dl}\right)^4 \left(\gamma_3 \mathbf{D}_{defocus}\right)^2 \\ &+ 5\gamma_1^2\gamma_3^2 \mathbf{D}^4_{defocus} \left(\gamma_2 \mathbf{D}_{dl}\right)^4 + 2\gamma_1^4\gamma_3^2 \mathbf{D}^6_{defocus} - 2\gamma_1\gamma_3^3 \mathbf{D}^4_{defocus} \left(\gamma_2 \mathbf{D}_{dl}\right)^2 \\ &+ \left(\gamma_2 \mathbf{D}_{dl}\right)^2 \left(\gamma_3 \mathbf{D}_{defocus}\right)^4 + 2\gamma_1^2\gamma_3^4 \mathbf{D}^6_{defocus} + \left(\gamma_3 \mathbf{D}_{defocus}\right)^6\end{aligned}, \tag{4.9}$$

$$\Phi = -\gamma_1 \mathbf{D}_{defocus} \left(\gamma_2 \mathbf{D}_{dl}\right)^3 - \left(\gamma_1 \mathbf{D}_{defocus}\right)^3 \gamma_2 \mathbf{D}_{dl} - \left(\gamma_2 \mathbf{D}_{dl}\right)^3 \gamma_3 \mathbf{D}_{defocus} - \gamma_2 \mathbf{D}_{dl} \left(\gamma_3 \mathbf{D}_{defocus}\right)^3, \tag{4.10}$$

$$\begin{aligned}\Gamma = &\left(\gamma_1 \mathbf{D}_{defocus}\right)^2 \left(\gamma_2 \mathbf{D}_{dl}\right)^2 + \gamma_1\gamma_3 \mathbf{D}^2_{defocus} \left(\gamma_2 \mathbf{D}_{dl}\right)^2 - \gamma_1^3\gamma_3 \mathbf{D}^4_{defocus} + \left(\gamma_2 \mathbf{D}_{dl}\right)^2 \left(\gamma_3 \mathbf{D}_{defocus}\right)^2 \\ &- \gamma_1\gamma_3^3 \mathbf{D}^4_{defocus}\end{aligned}, \tag{4.11}$$

$$\Psi = -\left(\gamma_1 \mathbf{D}_{defocus} \gamma_2 \mathbf{D}_{dl} + \gamma_2 \mathbf{D}_{dl} \gamma_3 \mathbf{D}_{defocus}\right)^2 + \left(\left(\gamma_1 \mathbf{D}_{defocus}\right)^2 + \left(\gamma_2 \mathbf{D}_{dl}\right)^2 + \left(\gamma_3 \mathbf{D}_{defocus}\right)^2\right)^2, \tag{4.12}$$

and

$$\Lambda = -\gamma_1^2\gamma_3^2 \mathbf{D}^4_{defocus} + \left(\left(\gamma_1 \mathbf{D}_{defocus}\right)^2 + \left(\gamma_2 \mathbf{D}_{dl}\right)^2 + \left(\gamma_3 \mathbf{D}_{defocus}\right)^2\right)^2. \tag{4.13}$$

Therefore, we have

$$\mathbf{C}_1 = \frac{1}{\Delta}\left[\gamma_3 \mathbf{D}_{defocus} \Gamma + \gamma_2 \mathbf{D}_{dl} \Phi + \gamma_1 \mathbf{D}_{defocus} \Psi\right], \tag{4.14}$$

$$\mathbf{C}_2 = \frac{1}{\Delta}\left[\gamma_3 \mathbf{D}_{defocus} \Phi + \gamma_2 \mathbf{D}_{dl} \Lambda + \gamma_1 \mathbf{D}_{defocus} \Phi\right], \tag{4.15}$$

$$\mathbf{C}_3 = \frac{1}{\Delta}\left[\gamma_3 \mathbf{D}_{defocus} \Psi + \gamma_2 \mathbf{D}_{dl} \Phi + \gamma_1 \mathbf{D}_{defocus} \Gamma\right]. \tag{4.16}$$

Using this estimator, we use all the received signals that contain the measurements of the desired data page. Thus, there is a delay of at least the inter-page arrival time before the estimate is produced. However, assumed that files occupy several sequential data pages, so that if $\mathbf{a}_k$ is read, then $\mathbf{a}_{k-1}$ and $\mathbf{a}_{k+1}$ are probably also read. In

such case, the batch mode least square equalization performs estimation as soon as the measurement received. This estimator is the same as the *inverse filter* used for image restoration in image processing applications.

### 4.2 *Recursive* Mode Operation

The recursive least square estimator generates a sequence of estimates based on a partial set of received signals. Based on the batch mode operation, we derive a recursive formula for our data retrieval system. In this mode, we do not need to wait for all the received signals, i.e. $\mathbf{r}_{k+1}$, $\mathbf{r}_k$, $\mathbf{r}_{k-1}$ to estimate the data page $\mathbf{a}_k$. First, if we only receive signal $\mathbf{r}_{k-1}$, Eq. (4.1) becomes

$$\mathbf{r}_{k-1} = \begin{bmatrix} 0 & 0 & \gamma_3 \mathbf{D}_{defocus} & \gamma_2 \mathbf{D}_{dl} & \gamma_1 \mathbf{D}_{defocus} \end{bmatrix} \begin{bmatrix} \mathbf{a}_{k+2} \\ \mathbf{a}_{k+1} \\ \mathbf{a}_k \\ \mathbf{a}_{k-1} \\ \mathbf{a}_{k-2} \end{bmatrix} + \mathbf{v}_{k-1}. \tag{4.17}$$

We write the above equation in compact form as $\mathbf{r}_{k-1} = \mathbf{d}_1^T \underline{\mathbf{a}} + \mathbf{v}_{k-1}$, where

$$\mathbf{d}_1 = \begin{bmatrix} 0 \\ 0 \\ \gamma_3 \mathbf{D}_{defocus} \\ \gamma_2 \mathbf{D}_{dl} \\ \gamma_1 \mathbf{D}_{defocus} \end{bmatrix}, \tag{4.18}$$

and

$$\underline{\mathbf{a}} = \begin{bmatrix} \mathbf{a}_{k+2} & \mathbf{a}_{k+1} & \mathbf{a}_k & \mathbf{a}_{k-1} & \mathbf{a}_{k-2} \end{bmatrix}^T. \tag{4.19}$$

We add a subscript to the estimated $\underline{\mathbf{a}}$ variable to denote the sequence of measurements that are received. This subscript is called a *recursive loop index.* Hence, our least square estimate at loop index 1 is

$$\hat{\underline{\mathbf{a}}}_1 = \mathbf{d}_1\left[\mathbf{d}_1^T\mathbf{d}_1\right]^{-1}\mathbf{r}_{k-1}. \tag{4.20}$$

When the second received signal $\mathbf{r}_k$ comes in, the recursive least square estimator is

$$\hat{\underline{\mathbf{a}}}_2 = \hat{\underline{\mathbf{a}}}_1 + \mathbf{d}_2\mathbf{P}_2\left[\mathbf{r}_k - \mathbf{d}_2^T\hat{\underline{\mathbf{a}}}_1\right], \tag{4.21}$$

and

$$\mathbf{P}_2 = \left[\mathbf{P}_1^{-1} + \mathbf{d}_2^T\mathbf{d}_2\right]^{-1}, \tag{4.22}$$

where

$$\mathbf{P}_1 = \left[\mathbf{d}_1^T\mathbf{d}_1\right]^{-1}, \tag{4.23}$$

and

$$\mathbf{d}_2 = \begin{bmatrix} 0 \\ \gamma_3\mathbf{D}_{defocus} \\ \gamma_2\mathbf{D}_{dl} \\ \gamma_1\mathbf{D}_{defocus} \\ 0 \end{bmatrix}. \tag{4.24}$$

Finally, when the third received signal $\mathbf{r}_{k+1}$ arrives, we have the final estimate $\hat{\underline{\mathbf{a}}}_3$ as

$$\hat{\underline{\mathbf{a}}}_3 = \hat{\underline{\mathbf{a}}}_2 + \mathbf{d}_3\mathbf{P}_3\left[\ \mathbf{r}_{k+1} - \mathbf{d}_3^T\hat{\underline{\mathbf{a}}}_2\right], \tag{4.25}$$

and

$$\mathbf{P}_3 = \left[\mathbf{P}_2^{-1} + \mathbf{d}_3^T\mathbf{d}_3\right]^{-1}, \tag{4.26}$$

where

$$\mathbf{d}_3 = \begin{bmatrix} \gamma_3 \mathbf{D}_{defocus} \\ \gamma_2 \mathbf{D}_{dl} \\ \gamma_1 \mathbf{D}_{defocus} \\ 0 \\ 0 \end{bmatrix}. \tag{4.27}$$

In general, the recursive least square estimator at loop index $l$ is

$$\underline{\hat{\mathbf{a}}}_l = \left[\mathbf{I} - \mathbf{d}_l \mathbf{P}_l \mathbf{d}_l^T\right] \underline{\hat{\mathbf{a}}}_{l-1} + \mathbf{d}_l \mathbf{P}_l \mathbf{r}_{k+l-2} \tag{4.28}$$

and

$$\mathbf{P}_l = \left[\mathbf{P}_{l-1}^{-1} + \mathbf{d}_l^T \mathbf{d}_l\right]^{-1} \tag{4.29}$$

and $\mathbf{d}_l$ is the corresponding transposed row vector of the matrix $\underline{\mathbf{D}}$. The recursive mode estimate for a data page $\mathbf{a}_k$ is the same as that from the batch mode given all relevant observations $\mathbf{r}_{k+1}$, $\mathbf{r}_k$, $\mathbf{r}_{k-1}$. However, in recursive mode operation, the estimate is updated based on the previous estimate and the new received signal in each recursive loop. The estimate result is improved in each recursive loop. Thus the recursive mode results are increasingly accurate as additional signals are received. In addition, the matrices $\mathbf{P}_1$, $\mathbf{P}_2$, $\mathbf{P}_3$ and $\mathbf{d}_1$, $\mathbf{d}_2$, $\mathbf{d}_3$ are all fixed through the readout process. We can pre-calculate them and save computation.

## 4.3 Time Complexity Analysis

### 4.3.1 *Batch* Mode Estimation

Any data page $\mathbf{a}_k$ with size $n \times n$ can be formed into a lexicographically ordered vector with size $n^2 \times 1$. If we truncate the two dimensional point spread function into

an $m \times m$ mask, we represent the two dimensional convolution of the data page with the point spread function by a matrix multiplication of the form as

$$\mathbf{r}_k = \gamma \mathbf{D}_{defocus}(\mathbf{a}_{k-1} + \mathbf{a}_{k+1}) + (1-2\gamma)\mathbf{D}_{dl}\mathbf{a}_k + \mathbf{v}_k, \tag{4.30}$$

where $\mathbf{D}_{defocus}$ and $\mathbf{D}_{dl}$ are lexicographically ordered measurement matrices derived from impulse response matrices with size $(n+m-1)^2 \times n^2$. Therefore, the received signal $\mathbf{r}_k$ is a vector of size $(n+m-1)^2 \times 1$. The general form of least square batch mode estimation is

$$\hat{\mathbf{a}}_k = \mathbf{C}_1\mathbf{r}_{k+1} + \mathbf{C}_2\mathbf{r}_k + \mathbf{C}_3\mathbf{r}_{k-1}. \tag{4.31}$$

The matrices $\mathbf{C}_1$, $\mathbf{C}_2$ and $\mathbf{C}_3$ are of size $n^2 \times (n+m-1)^2$ respectively. Hence, to get the estimate of each data page, we need $3 \times n^2 \times (n+m-1)^2$ operations. For any size of the PSF mask, the time complexity has the upper bound denoted as $O(n^4) \approx 3n^4$.

From Eqs. (4.14) through (4.16), the matrices $\mathbf{C}_1$, $\mathbf{C}_2$ and $\mathbf{C}_3$ consist of linear combinations of matrices $\mathbf{D}_{defocus}$ and $\mathbf{D}_{dl}$, which are lexicographically ordered impulse response matrices. Since the central part of the two dimensional impulse response contains significant amount of energy than the rest, it is often truncated into an $m \times m$ mask. Therefore, matrices $\mathbf{D}_{defocus}$ and $\mathbf{D}_{dl}$ only have limited entries with nonzero values, and the linear combinations of them also possess this property.

From the least square batch mode estimation shown in Eq. (4.31), the effective equalization processing window is the extent of the truncated PSF mask. To estimate

any given data mark in the data page, the complexity is proportional to the extent of the ISI window size $m$, which is much smaller than the data page size $n$. Thus the asymptotic time complexity is $\Omega(n^2)$.

### 4.3.2 *Recursive* Mode Estimation

The general form of least square recursive mode estimation from Eq. (4.28) is

$$\underline{\hat{\mathbf{a}}}_l = \left[\mathbf{I} - \mathbf{d}_l \mathbf{P}_l \mathbf{d}_l^T\right] \underline{\hat{\mathbf{a}}}_{l-1} + \mathbf{d}_l \mathbf{P}_l \mathbf{r}_{k+l-2}. \tag{4.32}$$

although in each recursive loop, the matrix $\mathbf{P}_l$ also needs to be updated using the equation

$$\mathbf{P}_l = \left[\mathbf{P}_{l-1}^{-1} + \mathbf{d}_l^T \mathbf{d}_l\right]^{-1}. \tag{4.33}$$

Because these matrices $\mathbf{d}_l$ are derived from the fixed matrix $\underline{\mathbf{D}}$, the matrix $\mathbf{P}_l$ is also fixed in each iteration step and can be computed in advance. Therefore, the term $\mathbf{d}_l \mathbf{P}_l$ and $\mathbf{d}_l^T$, with size $5n^2 \times (n+m-1)^2$ and $(n+m-1)^2 \times 5n^2$ respectively, can be pre-computed. With the estimated result $\hat{\mathbf{a}}_l$ of size $5n^2 \times 1$ and the received signal $\mathbf{r}_k$ of size $(n+m-1)^2 \times 1$, we need roughly $2 \times 5n^2 \times (n+m-1)^2$ operations in each iteration step. Assuming using estimates with three data pages, the upper bound of time complexity for the same estimation result as the batch mode is $10n^4$ operations, denoted as $O(n^4)$.

Since the estimation of any given data mark is done within a finite equalization window similar to the batch mode operation, the asymptotic complexity of the

recursive mode operation is also $\Omega(n^2)$. In the batch mode operation, all the received signals are processed in one shot to reach the estimation. However, in the recursive mode operation, only one received signal is used in each iteration loop to update the estimation. In Eq. (4.32), the first term on the right hand side can be seen as an overhead term associated with the update process. Thus, in comparison, the recursive mode generally needs slightly more operations than the batch mode to get the same estimation result.

# Chapter 5 Estimation and Detection Results

In this chapter we present experimental results based on the algorithms described in Chapter 4. We use the bit-error-rate (BER) and the signal-to-noise (SNR) ratio as performance measures for comparison of various systems. The simulations are carried out on both rectangular and hexagonal coordinate systems.

## 5.1 Experimental Parameters and Simulation Procedure

The physical parameters used in the simulation are adopted from an actual experimental system made by Call/Recall Inc.[4]. Their two photon multilayer disk has 400 layers. The related parameters are listed in Table 5.1.

| | |
|---|---|
| Square aperture size | 4 mm x 4 mm |
| Focal length | 8 mm |
| Readout wavelength | 635 nm |
| Pixel separation on the data page | 10 μm |
| Data page separation | 30 μm |
| Focusing error parameter $W_m$ | 0.0934 μm |

Table 5.1. System parameters used in simulations

The data page size in our simulations is $100 \times 100$ pixels. These data pages are filled with randomly generated 0s and 1s with equal probability. For each different

simulation setting, we create at least $10^5$ data pages and average the error statistics over all pages in order to have reliable BER and SNR comparisons. The diffraction-limited incoherent point spread function used in the rectangular array detectors from Eq.(3.8) is

$$h_{dl}(x,y) = \frac{1}{b_s^2}\mathrm{sinc}^2\left(\frac{x}{b_s},\frac{y}{b_s}\right), \tag{5.1}$$

where $b_s$ represents the degree of inter-symbol interference. When the value of $b_s$ is large, the pixel blur at the detector becomes worse and it degrades the system performance. Similarly, we use the diffraction-limited incoherent point spread function from Eq.(3.9) in the hexagonal array detectors as

$$h_{dl}(x,y) = \frac{1}{\pi b_c^2}\left[\frac{J_1(2\pi\sqrt{x^2+y^2}/b_c}{\sqrt{x^2+y^2}/b_c}\right]^2. \tag{5.2}$$

The $b_c$ parameter controls the degree of ISI in the hexagonal array system as $b_s$ does in the rectangular array system. Therefore, we also call the $b_s$ and $b_c$ parameters as the blur factors of the PSF.

In addition to the ISI, the inter-page interference (IPI) further corrupts the received signal. From Eq. (3.14), we use the $\gamma_1$ and $\gamma_3$ parameters to represent the degree of inter-page interference. The values of these parameters range from 0 to 1. We assume that IPI from pages before and after the page of interest is the same, so the $\gamma_1$ and $\gamma_3$ parameters specifying the degree of IPI are the same.

The performance of the readout system is affected significantly by both ISI and IPI. The remainder of this section describes the choice of blur factors and the normalization procedure used in the simulations. We compare the effects of ISI for rectangular and hexagonal coordinate systems. The blur factors $b_s$ and $b_c$ are chosen such that the ISI effect ranges from low to severe with both coordinate systems. The PSFs with different blur factors are shown in Figure 5.1.

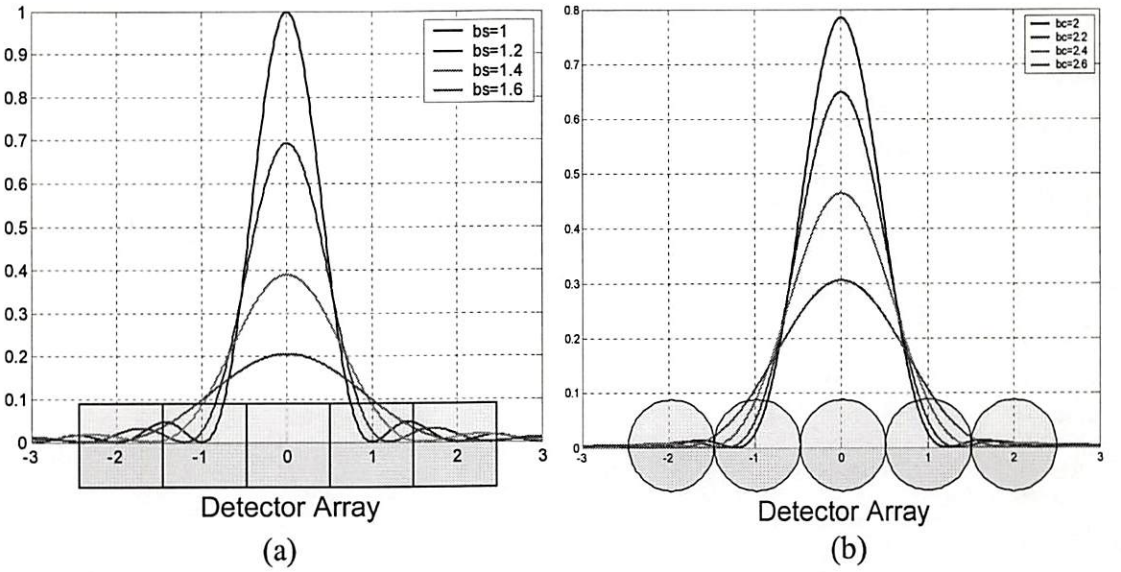



Figure 5.1. PSFs with different blur factors in (a) rectangular (b) hexagonal coordinates.

In Figure 5.1, the center detector array element is labeled with 0. For a fair comparison between two coordinate systems, we define the ratio of received energy integrated over the center detector array element to the total energy of the PSF (describe in Chapter 3) computed by integrating over all values of $(x,y)$ as

$$E = \frac{\iint_{\text{center detector element}} \text{PSF}}{\iint_{\text{all}(x,y)} \text{PSF}}. \tag{5.3}$$

Using the definition of PSF as given in Eqs. (5.1) and (5.2), we have already normalized these two functions, so that

$$\iint_{\text{all}(x,y)} \text{PSF} = 1. \tag{5.4}$$

Therefore, the received energy ratio in Eq. (5.3) for the rectangular coordinate system is

$$E_{\text{rect}} = \iint_{\text{center detector element}} \frac{1}{b_s^2} \text{sinc}^2\left(\frac{x}{b_s}, \frac{y}{b_s}\right) dxdy, \tag{5.5}$$

and for the hexagonal coordinate system is

$$E_{\text{hex}} = \iint_{\text{center detector element}} \frac{1}{\pi b_c^2} \left[\frac{J_1(2\pi\sqrt{x^2+y^2}/b_c}{\sqrt{x^2+y^2}/b_c}\right]^2 dxdy. \tag{5.6}$$

For all comparison of rectangular and hexagonal coordinates, we choose blur factors $b_s$ and $b_c$ that yield the same ratio of received energy, i.e. $E_{\text{rect}}$=$E_{\text{hex}}$, and pair the results. Table 5.2 lists the blur factors that we use in our simulations.

| Degree of ISI | Rectangular coordinate $b_s$ | Hexagonal coordinate $b_c$ | Received energy ratio $E$ |
|---|---|---|---|
| Moderate ISI | 1 | 1.7609 | 62.66% |
| Moderate ISI | 1.2 | 2.0606 | 50.85% |
| Moderate ISI | 1.4 | 2.3620 | 41.62% |
| Severe ISI | 1.5 | 2.5235 | 37.55% |
| Severe ISI | 2 | 3.3055 | 24.11% |
| Severe ISI | 2.5 | 4.1328 | 16.30% |

Table 5.2. Comparison of blur factors for rectangular and hexagonal coordinates.

From Eqs. (3.25) and (5.4), the received signal intensity is normalized such that the overall received intensity from three mark pixels at a particular detector element in three successive pages is 1, i.e., the sum of the three gamma variables is one, and

$$\gamma_1 \iint h_{defocus}(x,y)dxdy + \gamma_2 \iint h_{dl}(x,y)dxdy + \gamma_3 \iint h_{defocus}(x,y)dxdy = 1, \quad (5.7)$$

as shown in Figure 5.2.

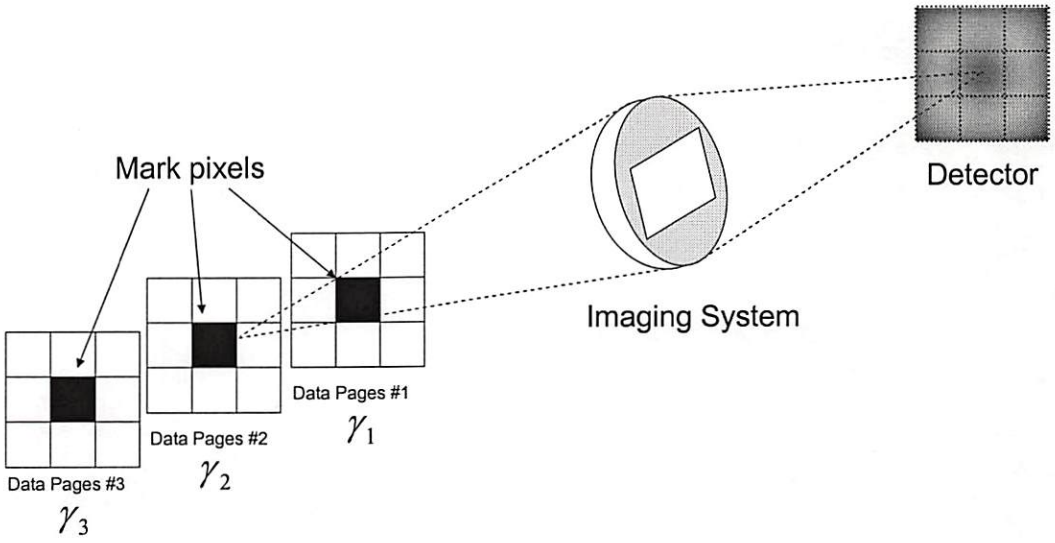


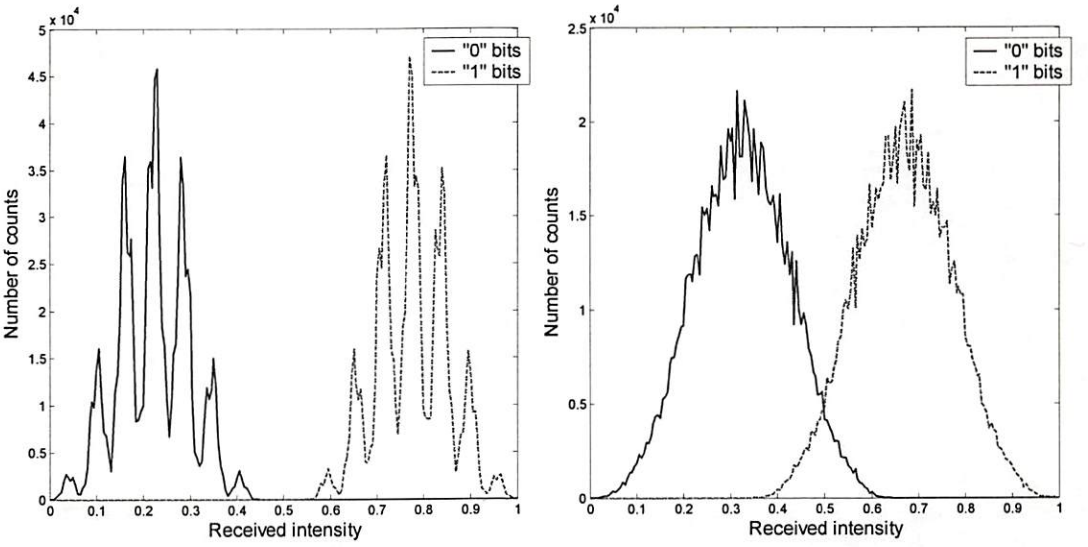

Figure 5.2. Normalization of the received signal intensity

Figure 5.3 through Figure 5.6 show the influence of ISI and IPI on the histograms of the received signal intensity. We eliminate the additive noise in these figures so that we can focus on the effects of ISI and IPI. We collect the received data at the detectors from systems with four different degrees of ISI and two different degrees of IPI.

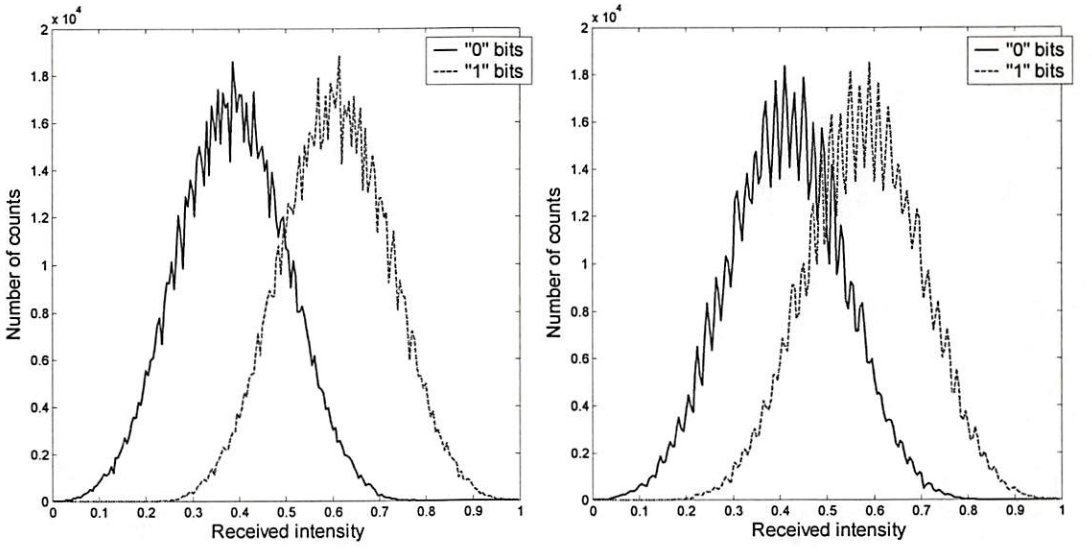
From the histogram figures, we notice that the horizontal extents of "0" bits and "1" bits increase rapidly as the blur factor increases. When these two histograms overlap each other, we cannot guarantee an error-free data retrieval based on a simple thresholding method. Generally, if the overlap area is large, the readout system has a higher error rate at the output.

Comparing the histograms of received signals on rectangular and hexagonal coordinate systems, the signal intensity distribution of rectangular system tends to be

slightly more compact than that of the hexagonal system under similar ISI and IPI effects. In other words, the received signal intensity of the rectangular system forms a narrower peak in the histogram plot compared to that of the hexagonal system. Therefore, we expect that the system performance degrades more with the hexagonal system if a simple thresholding detection method is employed.

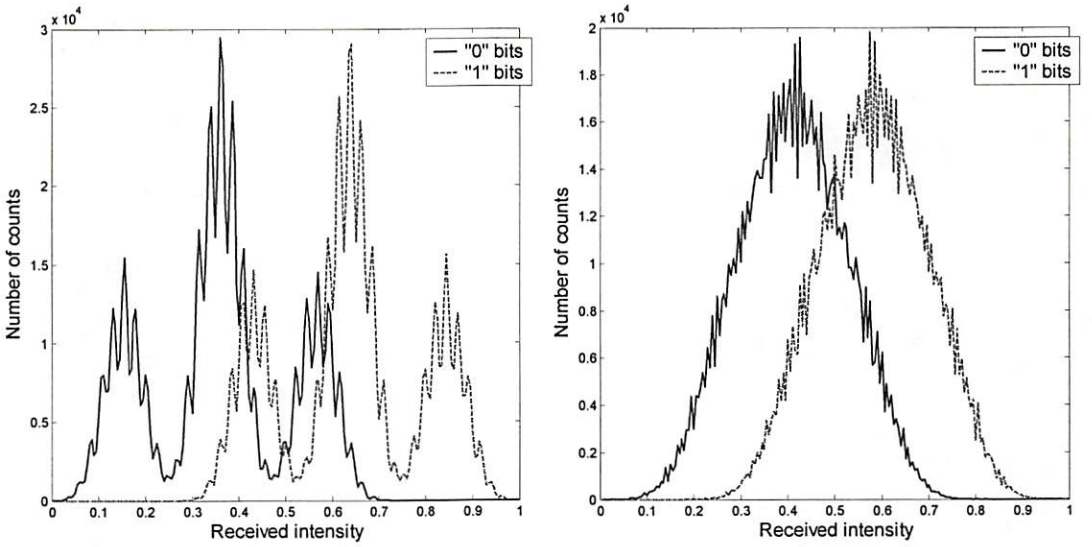



(a) $b_s$=1 (b) $b_s$=1.5

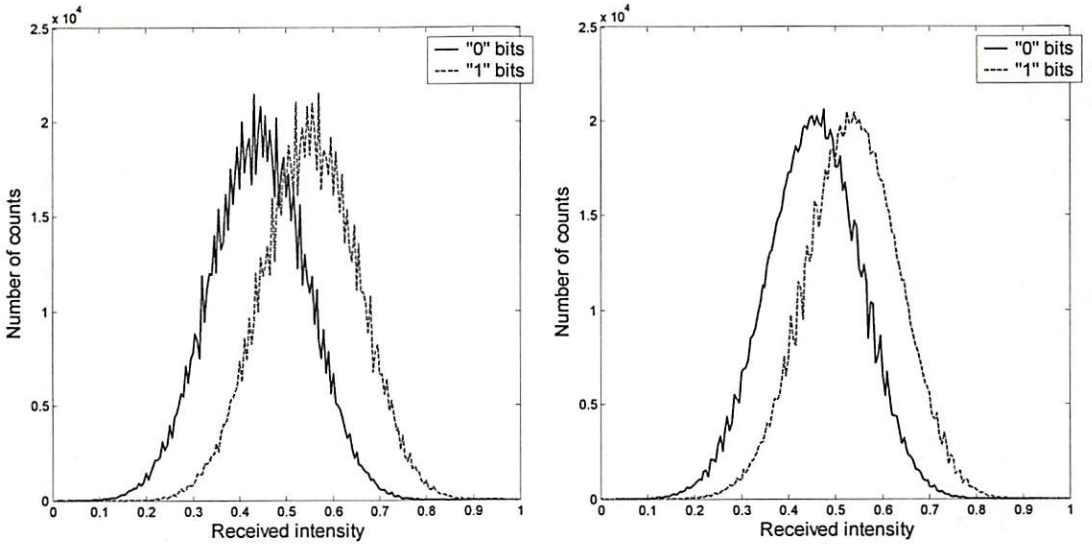



(c) $b_s$=2 (d) $b_s$=2.5

Figure 5.3. Histogram of the received signal intensity with IPI $\gamma_1 = \gamma_3 = 0.1$ on rectangular array detectors.

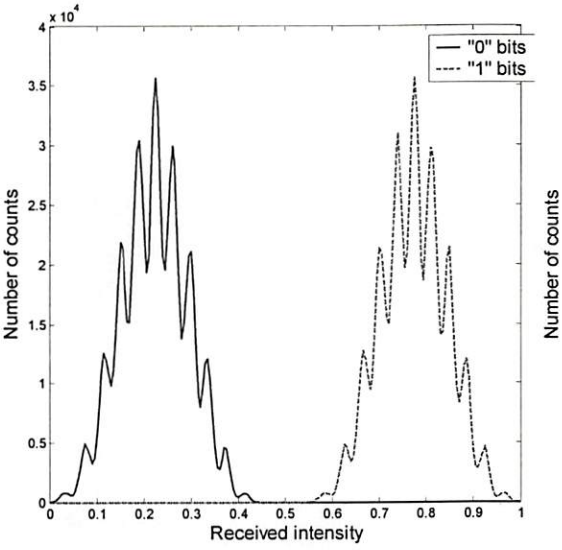



(a) $b_s$=1 (b) $b_s$=1.5

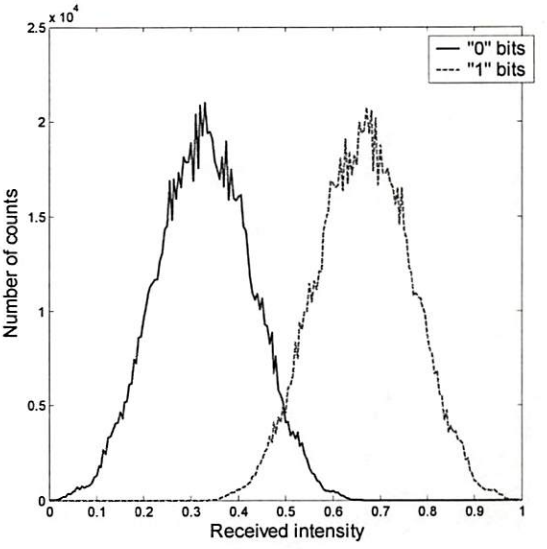



(c) $b_s$=2 (d) $b_s$=2.5

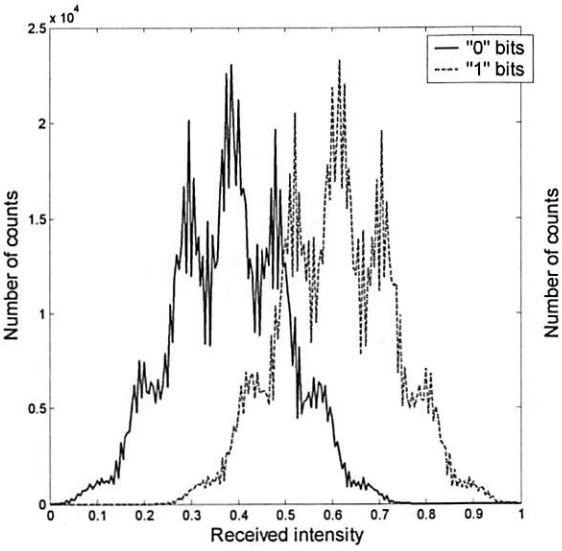
Figure 5.4. Histogram of the received signal intensity with IPI $\gamma_1 = \gamma_3 = 0.3$ on rectangular array detectors.



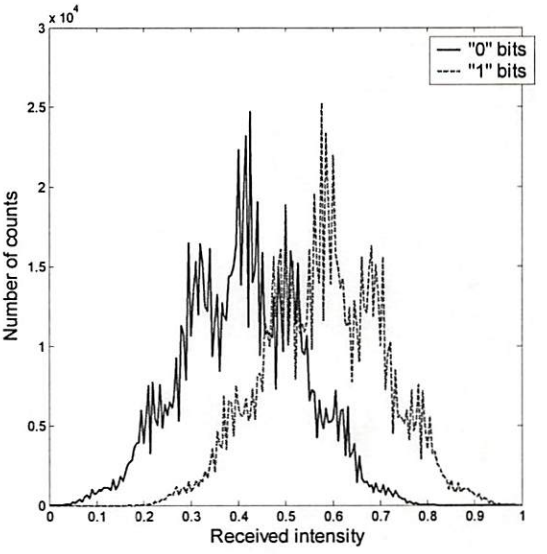

(a) $b_c$=1.76




(b) $b_c$=2.52

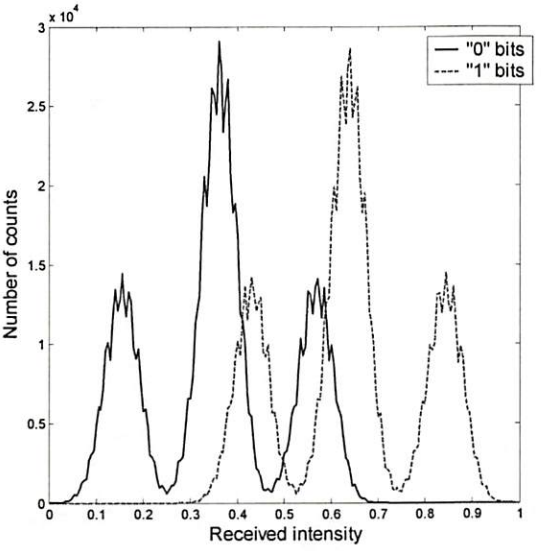



(a) $b_c$=3.31

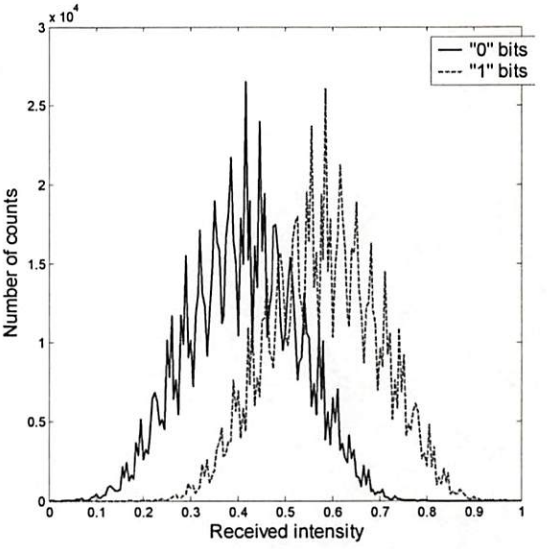



(b) $b_c$=4.13

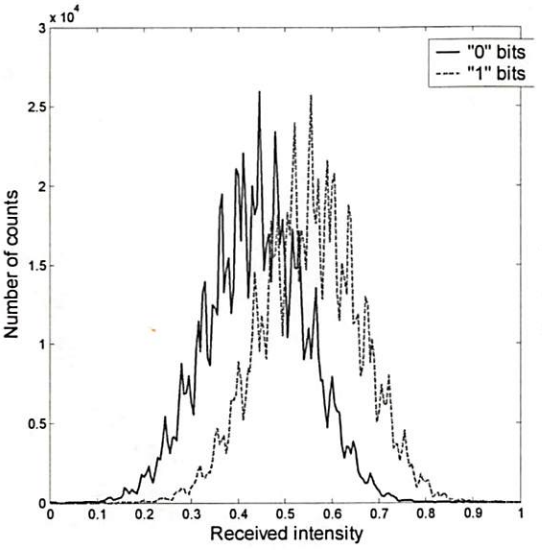
Figure 5.5. Histogram of the received signal intensity with IPI $\gamma_1 = \gamma_3 = 0.1$ on hexagonal array detectors.



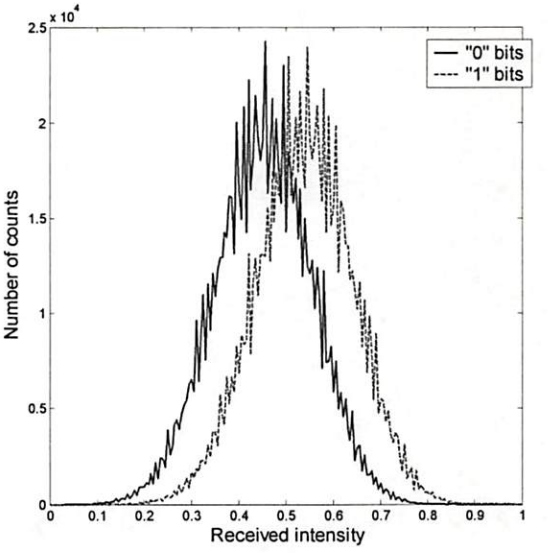

(a) $b_c$=1.76




(b) $b_c$=2.52




(a) $b_c$=3.31




(b) $b_c$=4.13

Figure 5.6. Histogram of the received signal intensity with IPI $\gamma_1 = \gamma_3 = 0.3$ on hexagonal array detectors.

From the histograms in Figure 5.3 through Figure 5.6, it is clear that only when $b_s$=1 and $\gamma_1 = \gamma_3 = 0.1$, can we achieve perfect detection using a simple thresholding method in the rectangular coordinate system. In all other cases, we need to use other detection methods to recover the data. Both ISI and IPI have a strong influence on the system performance.

Two-photon storage systems also have noise (measurement error). The noise may be due to electronic at the detector plane (thermal noise) and Poisson noise due to the statistics of the received photons. For simplicity in our initial study, we model the noise as additive white Gaussian noise (AWGN) with probability density function [1][36]

$$f_v(v) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left( -\frac{(v-m)^2}{2\sigma^2} \right), \quad -\infty < v < \infty \tag{5.8}$$

where $m$ and $\sigma^2$ are mean and variance, respectively. Other more detailed noise models can be used in the future.

From the simulation results shown in the following sections, the least square equalization method outperforms the simple fixed threshold method in most situations. When the inter-symbol interference and inter-page interference are dominant in the 3D Page-Oriented Data Storage System, our method provides a feasible and practical solution.

## 5.2 BER Performance with Different Inter-Page Interference

In this section, we compare the bit error rate performance versus different $\gamma$ values which represents the degree of inter-page interference. From the results, our method provides significant improvement over the conventional method.

Figure 5.7 through Figure 5.14 show the results of the linear minimum mean-square error equalization [30][20] and detection algorithm described in Chapter 4. The simulation results for a rectangular coordinate system are shown in Figure 5.7 through Figure 5.10, and the results for hexagonal coordinate systems are shown in Figure 5.11 through Figure 5.14. The equalization algorithm estimates data pages under different conditions of ISI and additive noise based on information about the inter-page interference coefficient and uses threshold detection following estimation. In the rectangular coordinate system, the parameter $b_s$ in Eq.(5.1) controls the degree of inter-symbol interference produced by the incoherent point spread function. When $b_s$=1, it corresponds to moderate ISI. If $b_s$=1.5, we consider it as severe ISI. Similarly, the $b_c$ controls the ISI effect generated by the PSF in Eq. (5.2) and we consider it as moderate ISI for $b_c$=2 and severe ISI for $b_c$=2.6.

Making the assumption of a symmetrical readout light profile, we use only one parameter $\gamma = \gamma_1 = \gamma_3$ to control the degree of inter-page interference as in Eq. (3.26). As mentioned in Chapter 3, the value of $\gamma$ is normalized to range between 0 and 1. As $\gamma$ increases, a larger portion of the readout light partially excites the adjacent

pages, which in turn produces out-of-focus interfering signals at the detector plane. Therefore, the received signal is corrupted by inter-page interference.

We model the detector noise as additive white Gaussian noise (AWGN), whose mean and variance are zero and $\sigma^2$, respectively. The noise variance $\sigma^2$ is another controlled parameter corresponding to the effects of noise. With the combination of the above three parameters, we can examine our method based on different situations.

The bottom curve in Figure 5.7 through Figure 5.14 shows the consistently better bit-error-rate (BER) performance of the algorithm on the vertical axis as a function of the IPI interference coefficient $\gamma$ (horizontal axis) compared to the unequalized result shown as the top curve. Although the least square equalization has similar performance on the rectangular and hexagonal coordinate systems, it achieves slightly better improvement on the hexagonal coordinate over the rectangular coordinate especially with severe ISI and additive noise.

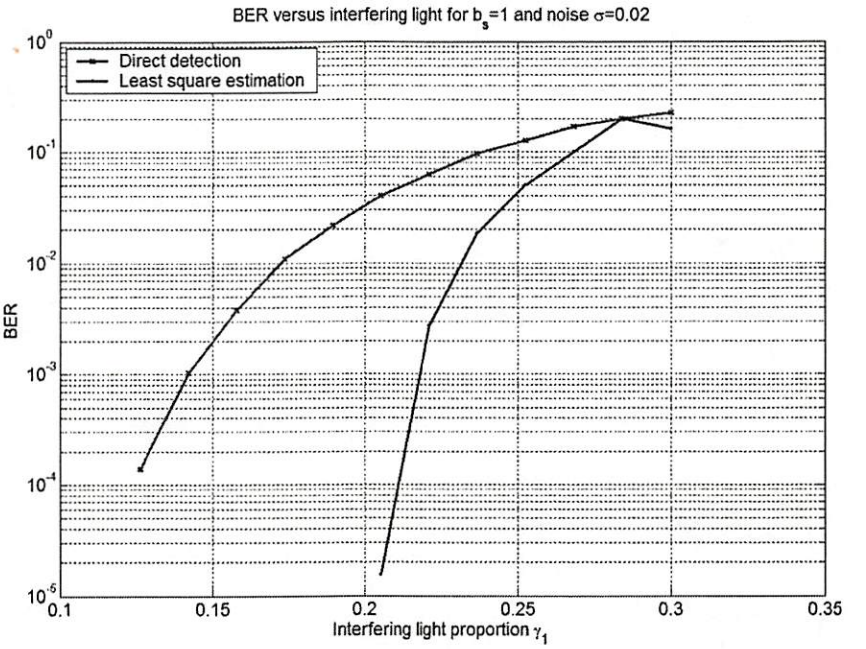



Figure 5.7. Least square equalization results for moderate ISI ($b_s$=1) and moderate noise ($\sigma$=0.02) on rectangular array detectors.

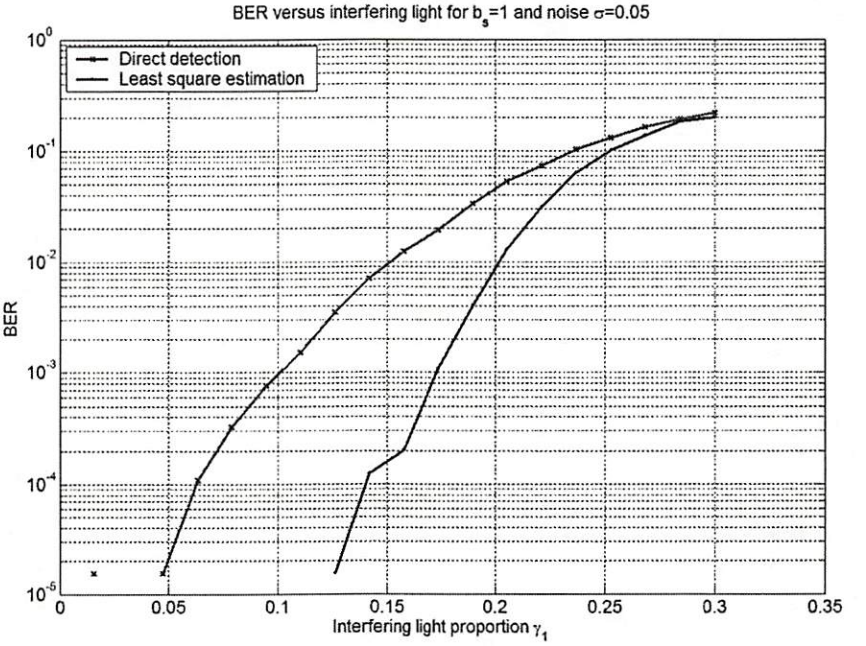



Figure 5.8. Least square equalization results for moderate ISI ($b_s$=1) and severe noise ($\sigma$=0.05) on rectangular array detectors.

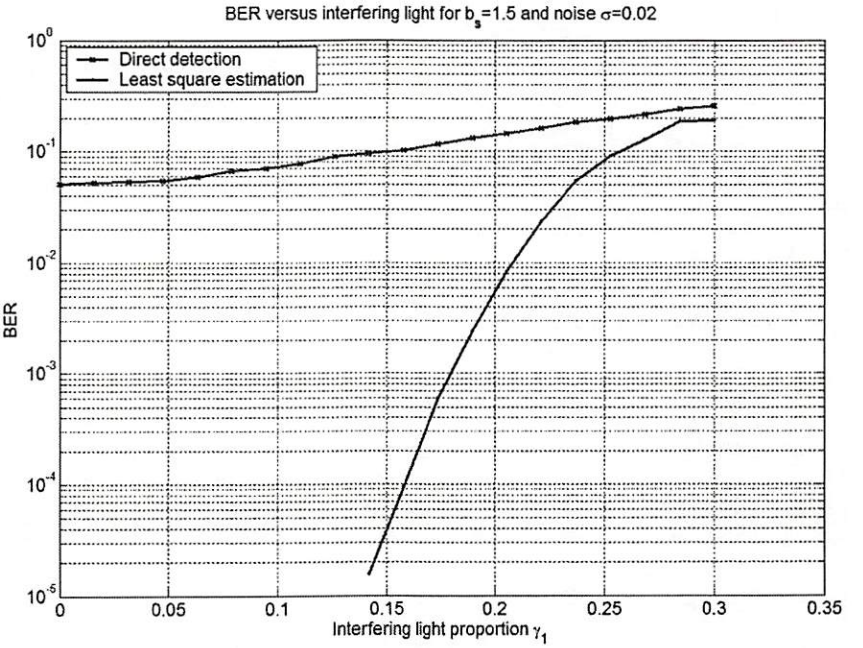



Figure 5.9. Least square equalization results for severe ISI ($b_s$=1.5) and moderate noise ($\sigma$=0.02) on rectangular array detectors.

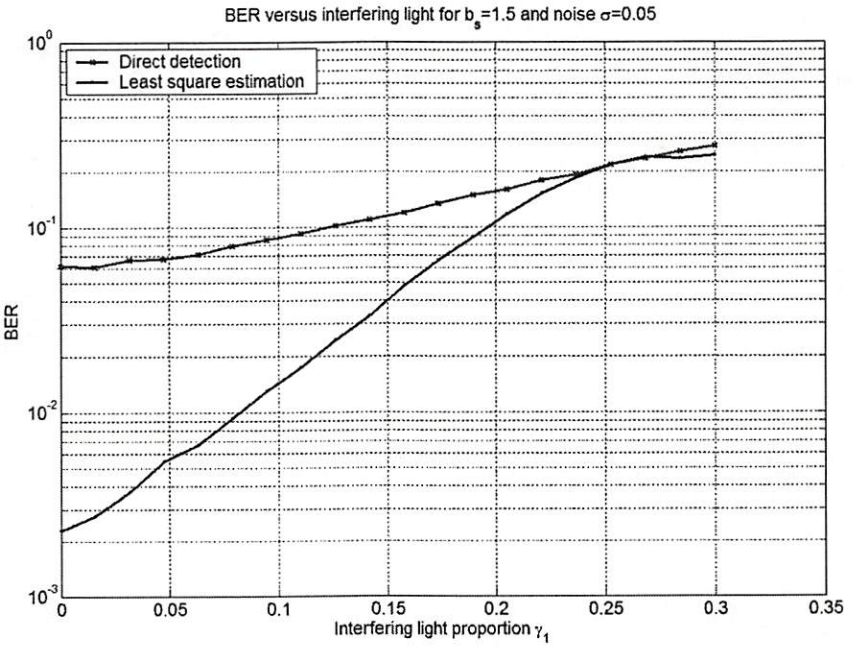



Figure 5.10. Least square equalization results for severe ISI ($b_s$=1.5) and severe noise ($\sigma$=0.05) on rectangular array detectors.

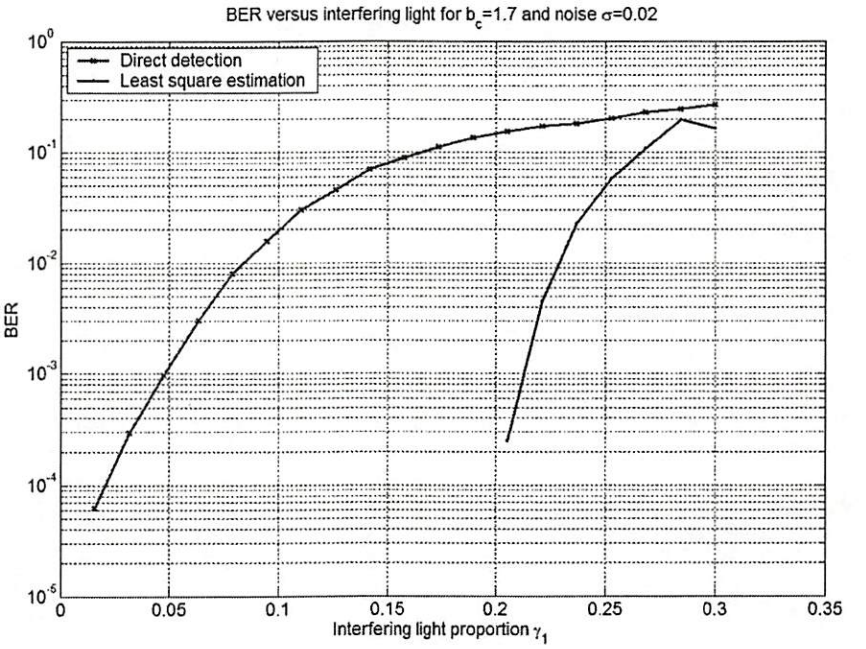



Figure 5.11. Least square equalization results for moderate ISI ($b_c$=1.76) and moderate noise ($\sigma$=0.02) on hexagonal array detectors.

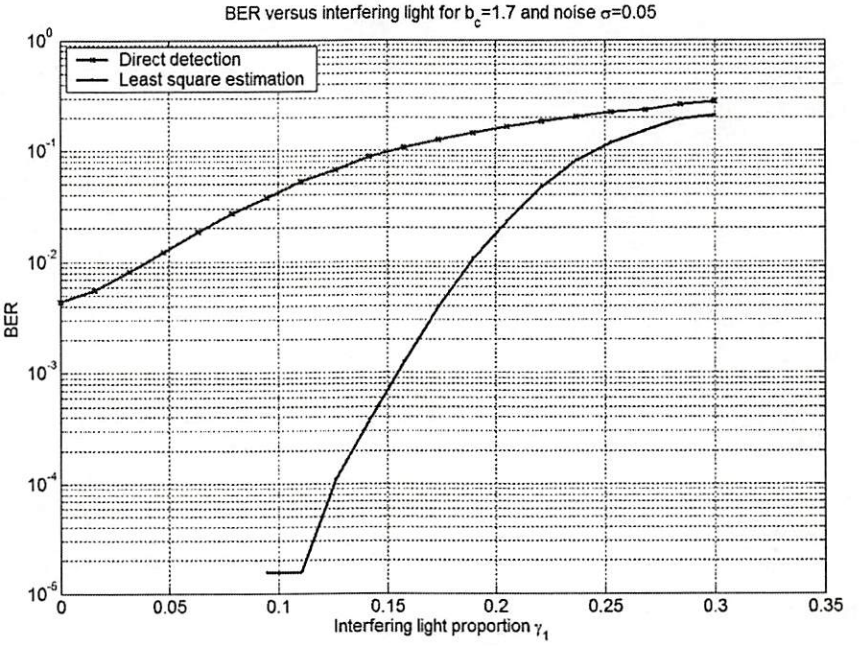



Figure 5.12. Least square equalization results for moderate ISI ($b_c$=1.76) and severe noise ($\sigma$=0.05) on hexagonal array detectors.

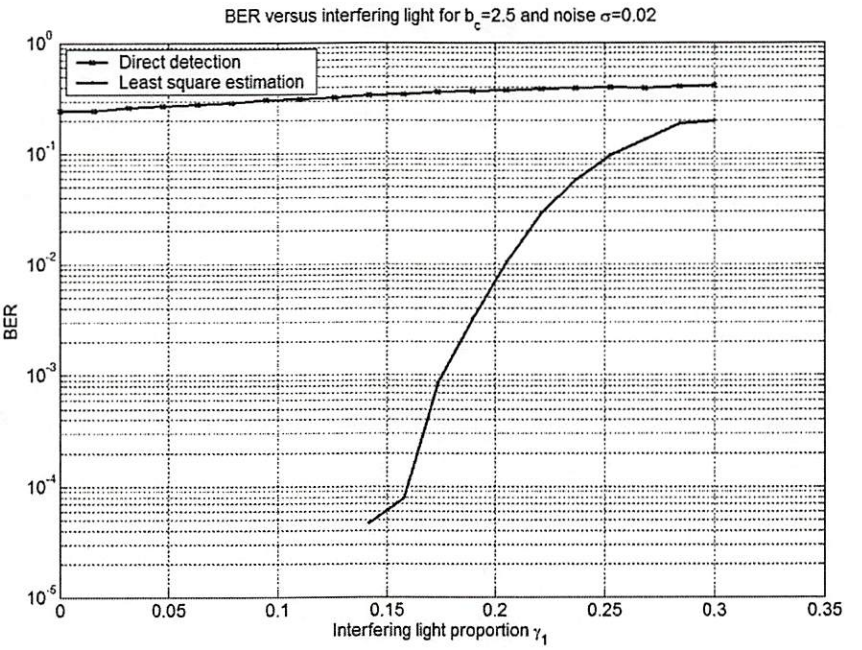



Figure 5.13. Least square equalization results for severe ISI ($b_c$=2.52) and moderate noise ($\sigma$=0.02) on hexagonal array detectors.

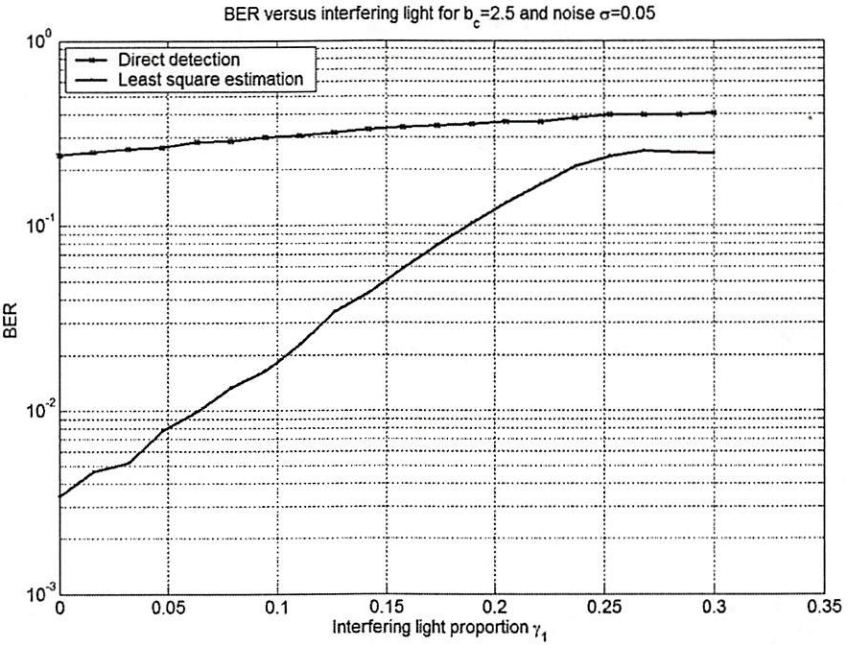



Figure 5.14. Least square equalization results for severe ISI ($b_c$=2.52) and severe noise ($\sigma$=0.05) on hexagonal array detectors.

### 5.3 BER Performance with Different Signal-to-Noise Ratio

In communication systems, a common performance index is the Signal-to-Noise ratio (SNR). We define SNR as

$$\mathrm{SNR} = \frac{\mathrm{var}(r)}{\sigma^2}, \tag{5.9}$$

where var($r$) is the signal variance and $\sigma^2$ is the variance of the AWGN. Figure 5.15 shows the results, in which the vertical axis is the bit-error rate (BER) while the horizontal axis is the SNR expressed in decibels (dB), where dB is defined as

$$(\mathrm{SNR})_{dB} = 10\log_{10}\mathrm{SNR}. \tag{5.10}$$

The curves represent BER as a function of SNR under different degrees of inter-page interference.

We performed the simulations on both the rectangular and hexagonal coordinate systems and plot the results on Figure 5.15 and Figure 5.16 respectively.

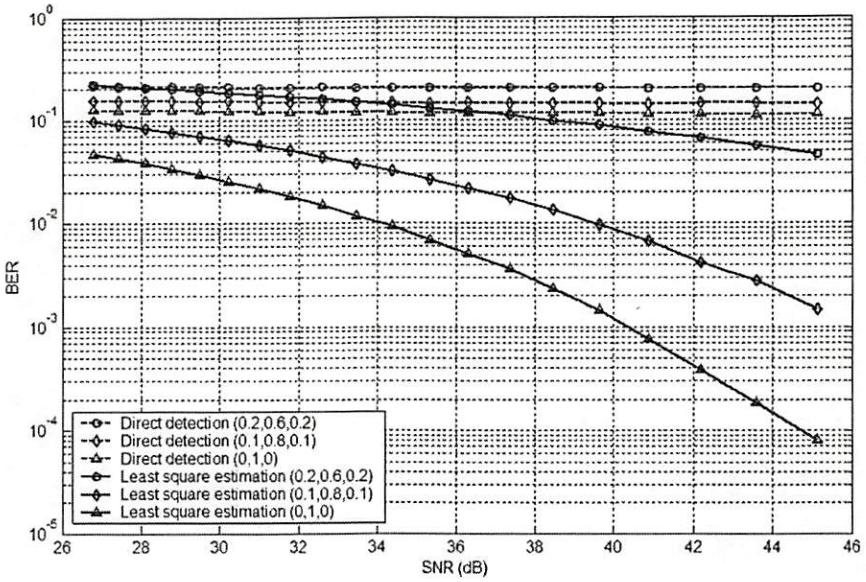



Figure 5.15. Least square equalization results for severe ISI ($b_s$=1.7) and variable SNR and IPI on rectangular array detectors.

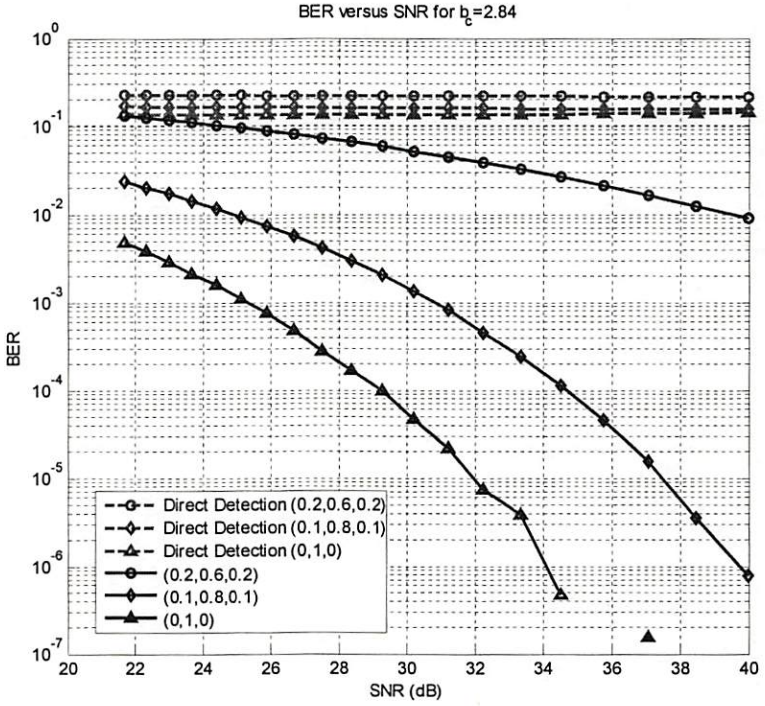



Figure 5.16. Least square equalization results for severe ISI ($b_c$=2.84) and variable SNR and IPI on hexagonal array detectors.

In Figure 5.15 and Figure 5.16, the legend $(\gamma_1, \gamma_2, \gamma_3)$ next to each plotted symbol denotes different degrees of inter-page interference. From Chapter 4, we know that $\gamma_1$ and $\gamma_3$ represent unwanted partially excited light from the adjacent pages. When either of them is large compared to $\gamma_2$, it is considered as severe inter-page interference. Therefore, the bottom curve in the figure where $\gamma_1 = \gamma_3 = 0$ is the case when IPI is absent. It is clear from the figure that our method performs well in all situations. When the IPI is worse, the performance gap between our method and the fixed threshold method is smaller. In this situation, all the received signals are corrupted by the IPI and it is hard to recover the data from those signals.

Comparing the system performance between rectangular and hexagonal coordinate systems with similar degrees of ISI, we realize that the least square equalization achieves better improvement in the hexagonal system over that in the rectangular system. The performance improvement with increasing SNR in the hexagonal system is slightly better than that for the rectangular system. When IPI and noise effects dominate, the least square equalization provides a significant improvement that is especially useful for the hexagonal coordinate system.

# Chapter 6 Extension of 3D Page-Oriented Optical Data Storage Systems

In this chapter, we explore the feasibility of using a multi-level (non-binary) format for data storage to further enhance the capacity. We apply least-square equalization methods to process the data and present some preliminary simulation results.

## 6.1 Multi-level Data Storage Systems

One technique to increase the data storage capacity of a system is to increase the number of signal levels encoded within a given spatial area (pixel) of the storage medium. For example, if we can reliably detect four stored levels rather than the usual two, we can encode $\log_2 4 = 2$ bits per pixel instead of one bit per pixel.

In a multi-level data storage system, there are at least two ways to encode multi-level signals. One is space encoding and the other is amplitude encoding. We concatenate these encoding schemes and form the multi-level encoder as shown in Figure 6.1.

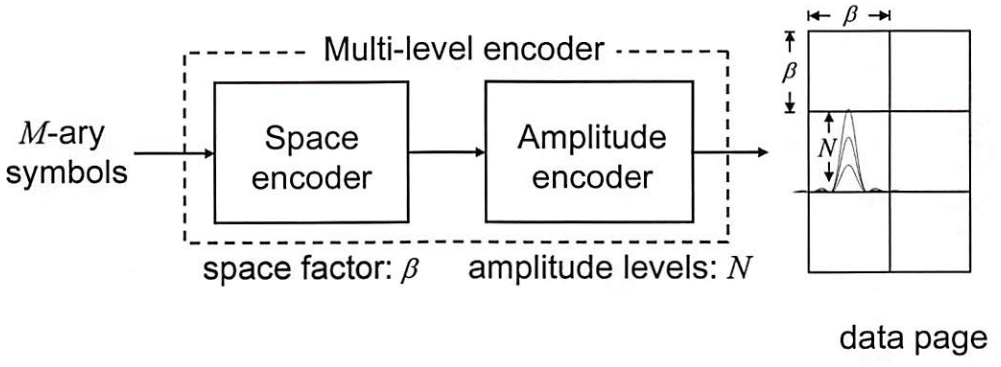



Figure 6.1. Block diagram of the multi-level encoder.

We normalize the pitch between regular size pixels in the $x$ and $y$ dimensions. For the system with space encoding, we define a one-dimensional spatial scale factor $\beta$ that determines the size of pixels and detectors. To encode more bits than the regular pixel size system in the same unit area, we choose the value of $\beta$ to be less than 1, i.e. $0 < \beta \leq 1$. For $\beta = 1$, the pixel pitch is the same size as the regular size detectors. The amplitude encoder encodes the signals into $N$ different intensity levels to be recorded on the data pages. We assume that these $N$ amplitude levels are equally probable and are spaced over the interval [0, 1]. Hence, the adjacent amplitude levels are separated by $1/(N-1)$.

To use multi-amplitude encoding efficiently, we choose $N$ such that $N = 2^k$, where each pixel conveys $k$ bits of information. In the planar data storage system, the bit packing density is defined as

$$\text{Bit packing density} = \frac{\text{Number of bits}}{\text{Unit area}}. \tag{6.1}$$

The multi-level encoder keeps the same pixel spatial packing density from the input to the output for consistency. Therefore, for the storage of $M$-ary symbols, the bit packing density is

$$\text{Bit packing density} = \log_2 M = \frac{\log_2 N}{\beta^2}. \tag{6.2}$$

Then we have

$$M = N^{\beta^{-2}}. \tag{6.3}$$

Using the simplest multi-level system with four levels ($M$=4) as an example, we have two different choices to achieve the same data capacity. The first choice is to use detectors with a pixel pitch of the same size, but with four different amplitude values ($N$=4). The other choice is to use smaller detectors and pixel pitch with binary amplitude values ($N$=2) as before. In Figure 6.2, a cross-section of the point spread function along with two different detector sizes is shown. The point spread function

$$h_{dl}(x, y) = \frac{1}{b_s^2}\mathrm{sinc}^2\left(\frac{x}{b_s}, \frac{y}{b_s}\right) \tag{6.4}$$

used in the rectangular coordinate system is shown at the top of the figure. The top detector array (regular detector) is the same unit diameter and unit pixel array defined in Chapter 2. The lower detector array (small detector) is a fraction ($\beta$) of the unit diameter and unit pixel array compared to the regular detector where $0 < \beta \leq 1$. For $M$ amplitude levels in the regular detector system, the $\beta$ value is $1/\sqrt{\log_2 M}$ as calculated from Eq. (6.3) in the small detector system to achieve the same data packing density.

As the controlling parameter $b_s$ increases, the main lobe width of the point spread function increases as well. This severe inter-symbol interference will have more effect on the small detector than the regular detector. However, in the small detector,

we only need to distinguish two levels of data amplitude. On the other hand, in the regular detector, we need to identify more levels of data amplitude.

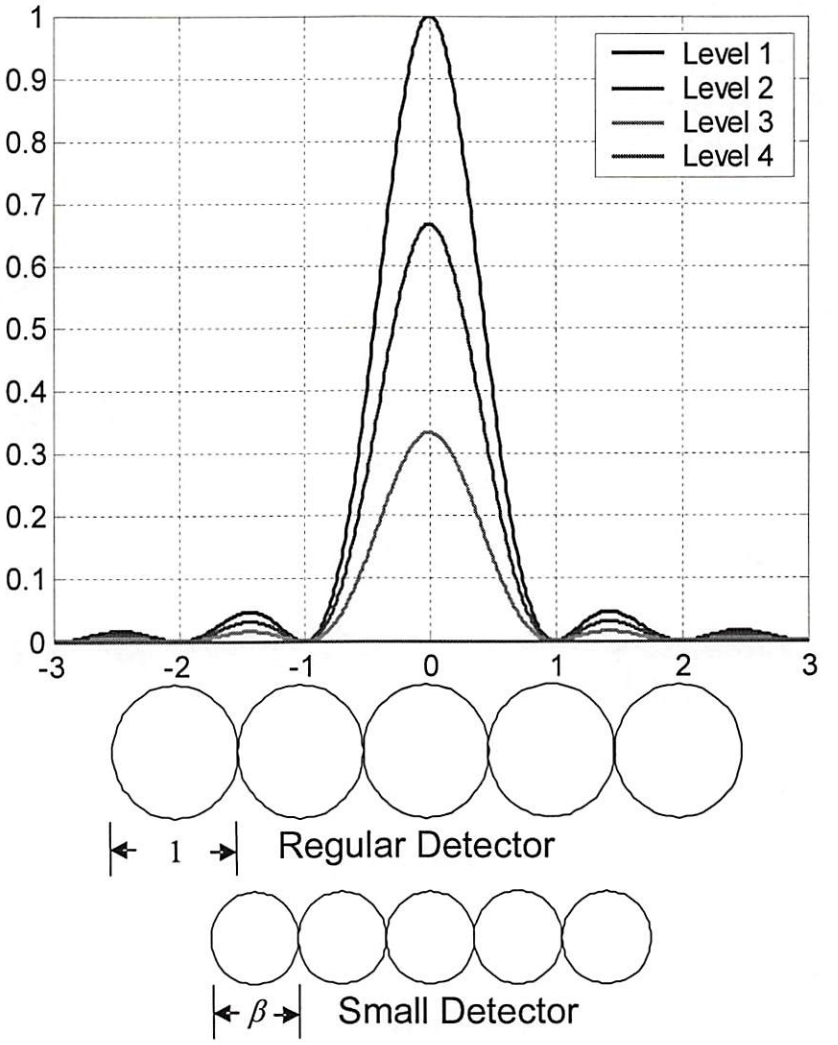



Figure 6.2. Comparison between regular detector size and small detector size.

In Figure 6.2, the amplitude levels are separated equally by $1/(N-1)$, where $N$=4. We choose amplitude values as 1, $2/3$, $1/3$, and 0. For these amplitude values, we

will have trouble determining the data if the center pixel suffers from interference with one sixth of its energy. Therefore, the noise margin is $\frac{1}{6}$ in four levels. On the contrary, when we use two levels in the small detector scaled down by $\sqrt{2}$ , this problem exists only if half of the energy is added or subtracted from the center pixel. The example is shown in Figure 6.3.

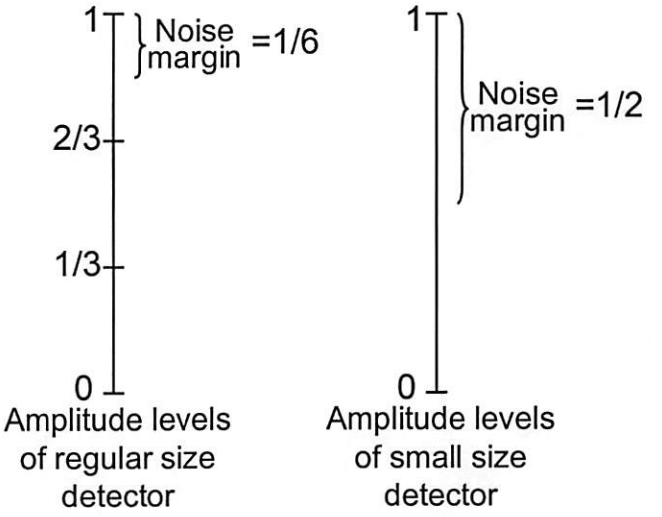



Figure 6.3. Comparison of noise margin between regular size detector and small size detector.

There are three different sources which cause the error in determining the amplitude levels. One is additive noise. It includes all the noise that is incurred by the readout system and is usually modeled as zero-mean white noise. Thus the effect of the white noise is both additive and subtractive.

The second source is the interference generated by the point spread function of the readout system. Depending on the situation, the interference can be either additive or

subtractive. In an extreme case, when there is some extra energy infused by the surrounding mark pixels to the central space pixel and it exceeds the noise margin, an error occurs. Since this interference increases the received signal energy, it is classified as additive interference. There is another error possibility when the energy spreads out and makes the received signal below the noise margin. Therefore, it is the subtractive interference. Two extreme cases of errors are shown in Figure 6.4.

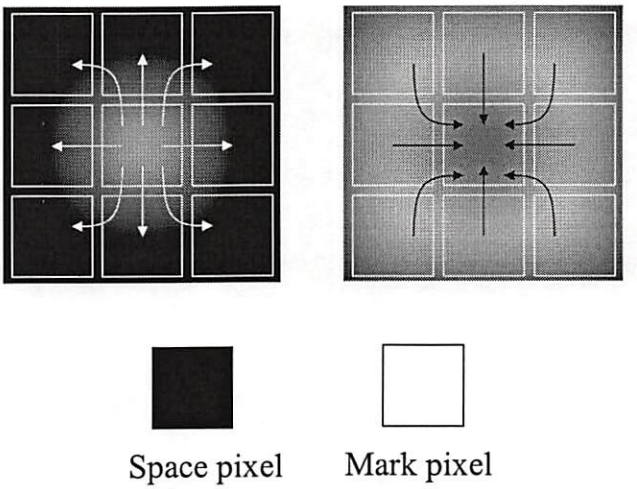



Figure 6.4. Two extreme cases of additive and subtractive interference.

Although the noise and interference can be additive or subtractive, there are two situations in which they do not affect the readout system. For pixels having a maximum amplitude value, additive noise or interference will not introduce errors since the amplitude saturated. On the contrary, no erroneous results will be made if subtractive noise or interference is applied to pixels having a minimum amplitude value. Except these two situations, multi-amplitude signals are limited by the noise margin and prone to both additive and subtractive noise and interference.

In general, for multi-amplitude data storage systems with equally spaced $N$-levels, the noise margin is

$$\text{Noise margin} = \frac{1}{2(N-1)}. \tag{6.5}$$

Given the noise margin, we can define the conditions when errors occur in multilevel amplitude systems for detector of any size. We then compare the performance between multi-level systems with regular detectors and binary systems with small detectors.

In the absence of noise and inter-page interference, we can compare various amplitude encoding methods using calculation or simulation. However, since there are many different data patterns to be considered, an analytical procedure to find the decision point is tedious, and is a complicated function of system parameters such as the PSF size and the type of coordinate systems. Instead, we use simulation method to reach the results. To have a fair comparison, we compare two systems based on the same packing density as in Eq. (6.1). The results shown in Table 6.1 are obtained from systems with rectangular coordinates.

| | Maximum $b_s$ value without error |
|---|---|
| Four amplitude levels with regular detector | 0.8894 |
| Binary amplitude levels with small detector scaled down to $1/\sqrt{2}$ of the regular detector | 0.9581 |

Table 6.1. Comparison between multi-amplitude level and binary level systems with rectangular coordinates.

From the results, it suggests that the system using small detectors with binary levels is more resistant to interference than the system with regular detectors. Not only is the noise margin larger, but also the signal is degraded by either additive or subtractive interference and noise. Since the pixels in the small detector system are either maximum or minimum amplitude level, they only suffer from either subtractive or additive interference. In contrast, there are two levels in the regular detector system that are not at maximum or minimum amplitude level. They are affected by both the subtractive and additive interferences. Therefore, we would expect that the system with multi-amplitude levels and regular size detectors performs worse in most situations.

### 6.2 Noise Effects and Simulation

In practice, the readout system is never noise-free. Therefore, we include additive white Gaussian noise in the simulation. For simplicity, we consider the system with four levels using regular detector and the system with two levels using a small detector. We examine the system performance in both rectangular and hexagonal

coordinate systems. The results are shown in Figure 6.5 through Figure 6.8 as a function of different degrees of interference represented by the blur factor.

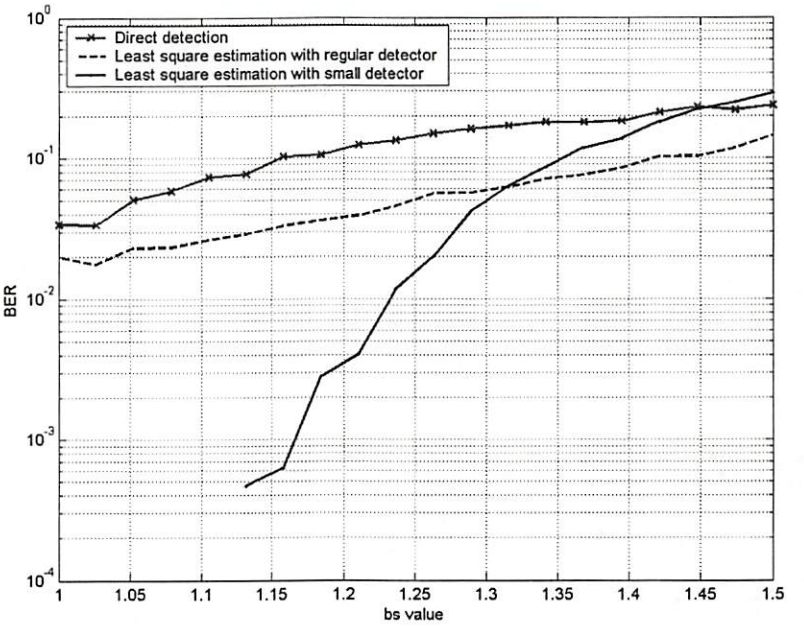



Figure 6.5. Comparisons of rectangular coordinate multi-level systems with moderate AWGN $(\sigma = 0.03)$ and without IPI effect $(\gamma_1 = 0)$.

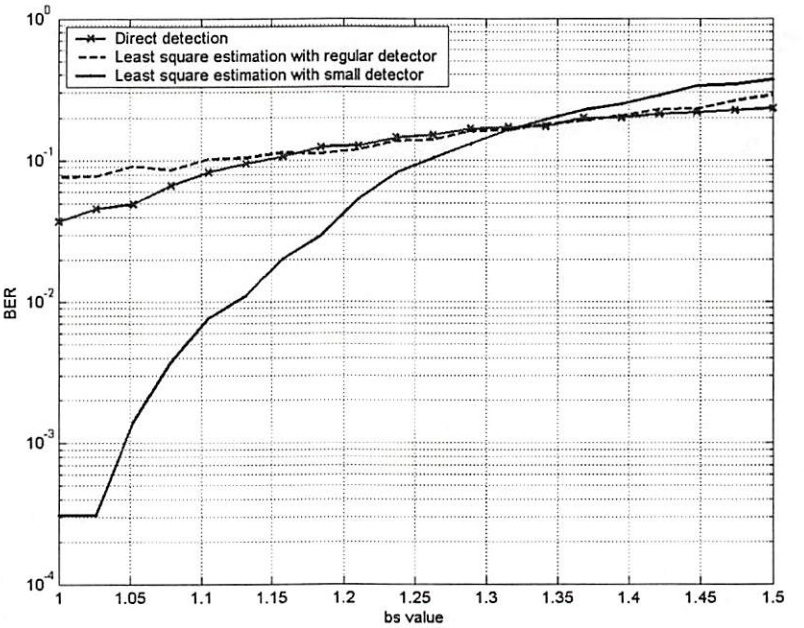



Figure 6.6. Comparisons of rectangular coordinate multi-level systems with severe AWGN $(\sigma = 0.05)$ and without IPI effect $(\gamma_1 = 0)$.

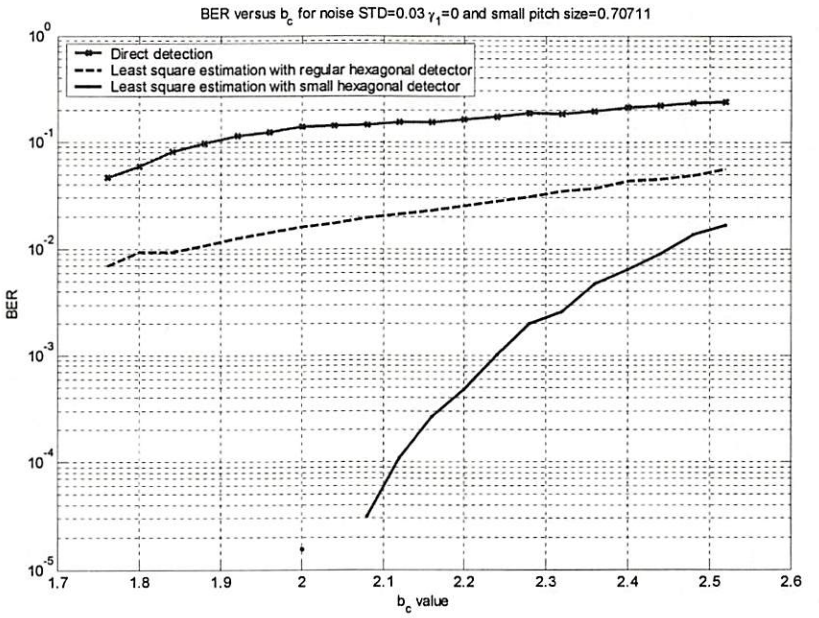



Figure 6.7. Comparisons of hexagonal coordinate multi-level systems with moderate AWGN $(\sigma = 0.03)$ and without IPI effect $(\gamma_1 = 0)$.

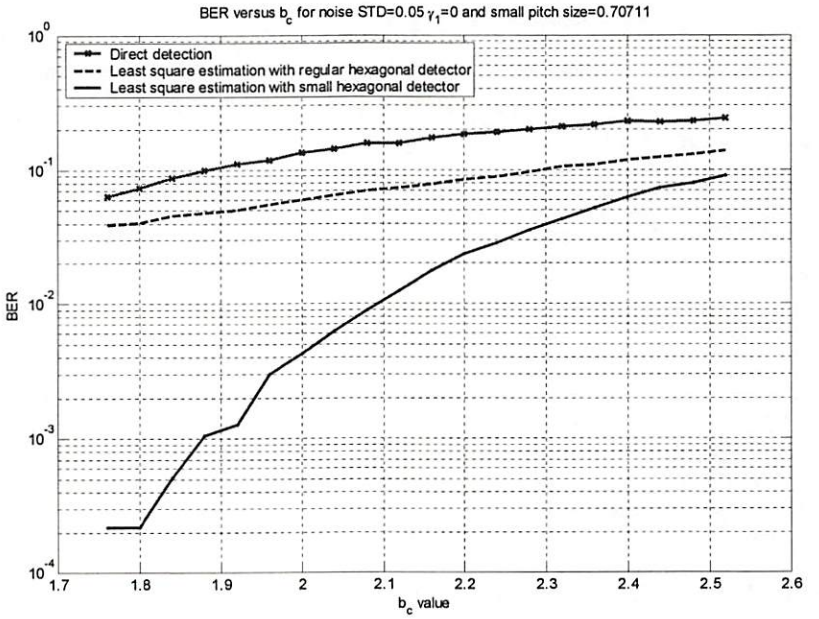



Figure 6.8. Comparisons of hexagonal coordinate multi-level systems with severe AWGN $(\sigma = 0.05)$ and without IPI effect $(\gamma_1 = 0)$.

From the simulation results, we notice that the system with four levels and regular detectors performs badly due to the noise. On the other hand, the system with two levels and small detectors is relatively robust, even with noise. One explanation for this is that the additive noise has a greater effect on the pixel amplitude and the system with smaller noise margin degrades significantly.

However, when the value of $b_s$ is high, the four level system outperforms the two level system in the rectangular coordinate system, since the influence of the interference produced by PSF is more severe than that of the noise. The PSF becomes wider and extends the interference to obscures more detectors. Hence, the small detector system receives more interference and the system performance deteriorates. Nevertheless, similar situations do not happen in the hexagonal coordinate system.

It should be noted that the above simulations assume that the inter-page interference is absent, i.e. $\gamma_1 = \gamma_3 = 0$. However, we will show that our least square equalization method still works even with IPI effects in the last section of this chapter.

### 6.3 Preliminary Simulation Results for Multi-Level Data Storage Systems

In this section, we apply our least square equalization method to the multi-level system with the presence of the inter-page interference. We use the Golay code to encode the amplitude levels and calculate the bit-error-rate [7]. The encoding scheme is shown in Table 6.2.

| Level | Golay code |
|---|---|
| 1 | 10 |
| 2 | 00 |
| 3 | 01 |
| 4 | 11 |

Table 6.2. Example of Golay codes for four amplitude level system.

With the Golay code, the number of error bits is 1 when the received signal is detected erroneously as one of its neighboring levels. The number of error bits is 2 if the signal is detected as a signal two levels away from its correct value. Therefore, using Golay code is a fair way to compare systems with different number of levels using bit-error-rate. We apply our least square equalization method to the two level system as well as the four level system. The results with different degree of IPI are shown in Figure 6.9 through Figure 6.12 for both rectangular and hexagonal coordinate systems. According to Table 5.2, the blur factors are chosen such that the PSF produces comparable interference in both systems. Since we select blur factors that exceed the Rayleigh resolution limit, the interference is considered to be moderate.

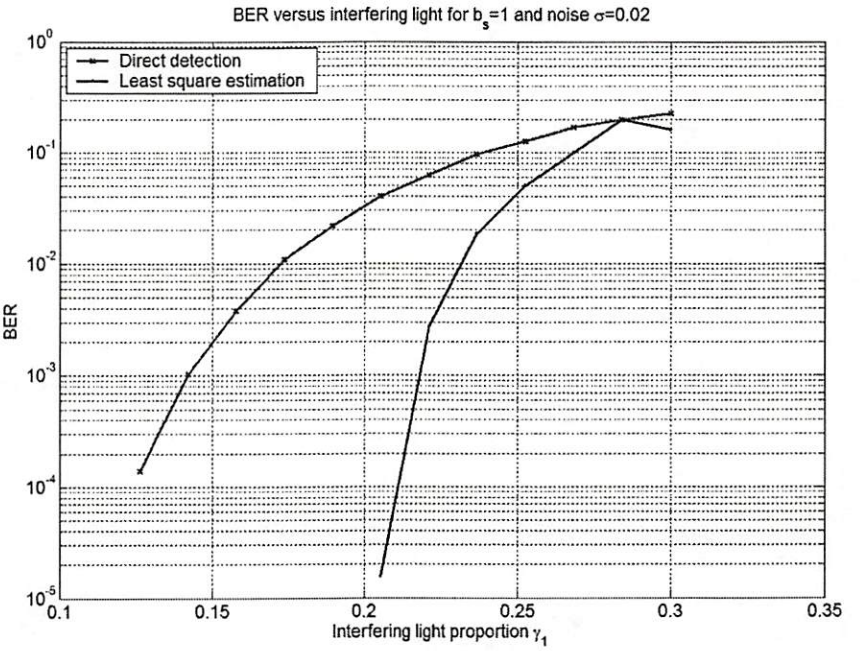



Figure 6.9. Least square equalization result for rectangular coordinates 2 levels regular detector system with moderate ISI ($b_s$=1) and moderate noise $(\sigma = 0.02)$.

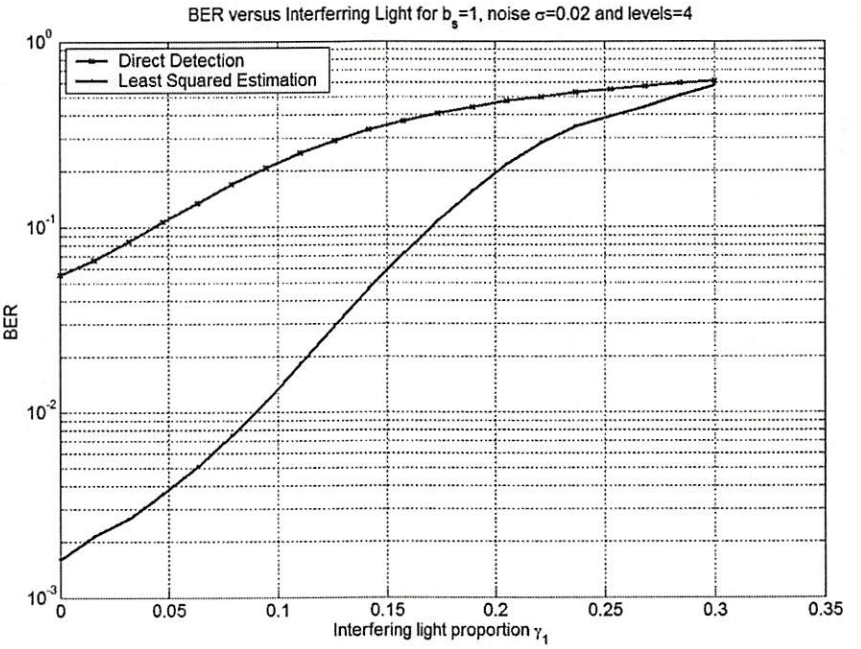



Figure 6.10. Least square equalization result for rectangular coordinates 4 levels regular detector system with moderate ISI ($b_s$=1) and moderate noise $(\sigma = 0.02)$.

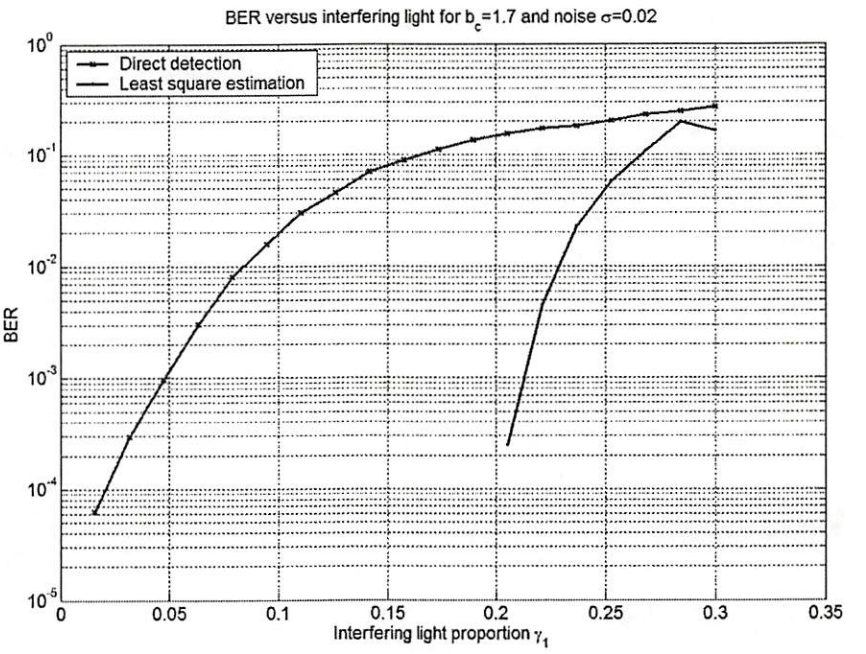



Figure 6.11. Least square equalization result for hexagonal coordinates 2 levels regular detector system with moderate ISI ($b_c$=1.76) and moderate noise $(\sigma = 0.02)$.

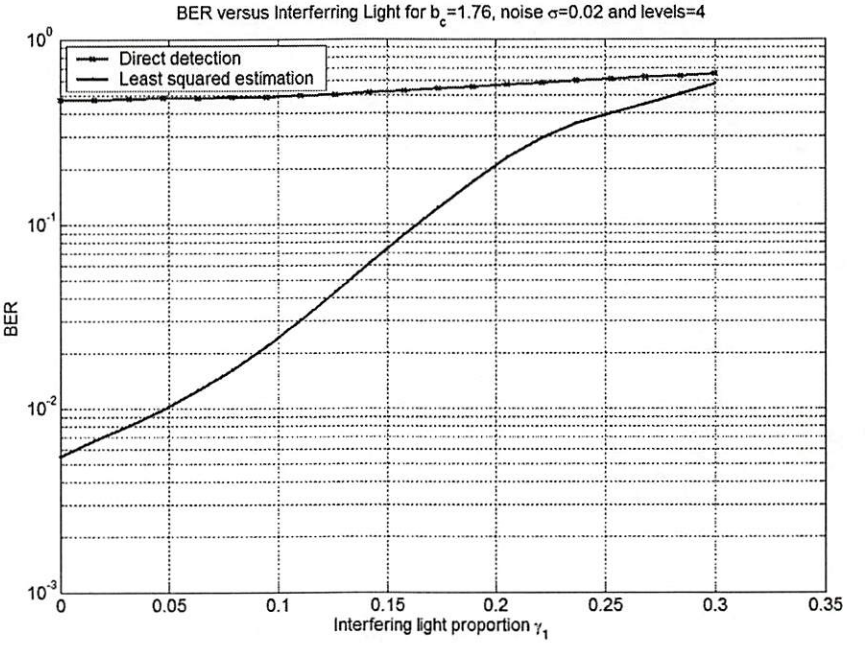



Figure 6.12. Least square equalization result for hexagonal coordinates 4 levels regular detector system with moderate ISI ($b_c$=1.76) and moderate noise $(\sigma = 0.02)$.

In the above figures, the bit-error-rates of both systems are plotted as a function of the inter-page interference (the horizontal axis). The bottom curves show that our method can provide consistent better results for multi-level system with different degree of IPI.

# Chapter 7 Variable Threshold Method

In this chapter we describe a new method to determine dynamically the threshold of each pixel under ISI and IPI effects. The advantage of this method over the conventional binary threshold method is that it counteracts both the ISI and IPI effects and adjusts the threshold locally. The low computational complexity of this method is especially useful for 3D PODS systems.

## 7.1 Fixed Threshold Method

In traditional data storage systems, signal detection using a fixed threshold is widely used because of its simplicity. The fixed threshold mechanism is employed in the readout system shown in Figure 7.1.

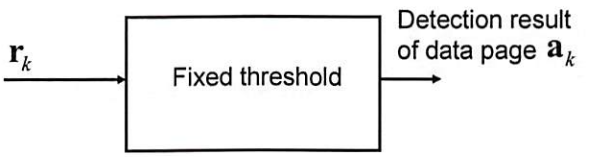



Figure 7.1. Diagram of fixed threshold detection.

Here we use the notation as in Eq. (4.1), with $\mathbf{r}_k$ denoting the received signal and $\mathbf{a}_k$ denoting the data page at time index $k$. The fixed threshold provides a simple and reliable way to distinguish the received mark signals from received space signals. For the binary signaling data storage system, the fixed threshold is set according to the histogram of received signal intensity shown in Figure 7.2.

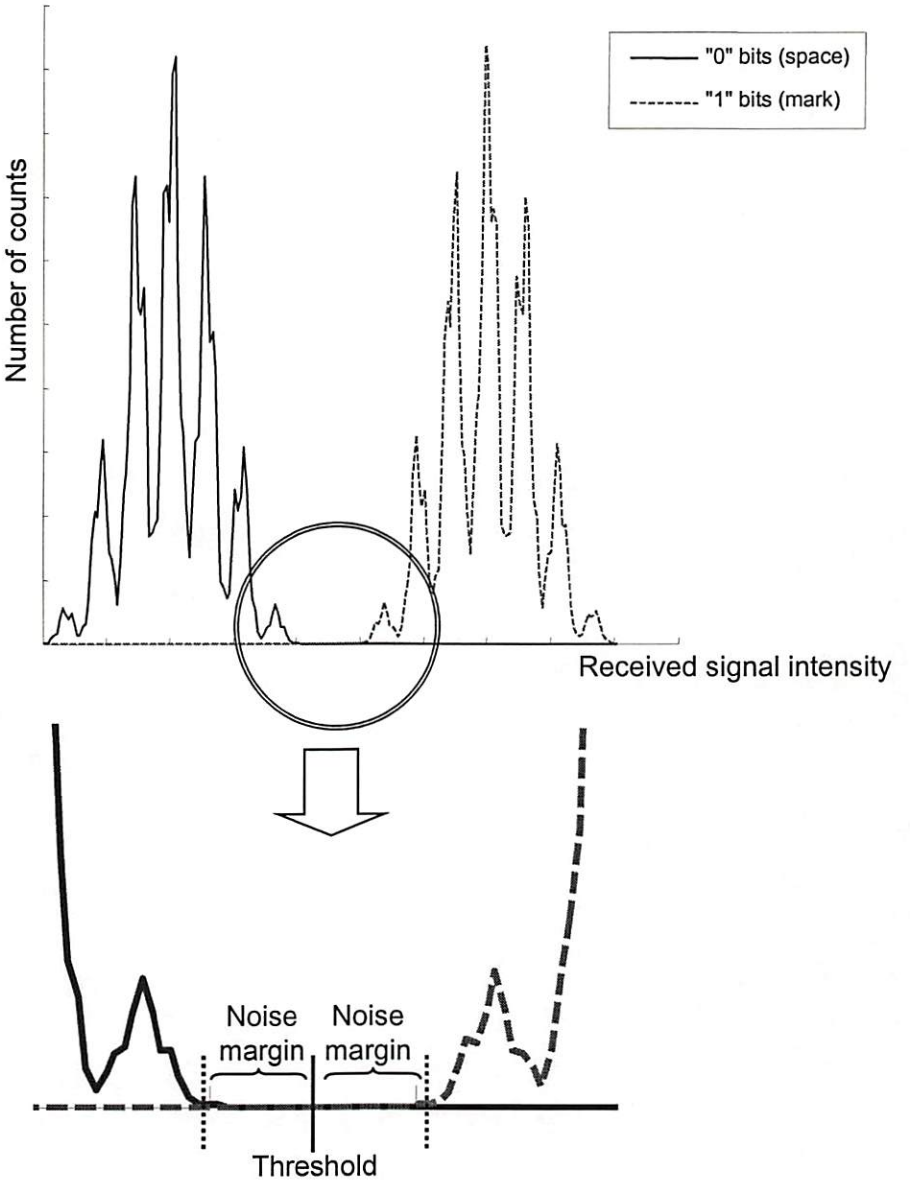



Figure 7.2. Histogram of received signal intensity for $b_s$=1 and $\gamma_1 = 0.1$.

Figure 7.2 shows an example of setting the fixed threshold. The solid line represents "0" (space) bits and the dashed line represents "1" (mark) bits. The received signal intensity is sampled from a noise-free system with $b_s$=1 and $\gamma_1 = 0.1$, hence this

system only suffers moderate ISI and IPI effects. Since the received mark intensity and space intensity each occupies a defined region on the horizontal axis of the histogram, and the "0" and "1" bits are assumed equally probable, the fixed threshold is set at halfway between their limits. For received signal intensities smaller than the threshold, it classifies the data as a space or "0", otherwise it classifies the data as a mark or "1" as described by

$$\hat{a}_k[i,j] = \begin{cases} 0, & if\ r_k[i,j] < T \\ 1, & if\ r_k[i,j] > T \end{cases}, \tag{7.1}$$

where $T$ is the threshold. The error occurs when $a_k[i,j]$ differs from $\hat{a}_k[i,j]$.

The region between the largest received space signal and the threshold or the smallest received mark signal and the threshold is called the noise margin. If the additive noise level is lower than the noise margin, the fixed threshold method can still detect the signal correctly. On the other hand, if the additive noise level is larger than the noise margin, the fixed threshold method is highly likely to fail.

The value of the noise margin is determined by ISI and IPI effects. When these effects become worse, the noise margin is reduced to zero. At this point, the histograms of received mark signals and that of received space signals overlap each other. Therefore, the fixed threshold method generates erroneous results even under noise free conditions.

In 3D page-oriented data storage systems, the ISI or IPI effects cause the fixed threshold method to generate unreliable detection results. Hence, we must resort to other detection methods to overcome both the ISI and IPI effects in high density and capacity page-oriented data storage systems.

### 7.2 Variable Threshold Method

From the 3D page-oriented data storage system model introduced in Chapter 3, we analyze the components of received signals. There are three major components: the direct measurement, the interference from the neighbors and the additive noise. By identifying these three components, we apply the variable threshold method to improve the system performance.

We rewrite the detector output signal $r_k[i,j]$ from Eq. (3.1) as

$$r_k[i,j] = \int_{j-\delta/2}^{j+\delta/2}\int_{-\delta/2}^{i+\delta/2}\gamma_1 a_{k-1}(x,y)\otimes h_1(x,y)dxdy + \int_{j-\delta/2}^{j+\delta/2}\int_{-\delta/2}^{i+\delta/2}\gamma_2 a_k(x,y)\otimes h_2(x,y)dxdy + \int_{j-\delta/2}^{j+\delta/2}\int_{-\delta/2}^{i+\delta/2}\gamma_3 a_{k+1}(x,y)\otimes h_3(x,y)dxdy + v_k[i,j] \quad , \tag{7.2}$$

where $\delta$ is the linear fill factor, and $\otimes$ denotes the discrete 2D convolution operator. For simplicity, we assume the fill factor is 1 and the extent of each PSF is $[-M \sim M, -N \sim N]$. Then we express all the signals in discrete spatial indices and represent the discrete 2D convolution operator as

$$r_k[i,j] = \gamma_1 \sum_{m=-M}^{M} \sum_{n=-N}^{N} a_{k-1}[m,n]\, h_1[i-m, j-n] + \gamma_2 \sum_{m=-M}^{M} \sum_{n=-N}^{N} a_k[m,n]\, h_2[i-m, j-n]$$
$$+ \gamma_3 \sum_{m=-M}^{M} \sum_{n=-N}^{N} a_{k+1}[m,n]\, h_3[i-m, j-n] + v_k[i,j] \tag{7.3}$$

We then rewrite Eq. (7.3) as

$$r_k[i,j] = \gamma_2 a_k[i,j]\, h_2[0,0] + \gamma_1 \sum_{m=-M}^{M} \sum_{n=-N}^{N} a_{k-1}[m,n]\, h_1[i-m, j-n]$$
$$+ \gamma_2 \sum_{m=-M}^{M} \sum_{\substack{n=-N \\ \{m,n\} \neq \{i,j\}}}^{N} a_k[m,n]\, h_2[i-m, j-n] + \gamma_3 \sum_{m=-M}^{M} \sum_{n=-N}^{N} a_{k+1}[m,n]\, h_3[i-m, j-n].$$
$$+ v_k[i,j] \tag{7.4}$$

By looking at the detector output signal $r_k[i,j]$ above, we identify the three major components of the received signal. The first term on the right hand side of Eq. (7.4) is the direct measurement. It contains partial information about the data that we want to retrieve. The last term on the right hand side of Eq. (7.4) is the additive noise. The remaining three terms on the right hand side constitute interference from the neighbors.

The direct measurement is the signal intensity produced by the data stored on the medium after the retrieving process. In that process, the signal energy tends to be spread out in space by the readout subsystem and only part of the signal energy is received by the detector. Hence, we call it the direct measurement. For a space pixel or the "0" bit, the direct measurement is always zero.

The second element is the interference signal generated from the neighbors. It has two components: ISI and IPI. As we mentioned in the previous chapter, ISI and IPI effects always exist in the readout subsystem and cause the interference between pixels. The interference produced from the surrounding neighbors within the same data page is referred to as the ISI, while the interference from other adjacent data pages is the IPI. From Eq. (7.4), the ISI is

$$\mathrm{ISI} = \gamma_2 \sum_{m=-M}^{M} \sum_{\substack{n=-N \\ \{m,n\} \neq \{i,j\}}}^{N} a_k[m,n]\ h_2[i-m, j-n], \tag{7.5}$$

and the IPI is

$$\mathrm{IPI} = \gamma_1 \sum_{m=-M}^{M} \sum_{n=-N}^{N} a_{k-1}[m,n]\ h_1[i-m, j-n] + \gamma_3 \sum_{m=-M}^{M} \sum_{n=-N}^{N} a_{k+1}[m,n]\ h_3[i-m, j-n]. \tag{7.6}$$

The last element in the received signal in Eq. (7.4) is the additive noise. It accounts for all measurement errors incurred during the readout process. Therefore, the additive noise can be analyzed using statistical methods.

The variable threshold method uses equalization to get an estimate of the bit value in the neighborhood of $a_k[i,j]$, and then estimate their contribution to ISI and IPI. This information is used to vary the decision threshold at each $a_k[i,j]$. Thus, the variable threshold method attempts to reduce the interference from the surrounding neighbors in three dimensions and produce more reliable detection results.

A block diagram of the variable threshold method is shown in Figure 7.3.

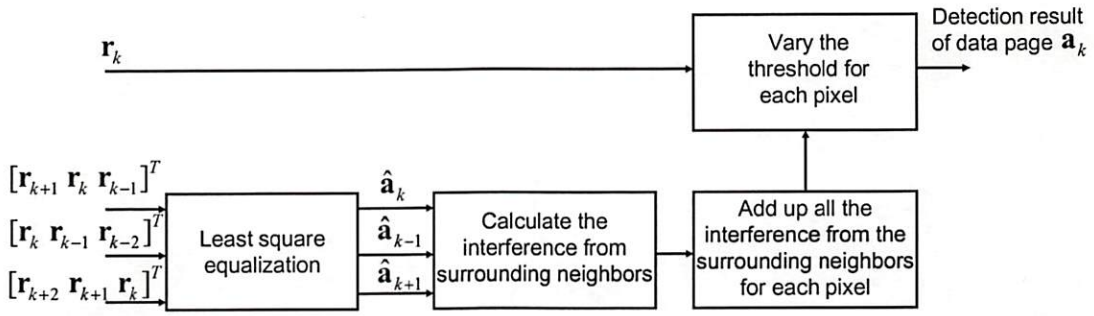



Figure 7.3. Block diagram of variable threshold method.

Here $\mathbf{r}_k$ is the received signal and $\mathbf{a}_k$ is the data page at time index $k$ as before. Least square equalization is used first to provide an estimate of the data. Then, using knowledge of the 3D point spread function we introduced in Chapter 3, we adjust the threshold value. Since the interference comes from all three dimensions, we need estimates not only from within the current data page $\mathbf{a}_k$ , but also from the adjacent data pages $\mathbf{a}_{k-1}$ in front of and $\mathbf{a}_{k+1}$ in back of $\mathbf{a}_k$. Therefore, the least square equalization performs the following calculations as Eq. (4.6).

$$\begin{cases} \hat{\mathbf{a}}_{k-1} = \mathbf{C}_1\mathbf{r}_k + \mathbf{C}_2\mathbf{r}_{k-1} + \mathbf{C}_3\mathbf{r}_{k-2} \\ \hat{\mathbf{a}}_k = \mathbf{C}_1\mathbf{r}_{k+1} + \mathbf{C}_2\mathbf{r}_k + \mathbf{C}_3\mathbf{r}_{k-1} \\ \hat{\mathbf{a}}_{k+1} = \mathbf{C}_1\mathbf{r}_{k+2} + \mathbf{C}_2\mathbf{r}_{k+1} + \mathbf{C}_3\mathbf{r}_k \end{cases}, \tag{7.7}$$

where matrices $\mathbf{C}_1$, $\mathbf{C}_2$ and $\mathbf{C}_3$ are derived from the pseudoinverse matrix $\underline{\mathbf{D}}^+$ as in Eqs. (4.14) (4.15) (4.16).

Based on these estimated pixel values, we calculate their interference contribution to the neighbors. The extent of the interference contribution to the neighbors for one

data mark is the extent of the PSF, i.e. $[-M \sim M, -N \sim N]$. Therefore, only the neighbors within $[-M \sim M, -N \sim N]$ are calculated for their interference contribution. We assume that the received signal of one data pixel is not affected by the surrounding neighbors outside that range. Hence the estimate of the interference contribution for each pixel is calculated from

$$f_k[i,j] = \gamma_1 \sum_{m=-M}^{M} \sum_{n=-N}^{N} a_{k-1}[m,n]\, h_1[i-m, j-n] + \gamma_2 \sum_{m=-M}^{M} \sum_{\substack{n=-N \\ \{m,n\} \neq \{i,j\}}}^{N} a_k[m,n]\, h_2[i-m, j-n] + \gamma_3 \sum_{m=-M}^{M} \sum_{n=-N}^{N} a_{k+1}[m,n]\, h_3[i-m, j-n], \tag{7.8}$$

where $f_k[i,j]$ is the interference contribution estimate for pixel at location $[i,j]$. Finally, we add up all the interference contributions from neighbors in three dimensions and determine the threshold value for each pixel. For the binary signaling system, i.e. $a_k[i,j]=1$ or $a_k[i,j]=0$, the direct measurement is

$$\gamma_2 a_k[i,j] h_2[0,0] = \begin{cases} 0, & \text{if } a_k[i,j]=0 \\ \gamma_2 h_2[0,0], & \text{if } a_k[i,j]=1 \end{cases}. \tag{7.9}$$

We set the threshold value at halfway between the space received signal and the mark received signal similar to the fixed threshold method. Therefore, the threshold value for the binary signaling system is set to be

$$T_k[i,j] = \frac{\gamma_2 h_2[0,0]}{2} + f_k[i,j]. \tag{7.10}$$

Since the interference contribution $f_k[i,j]$ is different for each pixel, the threshold value $T_k[i,j]$ varies for each pixel.

From Eqs. (7.4) through (7.6), there are three major elements in the received signal. In effect, we subtract an estimate of the interference from the surrounding pixels, only two other components are left in the received signal. The threshold is determined by the direct measurement part. The direct measurement of a mark pixel is calculated based on *a priori* knowledge of the 3D point spread function. Because the direct measurement of space pixels is always zero, we set the threshold value at the midpoint between the smallest mark direct measurement and zero.

During detection, the threshold value changes for each pixel because the interference from the surroundings is different for each. This method removes the estimated interference from all the surrounding neighbors of a pixel in three dimensions, and adjusts the local decision threshold. Based on this point of view, we can treat the variable threshold method as a variant to the fixed threshold method. Its strength is to reduce both the ISI and IPI effects using an algorithm with low complexity.

### 7.3 Simulation Results

We performed the simulation using both rectangular grid detectors and hexagonal grid detectors. The simulation results based on rectangular array detectors are shown in Figure 7.4 to Figure 7.11, while the results of hexagonal array detectors are shown in Figure 7.12 to Figure 7.19. These figures are given in order of increasing ISI effect.

Every four figures have the same amount of additive noise. We choose four different PSF parameters $b_s$ and $b_c$ such that the corresponding ISI effect on the pair of $b_s$ and $b_c$ are approximately the same.

For comparison, we show the simulation results with and without the variable threshold method. From these figures, it shows that the variable threshold method improves the system performance especially when the ISI effect is dominant. The variable threshold method has similar improvement on both rectangular and hexagonal coordinate systems. However, the improvement on the rectangular array system is slightly more than that on the hexagonal array system.

We divide the IPI effects into three categories: low IPI ($\gamma_1 = 0 \sim 0.1$), moderate IPI ($\gamma_1 = 0.1 \sim 0.2$) and high IPI ($\gamma_1 = 0.2 \sim 0.3$). The variable threshold method works well in the region of low IPI and moderate IPI. When the IPI or the noise effects are severe, this method degrades because least square equalization cannot provide a reliable estimate of the data page. Therefore, this imperfect estimate causes erroneous calculation of the interference contribution, and affects the selection of the threshold value. In practical situations, the system would not be used under these conditions. In most cases, the variable threshold method benefits the system performance by gaining one-half to one order of magnitude improvement in the bit error rate (BER).

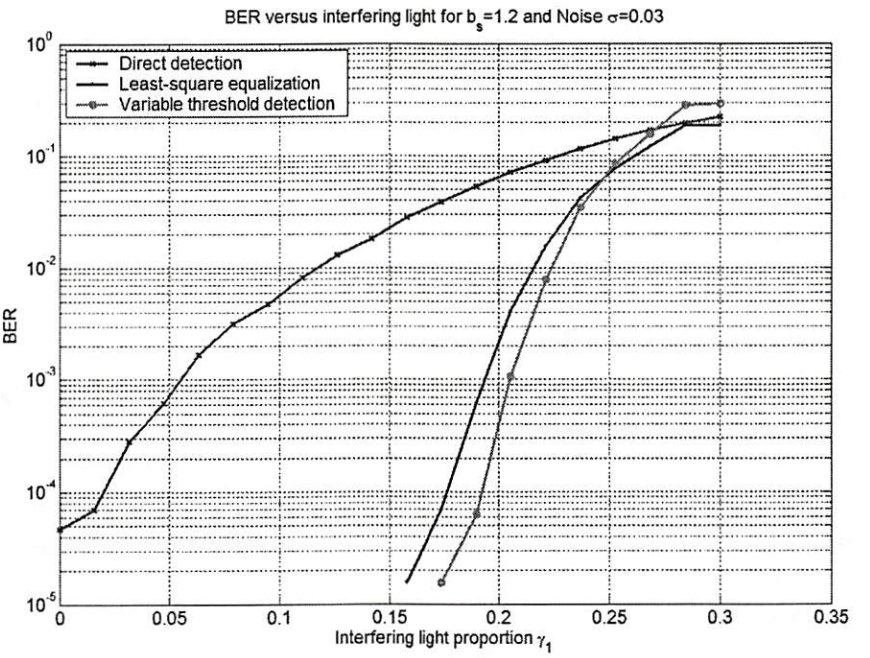



Figure 7.4. Variable threshold detection results for moderate ISI ($b_s$=1.2) and moderate noise ($\sigma$=0.03) on rectangular detector arrays.

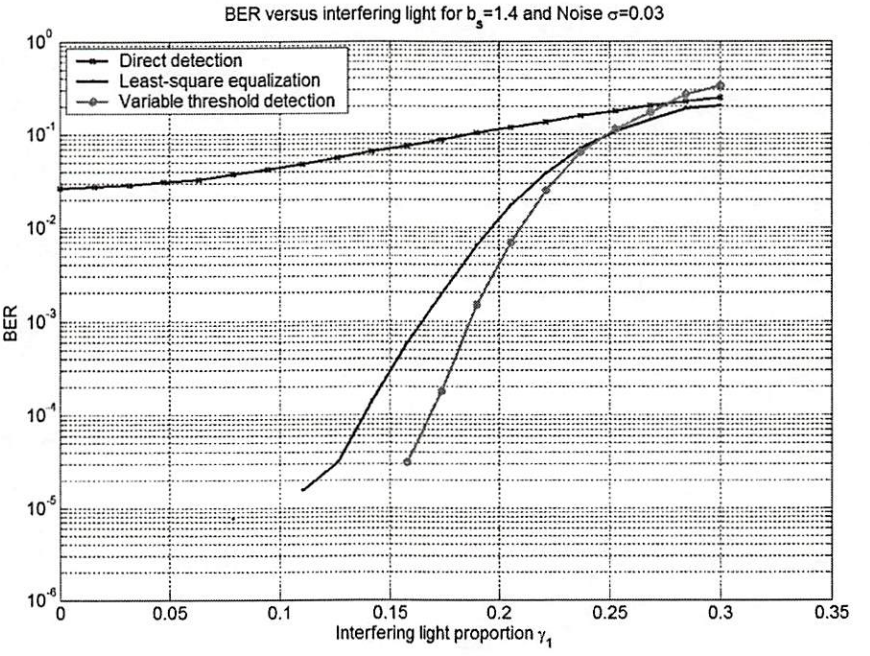



Figure 7.5. Variable threshold detection results for moderate ISI ($b_s$=1.4) and moderate noise ($\sigma$=0.03) on rectangular detector arrays.

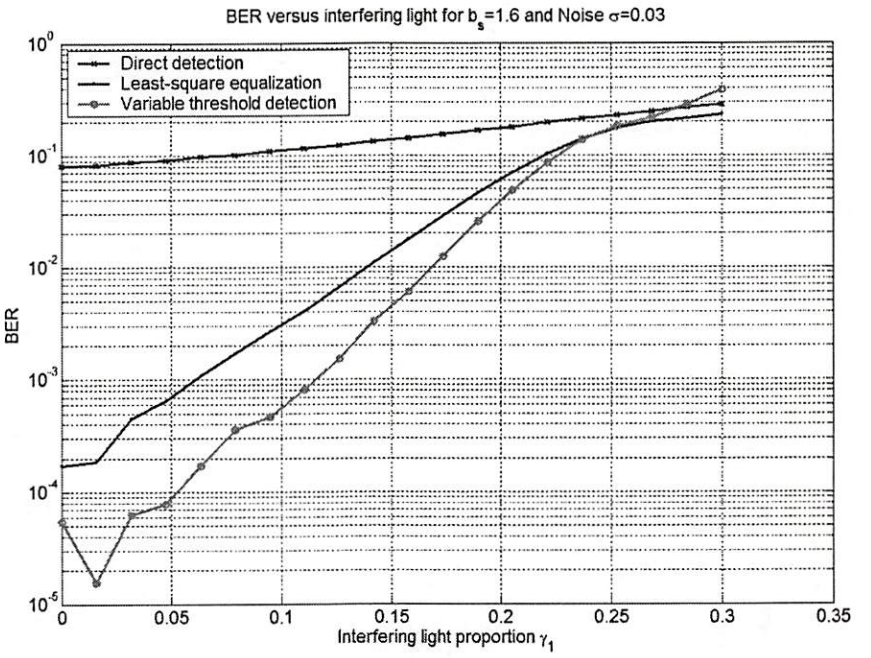



Figure 7.6. Variable threshold detection results for severe ISI ($b_s$=1.6) and moderate noise ($\sigma$=0.03) on rectangular detector arrays.

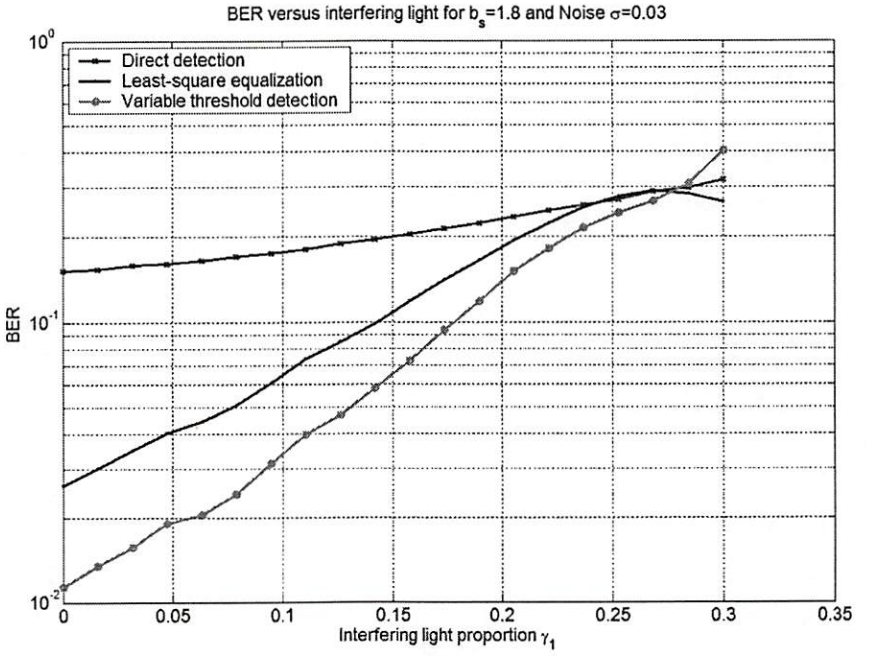



Figure 7.7. Variable threshold detection results for severe ISI ($b_s$=1.8) and moderate noise ($\sigma$=0.03) on rectangular detector arrays.

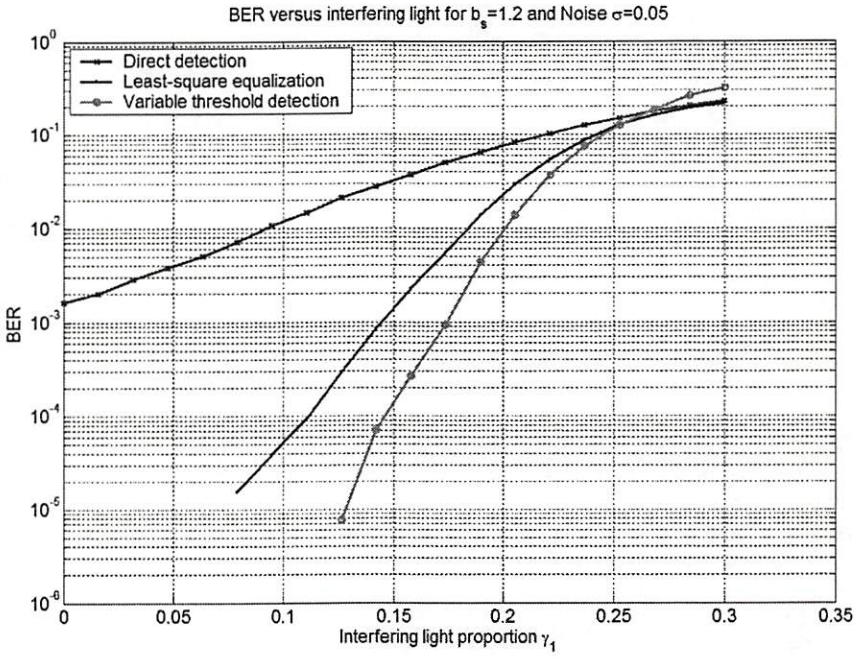



Figure 7.8. Variable threshold detection results for moderate ISI ($b_s$=1.2) and severe noise ($\sigma$=0.05) on rectangular detector arrays.

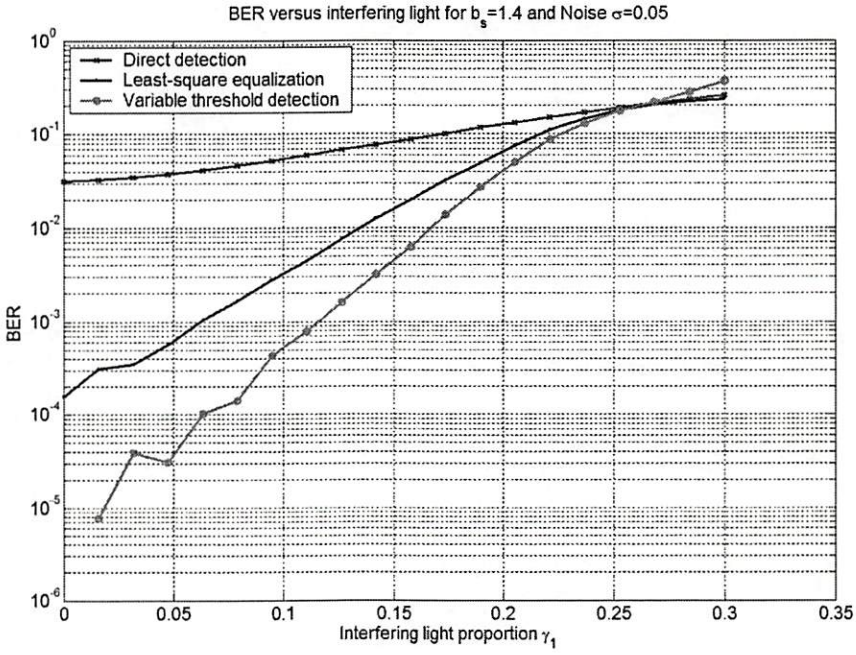



Figure 7.9. Variable threshold detection results for moderate ISI ($b_s$=1.4) and severe noise ($\sigma$=0.05) on rectangular detector arrays.

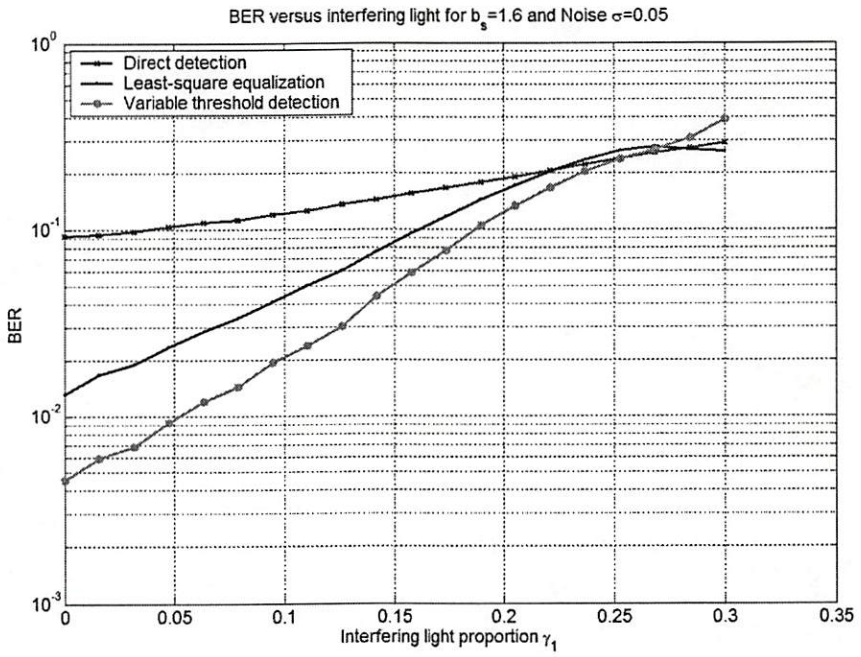



Figure 7.10. Variable threshold detection results for severe ISI ($b_s$=1.6) and severe noise ($\sigma$=0.05) on rectangular detector arrays.

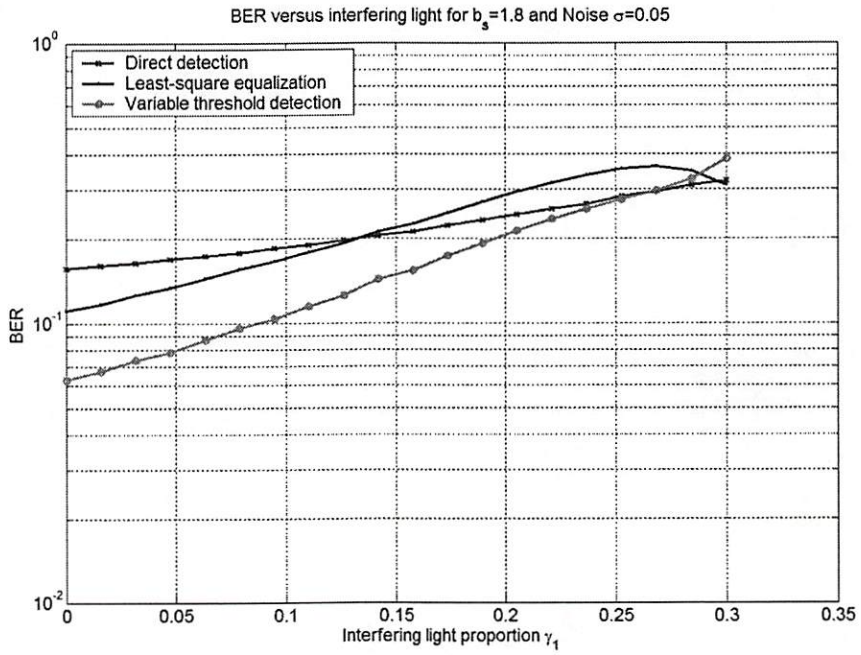



Figure 7.11. Variable threshold detection results for severe ISI ($b_s$=1.8) and severe noise ($\sigma$=0.05) on rectangular detector arrays.

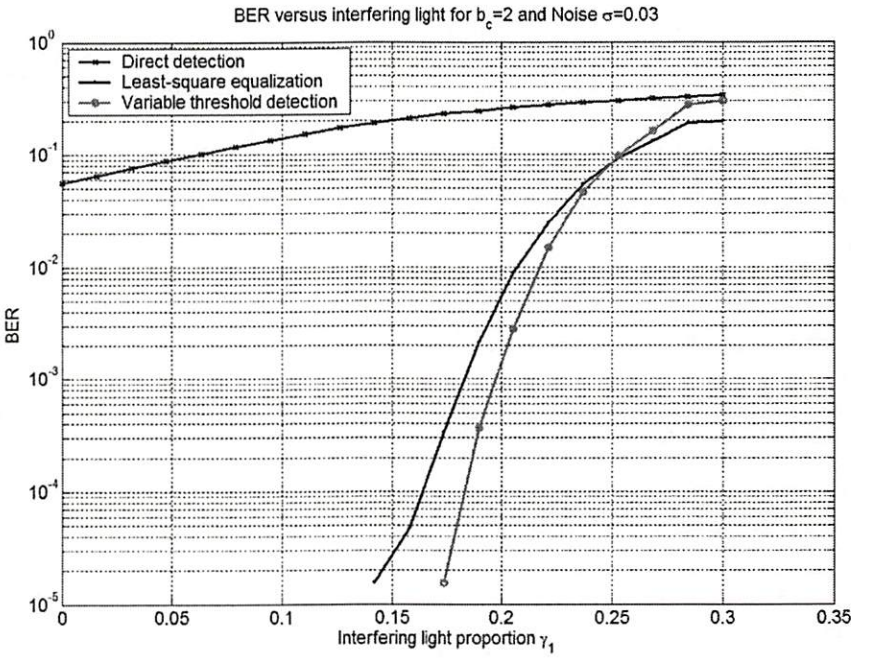



Figure 7.12. Variable threshold detection results for moderate ISI ($b_c$=2.06) and moderate noise ($\sigma$=0.03) on hexagonal detector arrays.

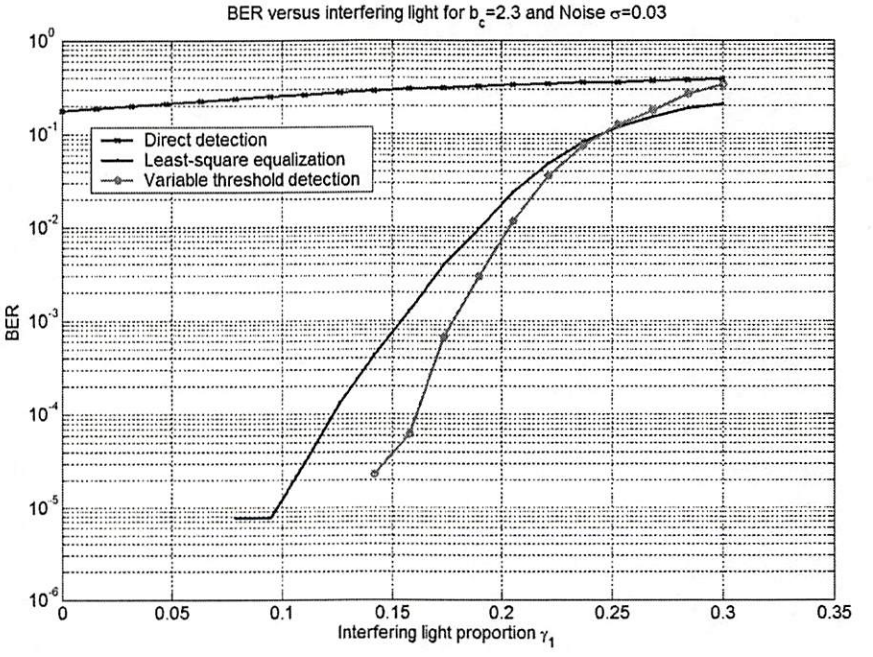



Figure 7.13. Variable threshold detection results for moderate ISI ($b_c$=2.36) and moderate noise ($\sigma$=0.03) on hexagonal detector arrays.

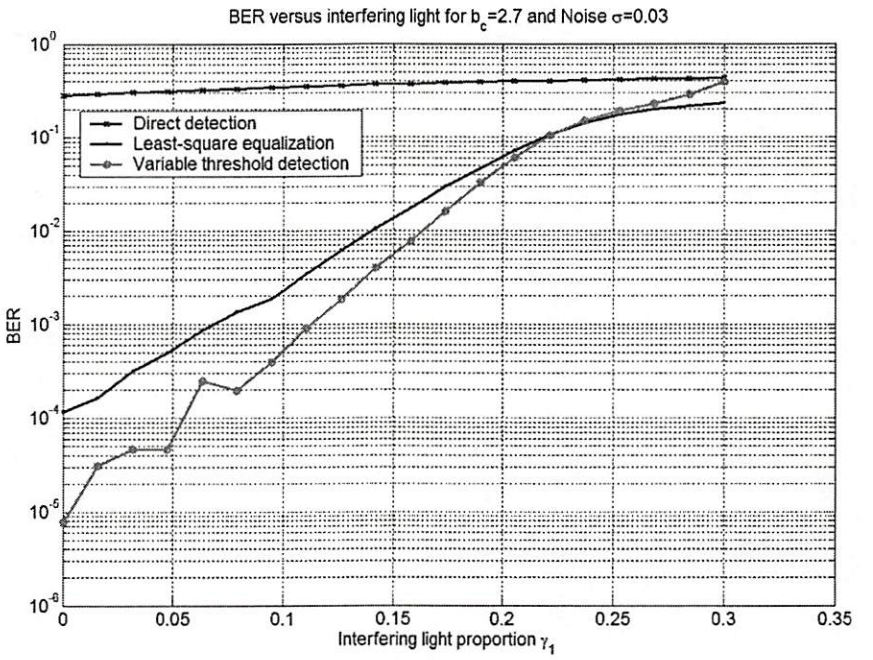



Figure 7.14. Variable threshold detection results for severe ISI ($b_c$=2.69) and moderate noise ($\sigma$=0.03) on hexagonal detector arrays.

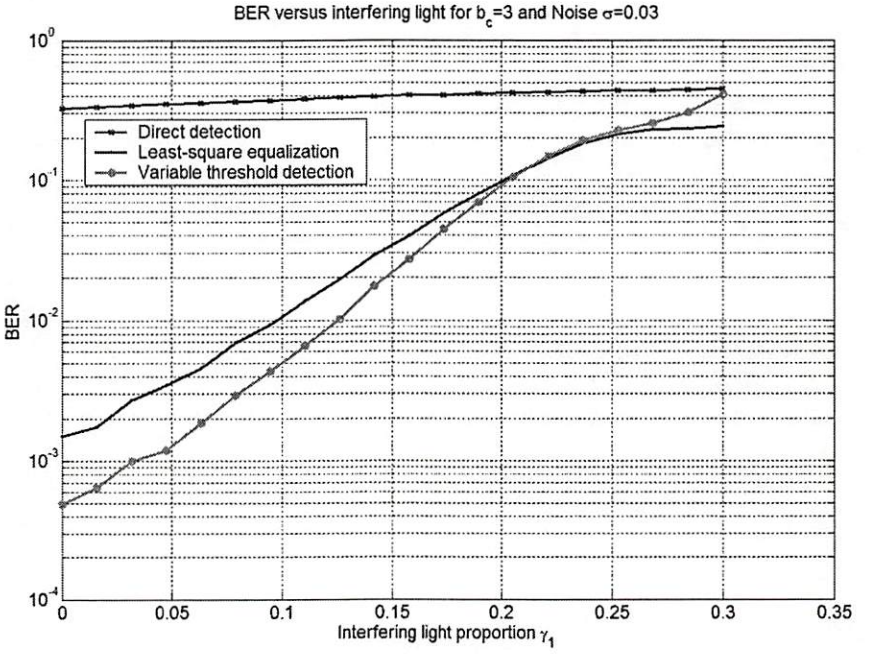



Figure 7.15. Variable threshold detection results for severe ISI ($b_c$=3) and moderate noise ($\sigma$=0.03) on hexagonal detector arrays.

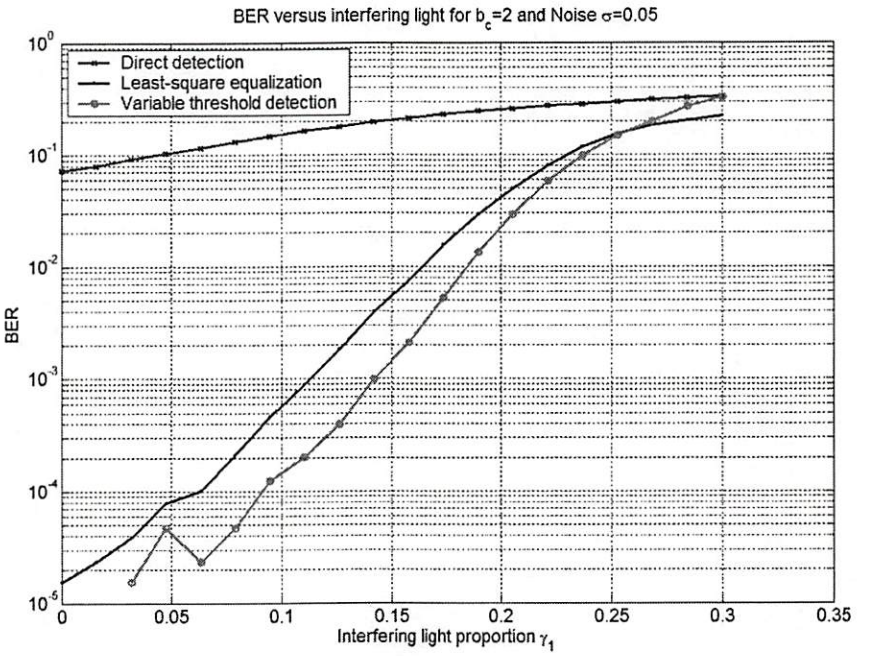



Figure 7.16. Variable threshold detection results for moderate ISI ($b_c$=2.06) and severe noise ($\sigma$=0.05) on hexagonal detector arrays.

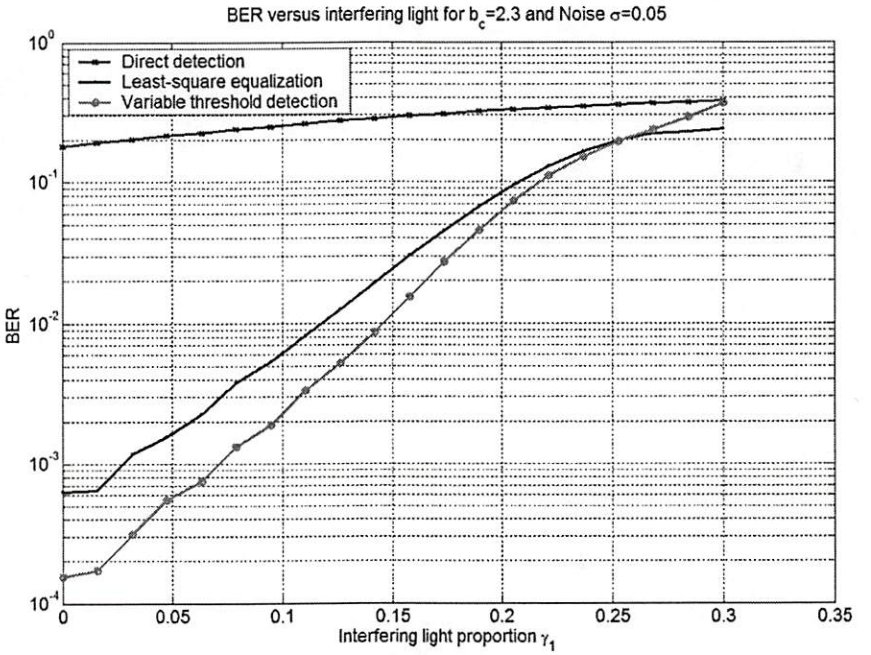



Figure 7.17. Variable threshold detection results for moderate ISI ($b_c$=2.36) and severe noise ($\sigma$=0.05) on hexagonal detector arrays.

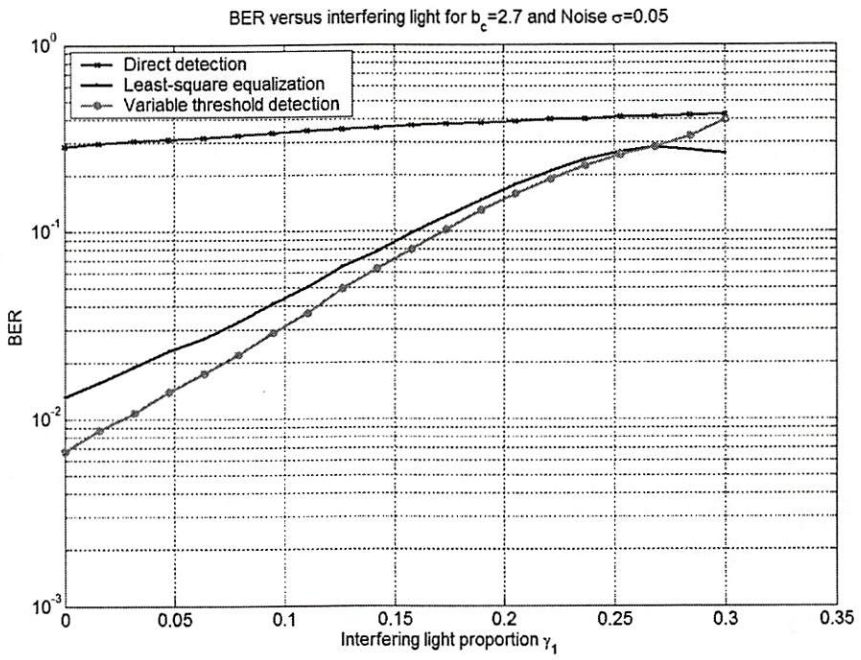



Figure 7.18. Variable threshold detection results for severe ISI ($b_c$=2.69) and severe noise ($\sigma$=0.05) on hexagonal detector arrays.

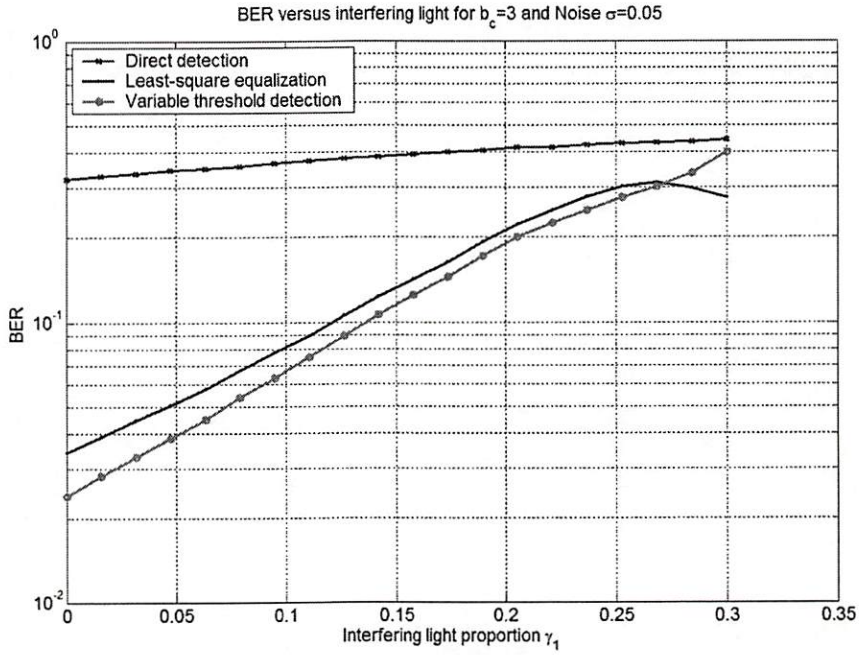



Figure 7.19. Variable threshold detection results for moderate ISI ($b_c$=3) and severe noise ($\sigma$=0.05) on hexagonal detector arrays.

# Chapter 8 Implementation of Least Square Equalization and Variable Threshold Method with Fixed-Point Arithmetic

From the results of the previous chapters, the least square equalization and variable threshold methods exhibit significant improvement over the simple fixed threshold method. These results assume that these algorithms are performed in floating-point with double precision arithmetic (64-bit). In this chapter, we explore the effects of finite precision arithmetic on least square equalization and variable threshold method algorithms. We explore the trade-off between system performance and simplifying the computation hardware using lower precision arithmetic.

## 8.1 Floating-Point Arithmetic

In most programming languages, it is common to use floating-point numbers with double-precision or single-precision to represent data. According to the IEEE 754 standard, there are three parts of the floating-point representation: sign, exponent, and fraction. The difference between double-precision and single-precision is the number of bits assigned to the exponent and fraction parts as shown in Table 8.1 and Figure 8.1.

| | Single-precision bit allocation | Double-precision bit allocation |
|---|---|---|
| **Sign (0 = positive, 1 = negative)** | 1 bit | 1 bit |
| **Exponent** | 8 bits (biased by 127) | 11 bits (biased by 1023) |
| **Fraction** | 23 bits | 52 bits |

Table 8.1. Bit assignment of double-precision and single-precision floating point arithmetic.

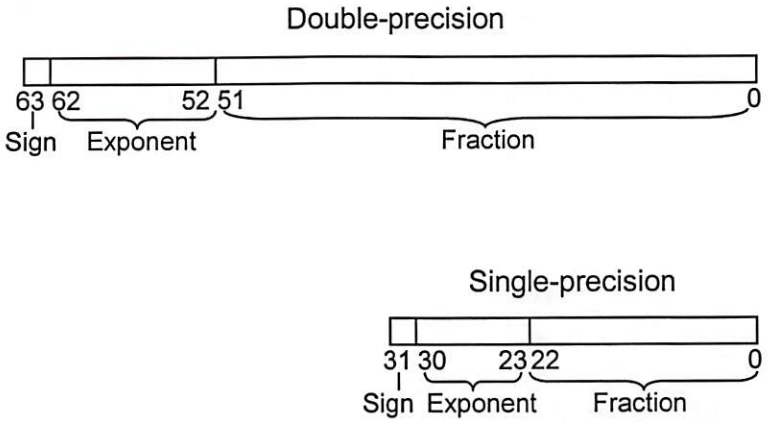



Figure 8.1. Bit allocation of double-precision and single-precision floating point arithmetic.

The numbering scheme is in ascending order starting from the least significant bit with number 0 to the most significant bit. The number it represents has the form

$$\pm 1.f \times 2^{E}, \tag{8.1}$$

where $f$ is the fraction part and $E$ is the exponent part.

The advantage of floating point representation is larger dynamic range compared to fixed-point arithmetic as described in the next section. The corresponding ranges for double-precision and single-precision representations are shown in Table 8.2.

| | **Single-precision** | **Double-precision** |
|---|---|---|
| **Minimum absolute numbers** | $1.17549 \times 10^{-38}$ | $2.22507 \times 10^{-308}$ |
| **Maximum absolute numbers** | $3.40282 \times 10^{38}$ | $1.79769 \times 10^{308}$ |

Table 8.2. Dynamic range of double-precision and single-precision floating point arithmetic.

Although the floating-point representation provides larger dynamic range, it also requires larger memory usage and more computing power. Hence, the hardware that supports floating-point computation usually costs more than the fixed-point counterpart. To reduce system costs, it is desirable to employ fixed-point algorithms if system performance is not sacrificed.

### 8.2 Fixed-Point Arithmetic

When implementing algorithms in digital hardware, we usually use fixed-point data types to save hardware cost and memory requirement. In terms of hardware, both the fixed-point and floating-point data consist of a series of bits (1's and 0's). The application or software interprets the representation of these bit sequences. The bit sequence of general fixed-point numbers is shown in Figure 8.2.

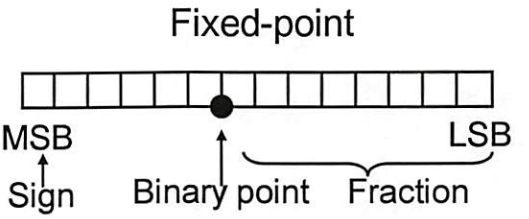



Figure 8.2. Bit allocation of fixed-point numbers.

The MSB denotes the most significant bit or the highest bit, whereas the LSB denotes the least significant bit or the lowest bit. The location of the binary point determines how this fixed-point value is represented. The part to the right of the binary point is the fractional part of the number. Therefore, for a word length of $n$ bits, moving the binary point toward the LSB will reduce the number of bits used in fractional part but increases the absolute value of the number it can represent. For example, a fixed-point number with word length $n$, 1 bit for signed bit, and $m$ bits for the fractional part, has the range shown in Figure 8.3.

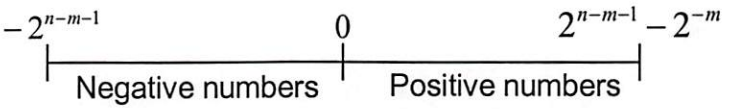



Figure 8.3. Numeric range of fixed-point numbers with word length $n$.

Here we use the two's complement representation that has a unique bit sequence for zero. Therefore, the number of positive and negative numbers are unequal. Since the range of numbers is limited, overflow will occur if the number to be represented exceeds the range. When overflow occurs, the number is truncated and represented by the closest available number in the given representation.

Because of quantization, the horizontal line shown in Figure 8.3 does not represent all the numbers within that range. Due to the limited precision, numbers on that horizontal line are quantized to a set of fixed-point values. The precision is the difference between successive numbers that can be expressed in fixed-point data format. The value of precision is equal to the value of least significant bit.

If any number produced from calculation does not equal exactly the expressible fixed-point numbers, a rounding quantization method is used and the precision is lost. There are three major rounding methods: round, floor, and ceiling. The round method uses the closest representable fixed-point number. On the contrary, the floor and ceiling methods round to the closest number in the direction of negative and positive infinity, respectively.

Because of the rounding operation, the error stems from the difference between the actual number and the representable number. When the round method is used, the maximum error is half the precision. However, the floor and ceiling methods produce twice the maximum error of the round method. Hence, the round method is the most frequently used among all three rounding methods.

Since the position of the binary point determines the value of the least significant bit, it in turn determines the precision. Hence, the binary point provides a trade off between range and precision. Moving the binary point toward the MSB increases the precision at the cost of reducing range, and vice versa. Some examples of range and precision for various fixed-point formats are shown in Table 8.3.

| Fixed-point data format (word length, fraction bits) | Range | Precision |
|---|---|---|
| (8, 2) | -32~31.75 | 0.25 |
| (8, 3) | -16~15.875 | 0.125 |
| (8, 4) | -8~7.9375 | 0.0625 |
| (8, 5) | -4~3.96875 | 0.03125 |
| (9, 3) | -32~31.875 | 0.125 |
| (9, 4) | -16~15.9375 | 0.0625 |
| (9, 5) | -8~7.96875 | 0.03125 |
| (9, 6) | -4~3.984375 | 0.015625 |
| (10, 6) | -8~7.984375 | 0.015625 |
| (11, 7) | -8~7.9921875 | 0.0078125 |
| (12, 8) | -8~7.99609375 | 0.00390625 |

Table 8.3. Examples of range and precision for various fixed-point formats.

Even with the same word length, the range and precision vary considerably for different choices of data formats. Therefore, we should choose the best data format based on the statistical characteristics of the actual data.

## 8.3 Fixed-Point Arithmetic Implementation

### 8.3.1 Least Square Equalization with Fixed-Point Arithmetic

In the least square equalization method, the data detection is done after an equalization step given by

$$\underline{\hat{\mathbf{a}}} = \underline{\mathbf{D}}^{+}\underline{\mathbf{r}}, \tag{8.2}$$

where $\hat{\underline{\mathbf{a}}}$ is the estimate of the data, $\underline{\mathbf{D}}^{+}$ is the pseudo inverse matrix, and $\underline{\mathbf{r}}$ is the received signal. In the fixed-point arithmetic implementation, the number of bits used to represent data is constant through the algorithm. Since the received signal is normalized such that the intensity value is within (0, 1), we allocate all the bits to express the fractional part. Therefore, the binary point is at the leftmost location in the fixed-point representation.

In fixed-point least square equalization, we quantize the pseudo inverse matrix $\underline{\mathbf{D}}^{+}$ in Eq. (8.2) into fixed-point numbers and perform the processing in fixed-point arithmetic. Matrix $\underline{\mathbf{D}}^{+}$ is obtained by inverting the PSF matrices as in Eq. (4.7). Although the numerical values in the PSF matrices are in the range (0, 1), the values in the pseudo inverse matrix are significantly different. Most of these values fall within the range $-8 \sim +8$ but there are also some small values around 0 whose absolute value is on the order of $10^{-2}$. Therefore, the choice between range and precision is crucial to the system performance.

In the simulation results, we notice that the quantization of the pseudo inverse matrix is a critical factor in the fixed-point implementation. Since the location of the binary point determines the trade off between range and precision, it is an important parameter in the quantization process. We use combinations of different numbers of bits and binary point locations to see the impact on the system performance as compared to the results using the floating-point algorithm.

### 8.3.2 Variable Threshold Method with Fixed-Point Arithmetic

Using the estimation results from least square equalization, the variable threshold method significantly improves the readout system performance. Hence, we implement the variable threshold method using fixed-point arithmetic to examine the balance point between system performance and computation complexity. The block diagram of the fixed-point variable threshold method is shown in Figure 8.4.

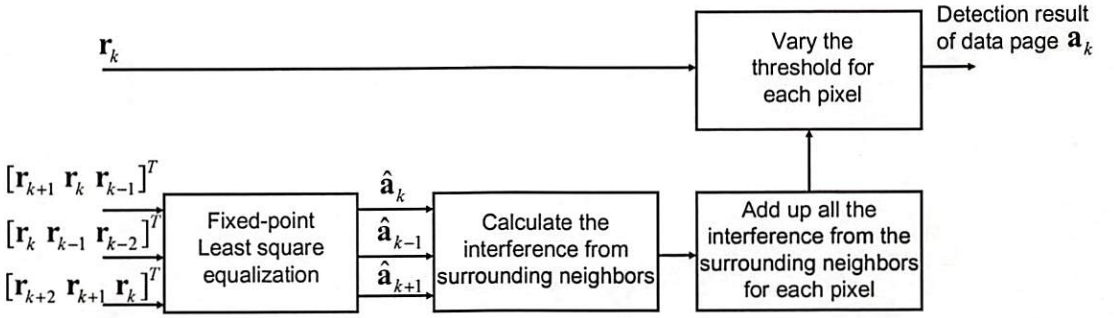



Figure 8.4. Block diagram of fixed-point variable threshold method.

The estimation of the data is provided by the fixed-point least square equalization. Since we use the same number of bits to represent data through the algorithm, the number of bits used in least square equalization and variable threshold detection is the same. However, due to different ranges of the numeric values, the location of the binary point is different in them. In the variable threshold method, the threshold is determined as in Eq. (7.4) by identifying interference components in the received signal as

$$T_k[i,j] = \frac{\gamma_2 h_2[0,0]}{2} + f_k[i,j], \tag{8.3}$$

where $f_k[i,j]$ is the interference contribution estimate for pixel at location $[i,j]$, and $h_2[0,0]$ is the center pixel PSF of the data page in focus.

To calculate the interference components, we convolve the estimated data with the 3D PSF. Since the algorithm is implemented in fixed-point arithmetic, the numeric value of 3D PSF is represented in the fixed-point format. In Eq. (5.4), the PSF is normalized such that the summation of all the entries is 1. Therefore, we conclude that the numerical values of PSF are in the range (0, 1). By using the similar quantization method for the received signal, we assign all the bits to express the fractional part of the PSF entries.

In the simulation, the quantization of least square equalization dominates the system performance over the quantization of the PSF. Since the least square equalization is used to provide estimation of the data, the reliability of the variable threshold method depends largely on the accuracy of the data estimation. Therefore, we focus on tuning the quantization parameters for the least square equalization.

## 8.4 Simulation Results

### 8.4.1 Fixed-Point Least Square Equalization Simulation Results

The simulation results for least square equalization with and without fixed-point arithmetic are shown in Figure 8.5 through Figure 8.26 for various fixed-point data formats. In Figure 8.5 through Figure 8.15, a rectangular array of detectors with a sinc point spread function (Eq. (3.8)) is assumed, while a hexagonal array of detectors

with a jinc point spread function (Eq. (3.9)) is used in Figure 8.16 through Figure 8.26. The rectangular and hexagonal blur parameters $b_s$ and $b_c$ respectively have been chosen to have equal ISI contribution for both coordinate systems. In Figure 8.5 through Figure 8.8, we compare fixed-point arithmetic with 8-bit word length to double-precision (64-bit) floating-point arithmetic on rectangular coordinates. Among these results, it is obvious that the (8, 4) fixed-point format achieves better performance than other fixed-point formats. Similar circumstances exist for hexagonal coordinates, as shown in Figure 8.16 through Figure 8.19.

From Table 8.3, the range of the (8, 4) fixed-point format is -8~7.9375, which just covers most of the numerical values in the pseudo inverse matrix. Although (8, 2) and (8, 3) formats cover a range even wider than that of (8, 4), they sacrifice precision and hence yield poor results.

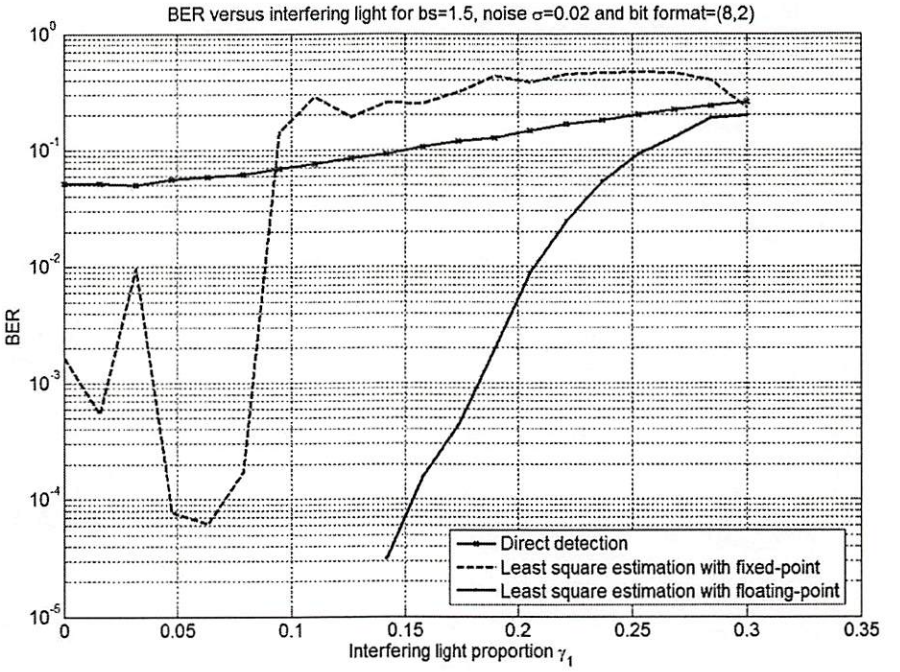
Comparing Figure 8.9 through Figure 8.12 and Figure 8.20 through Figure 8.23, we notice that the (9, 5) fixed-point is the best format for 9-bit word length. It has almost the same range (-8~7.96875) as the (8, 4) format but with two times finer precision. From the simulation results of (9, 5) and (8, 4), the increasing precision not only improves the detection, but also reduces the instability caused by the quantization error.

Based on these results, we test the addition of bits to increase the precision while keeping a suitable range to cover the pseudo inverse matrix entries. In Figure 8.13 through Figure 8.15 and Figure 8.24 through Figure 8.26, we compare three fixed-

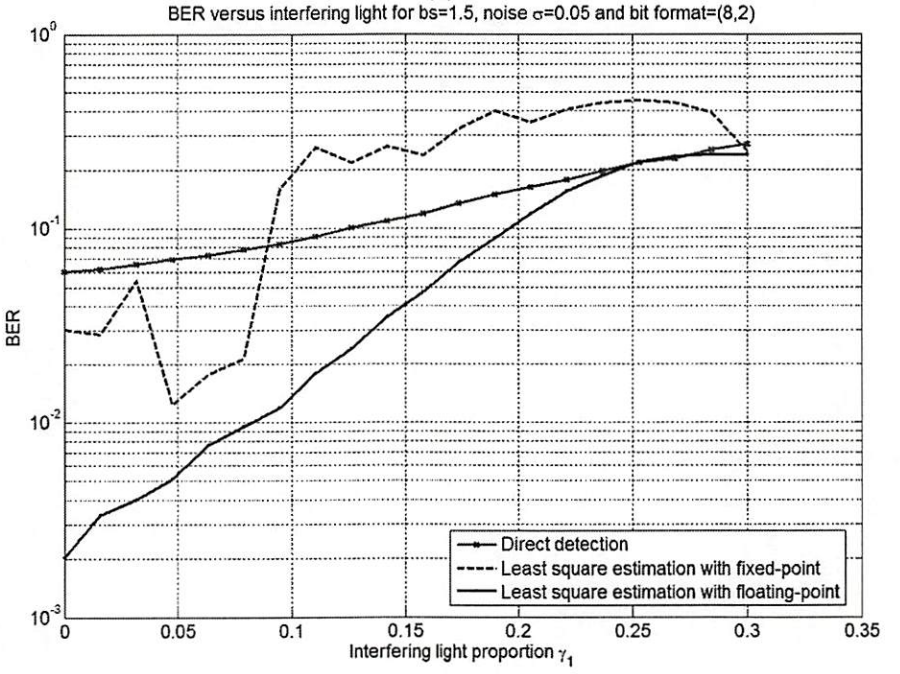
point formats with similar range and increasing precision. The simulation results confirm our earlier conclusion that increasing precision improves performance and stability. The results of the (12, 8) fixed-point format show very similar bit-error-rates to double-precision floating-point arithmetic on the rectangular coordinate system.

Since the (12, 8) fixed-point format only uses less than one fifth of the bits in 64-bit floating-point arithmetic, it exhibits a significant advantage over fixed-point arithmetic. With least square equalization implemented in fixed-point arithmetic, it saves considerable memory space, has better execution speed and reduces the hardware costs.

With the hexagonal coordinate, the fixed-point simulation results approaches the floating-point results with 10-bit and greater length fixed-point formats. It is also more robust against quantization errors than the rectangular coordinate system. A possible reason for this is that there are fewer surrounding neighbors that contribute to the interference in hexagonal coordinates. Therefore, hexagonal coordinates have an advantage for fixed point applications with a limited number of bits.

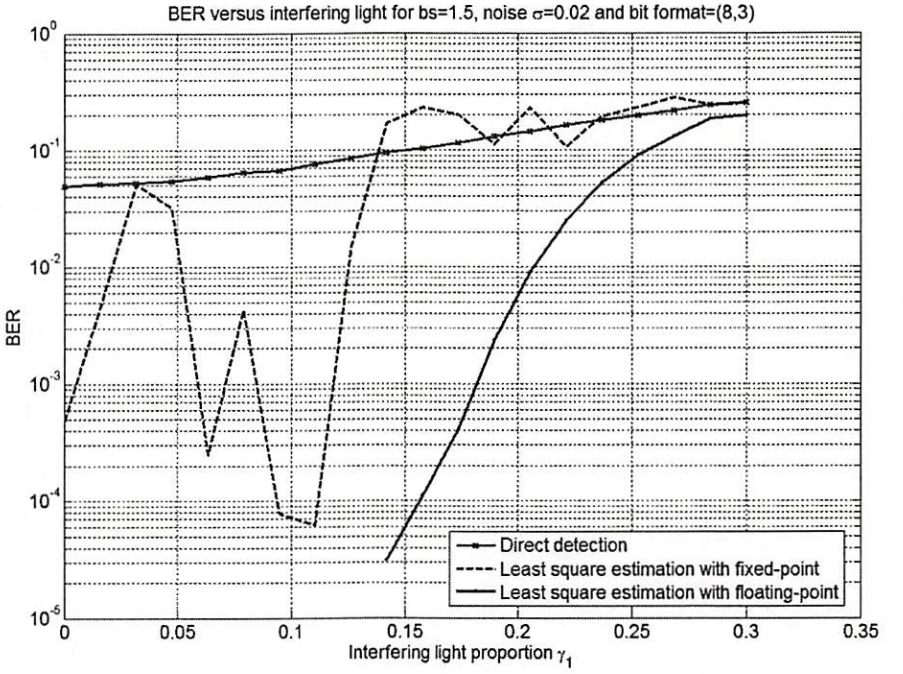



(a)

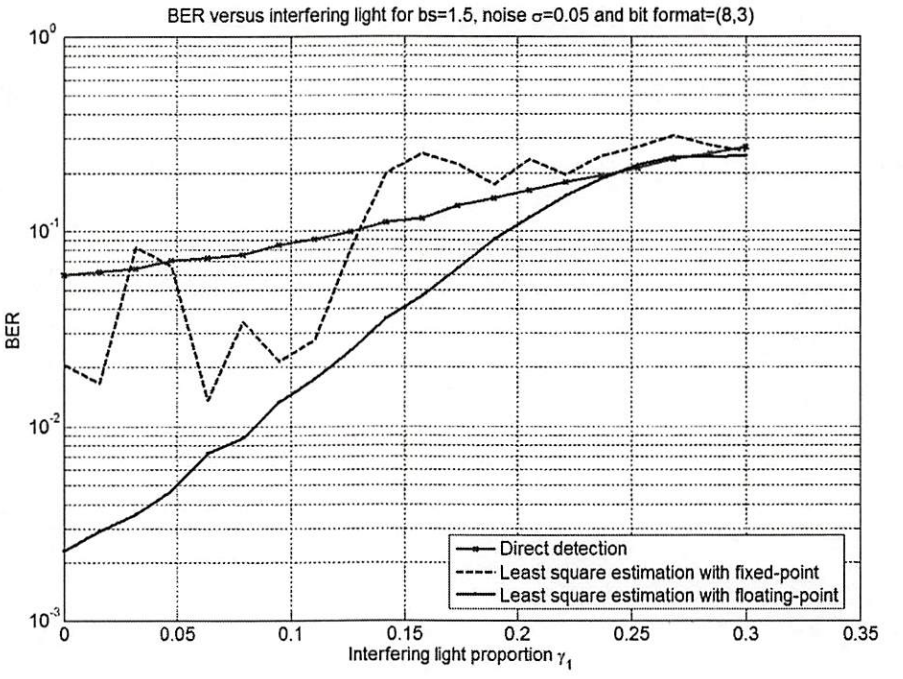



(b)

Figure 8.5. Fixed-point (8, 2) least square equalization simulation results on rectangular coordinates with $b_s$=1.5 and noise (a) $\sigma$ =0.02 (b) $\sigma$ =0.05.

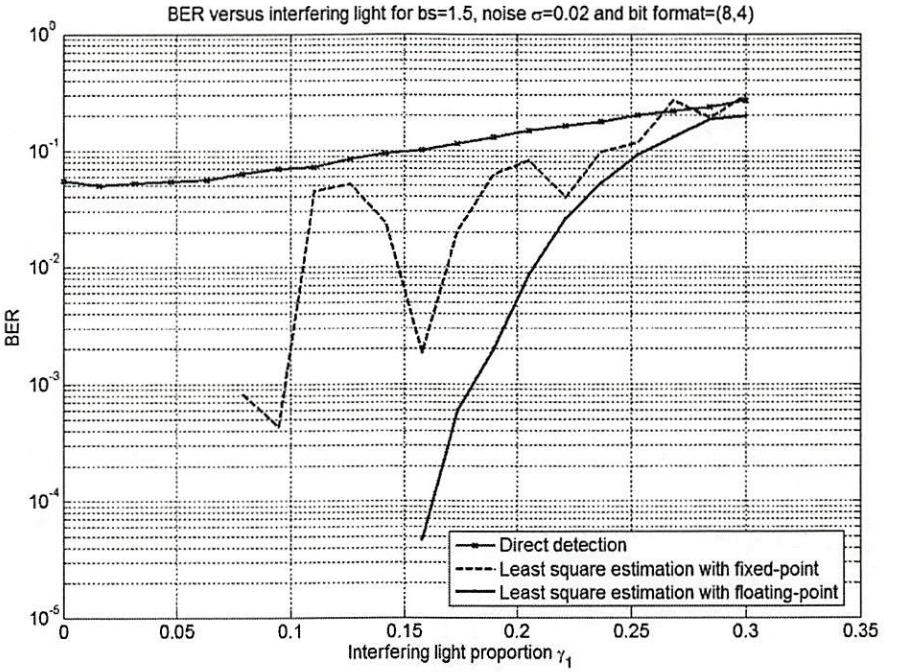



(a)

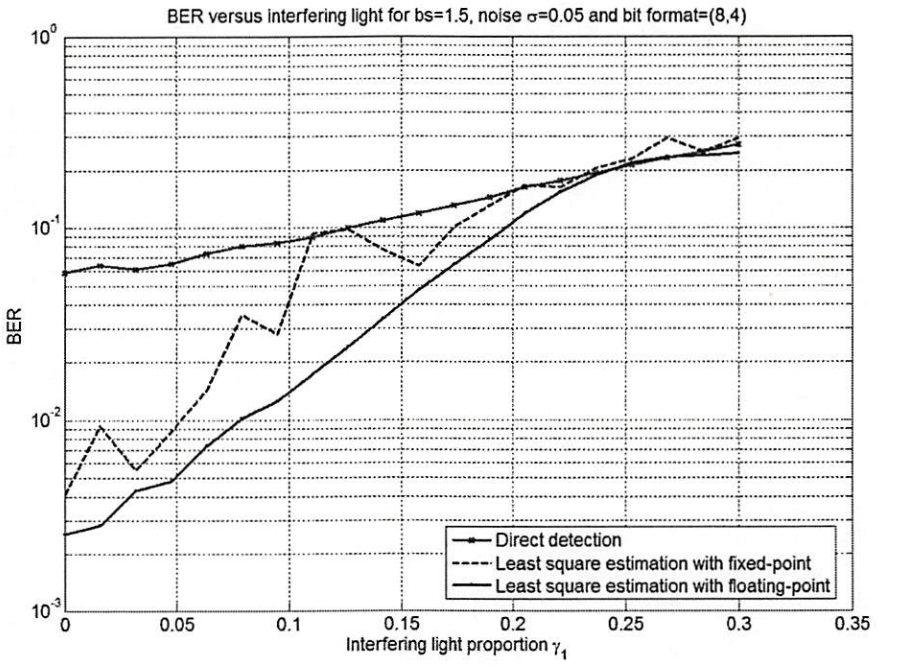



(b)

Figure 8.6. Fixed-point (8, 3) least square equalization simulation results on rectangular coordinates with $b_s$=1.5 and noise (a) $\sigma$ =0.02 (b) $\sigma$ =0.05.

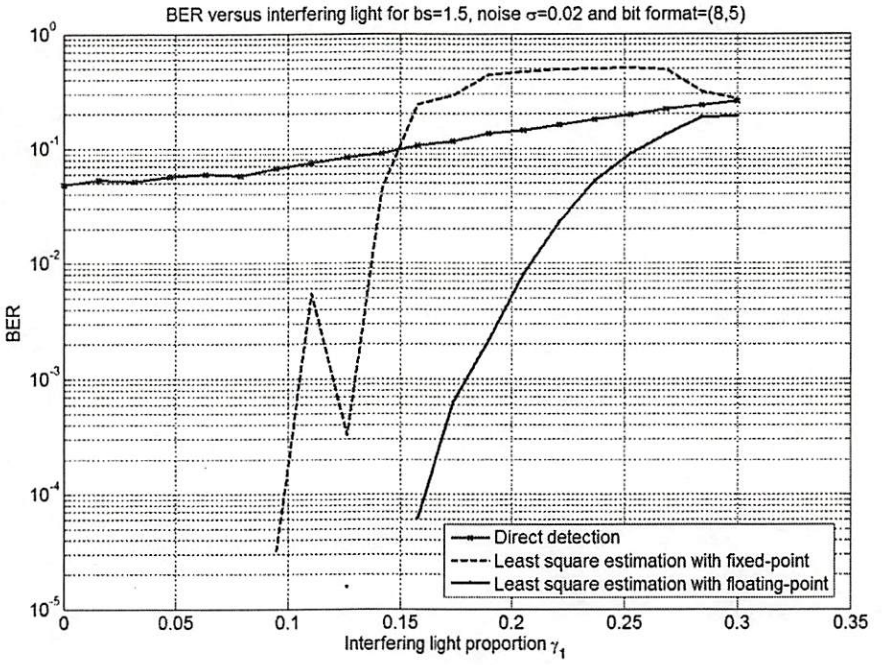



(a)

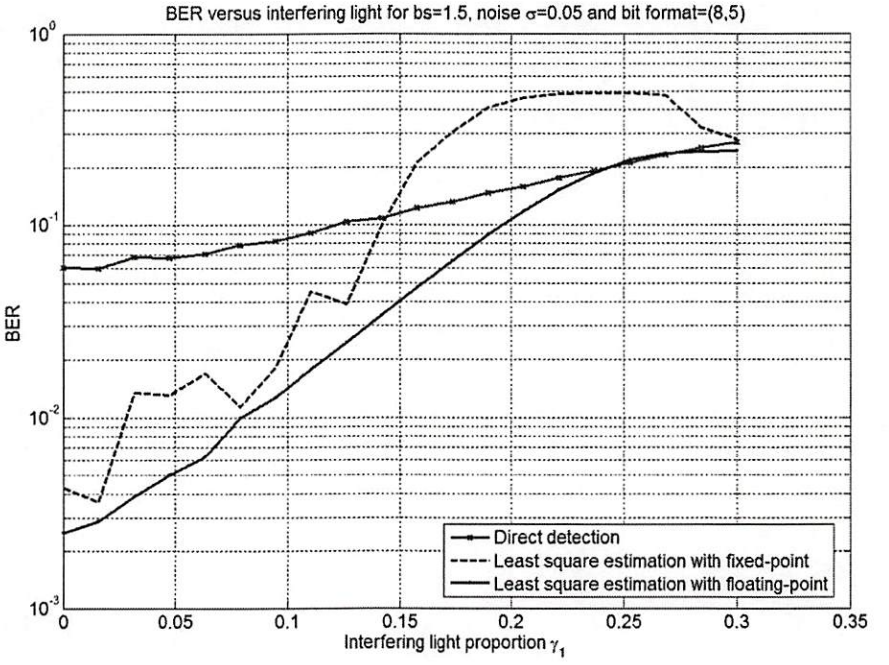



(b)

Figure 8.7. Fixed-point (8, 4) least square equalization simulation results on rectangular coordinates with $b_s$=1.5 and noise (a) $\sigma$ =0.02 (b) $\sigma$ =0.05.

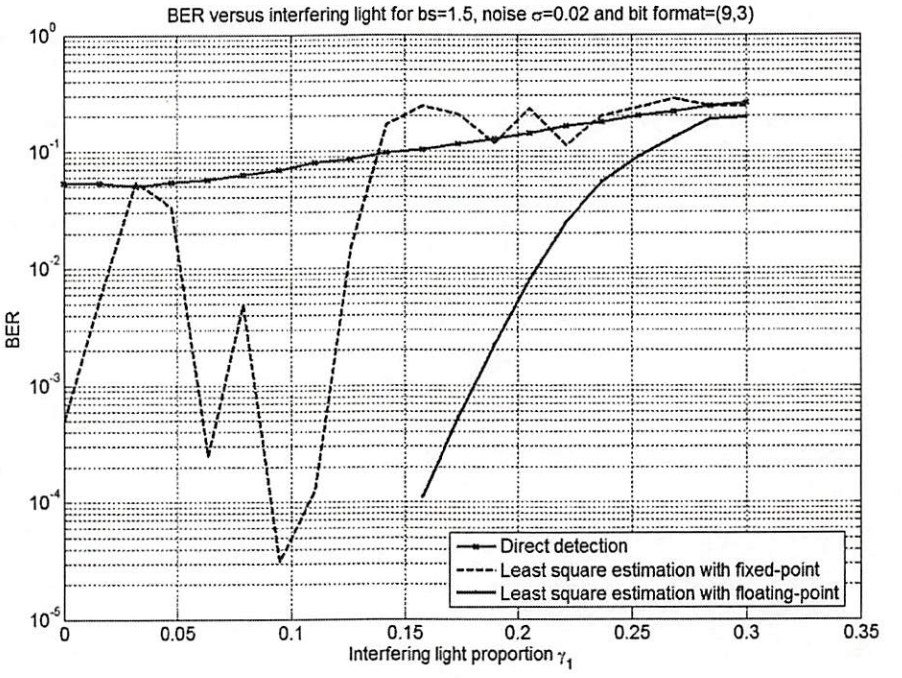



(a)

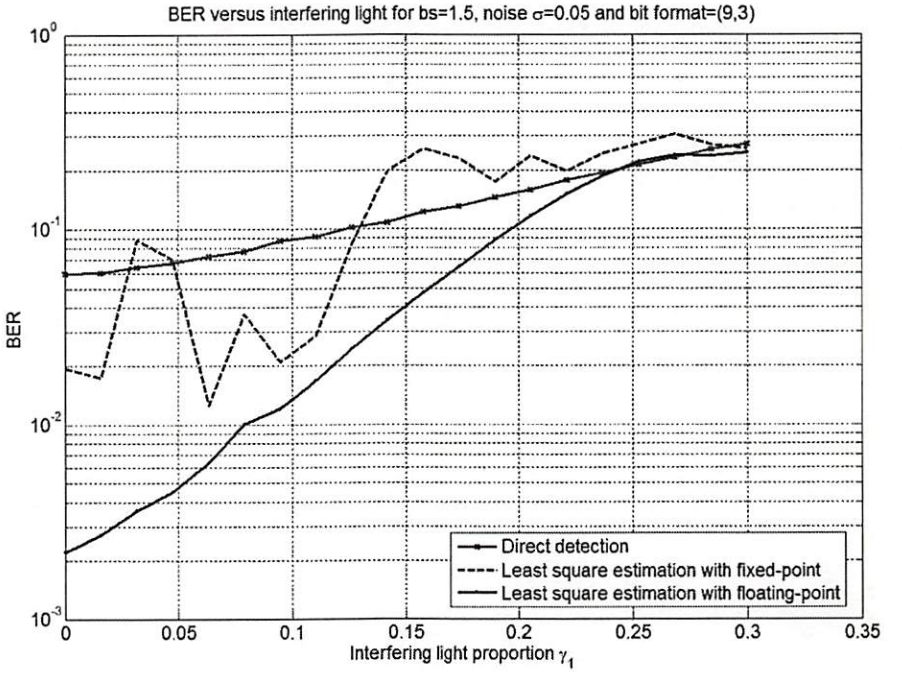



(b)

Figure 8.8. Fixed-point (8, 5) least square equalization simulation results on rectangular coordinates with $b_s$=1.5 and noise (a) $\sigma$ =0.02 (b) $\sigma$ =0.05.

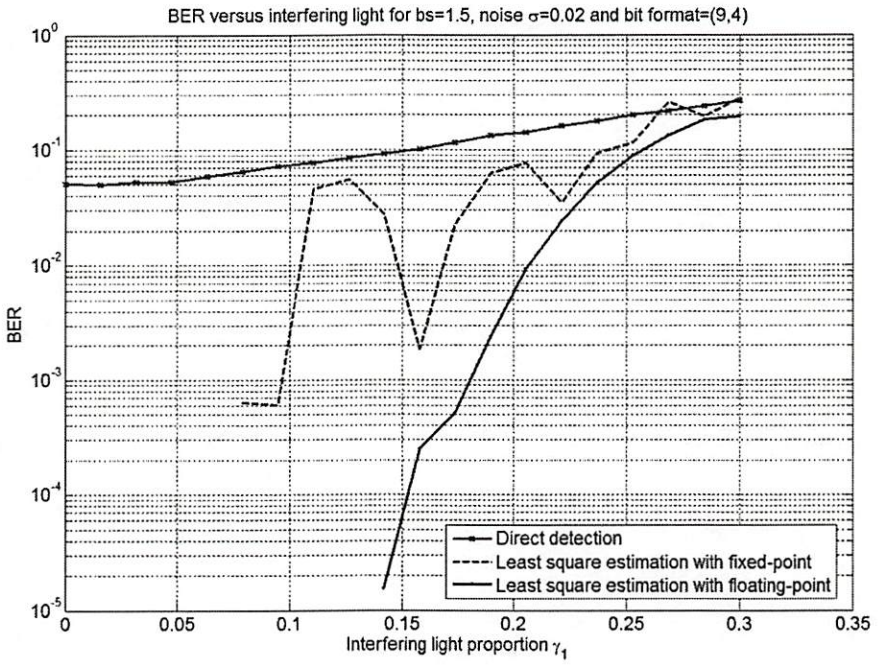



(a)

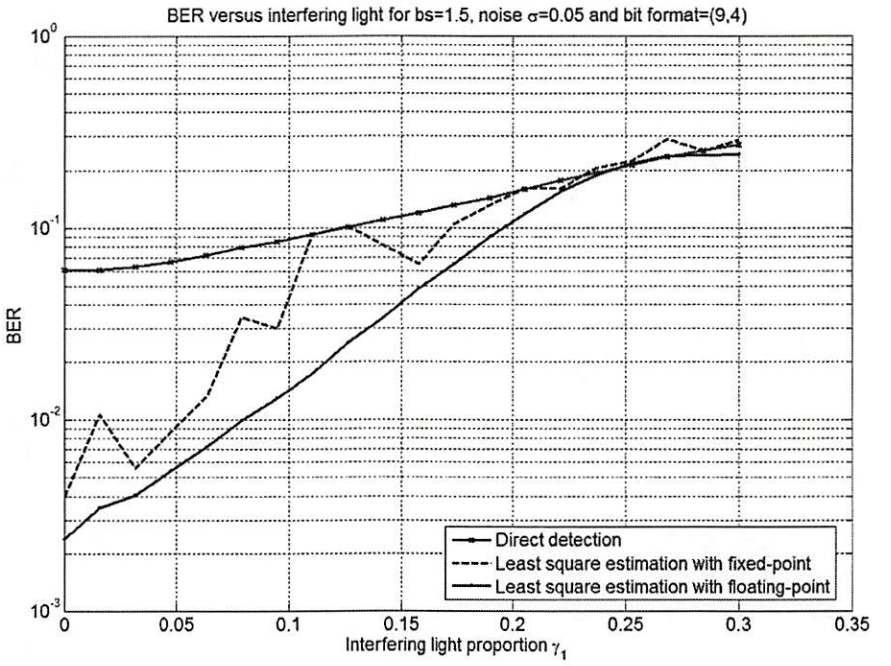



(b)

Figure 8.9. Fixed-point (9, 3) least square equalization simulation results on rectangular coordinates with $b_s$=1.5 and noise (a) $\sigma$ =0.02 (b) $\sigma$ =0.05.

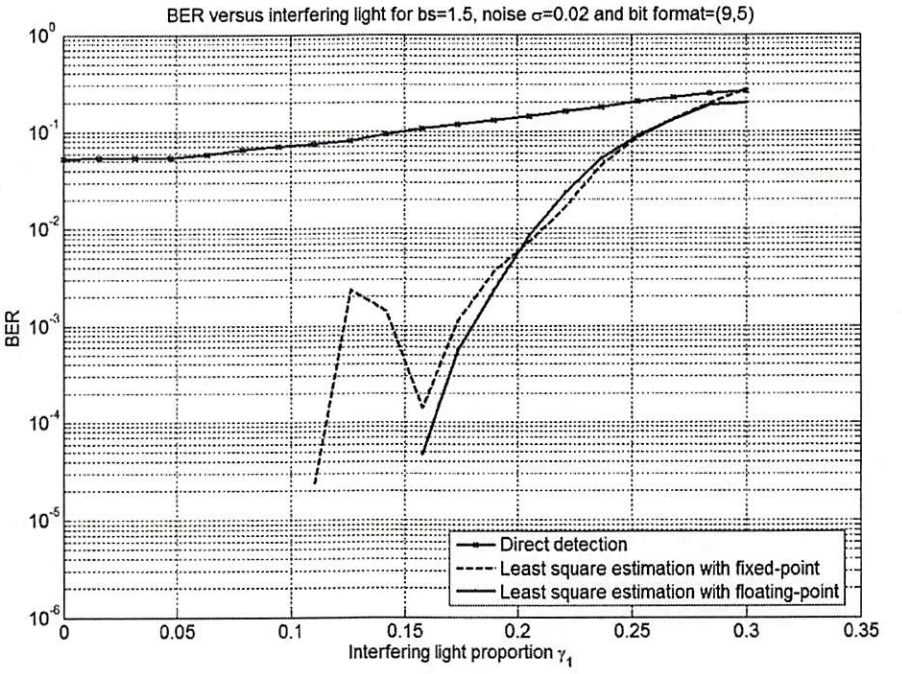



(a)

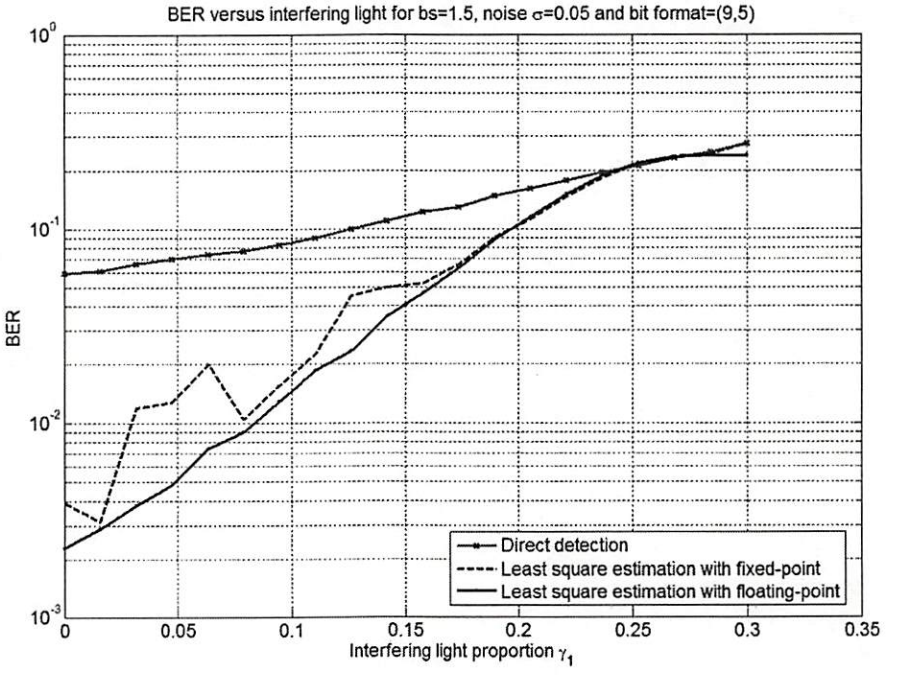



(b)

Figure 8.10. Fixed-point (9, 4) least square equalization simulation results on rectangular coordinates with $b_s$=1.5 and noise (a) $\sigma$=0.02 (b) $\sigma$=0.05.

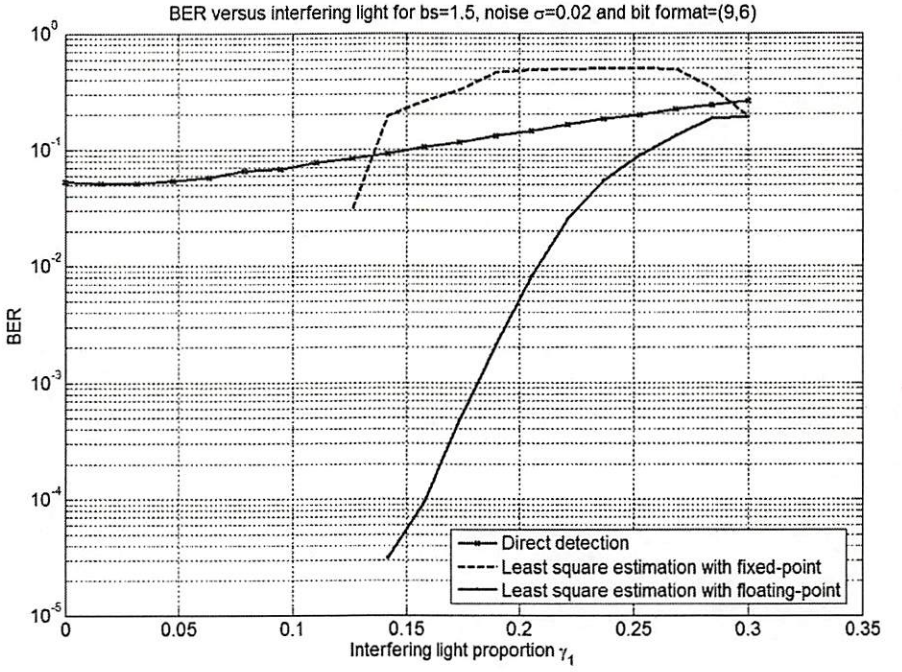



(a)

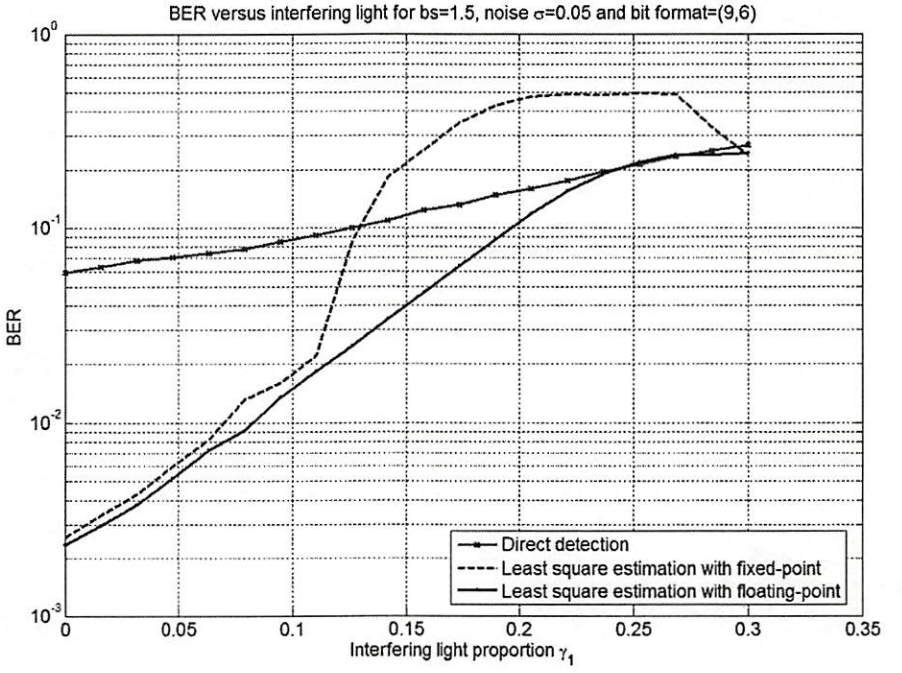



(b)

Figure 8.11. Fixed-point (9, 5) least square equalization simulation results on rectangular coordinates with $b_s$=1.5 and noise (a) $\sigma$ =0.02 (b) $\sigma$ =0.05.

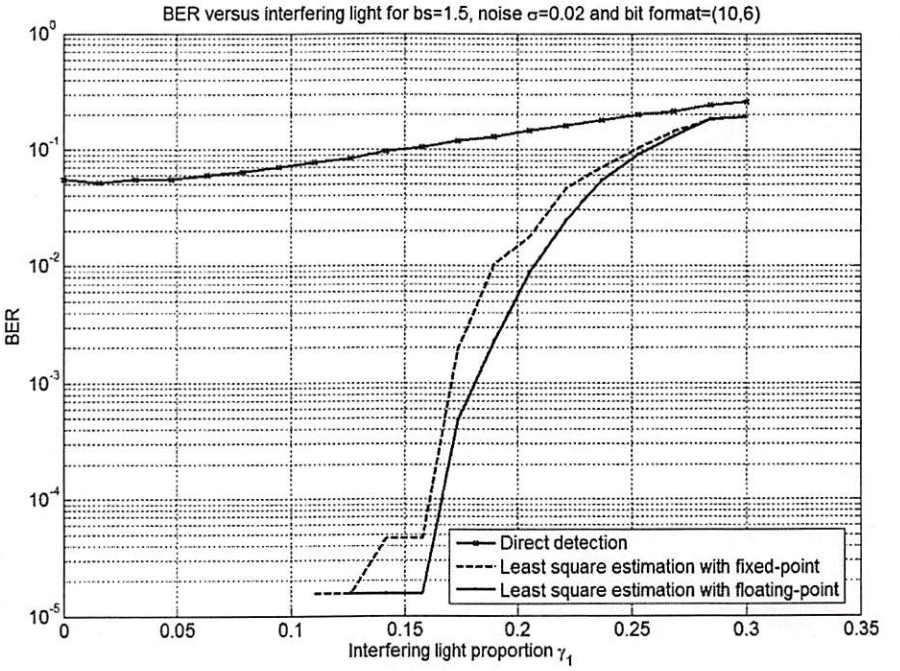



(a)

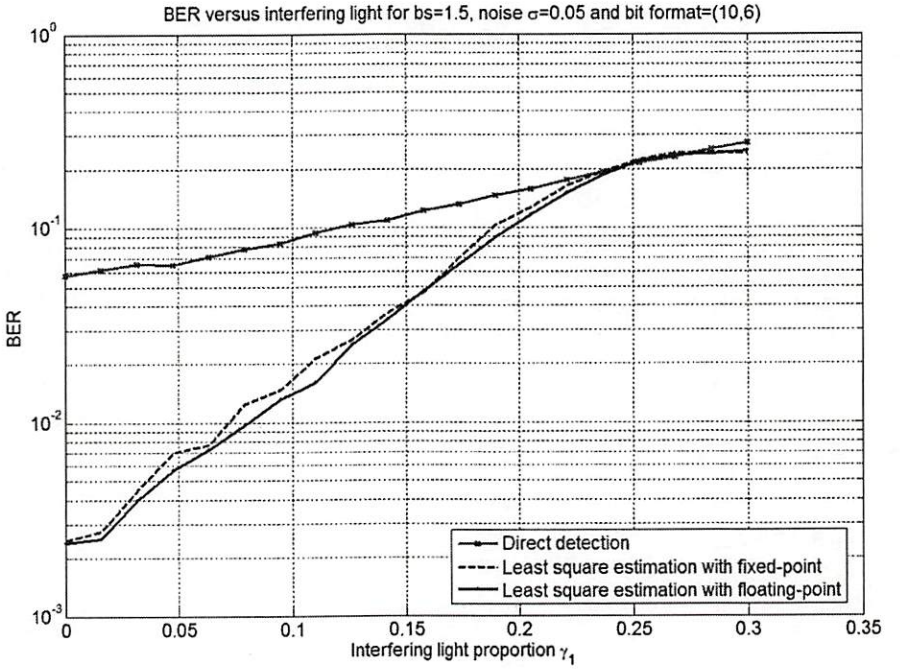



(b)

Figure 8.12. Fixed-point (9, 6) least square equalization simulation results on rectangular coordinates with $b_s$=1.5 and noise (a) $\sigma$ =0.02 (b) $\sigma$ =0.05.

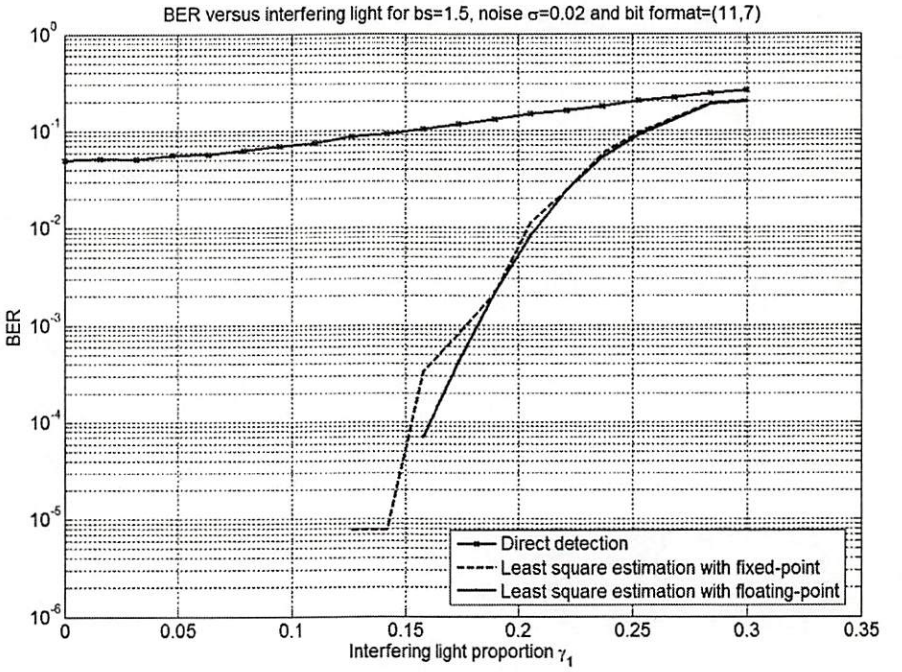



(a)

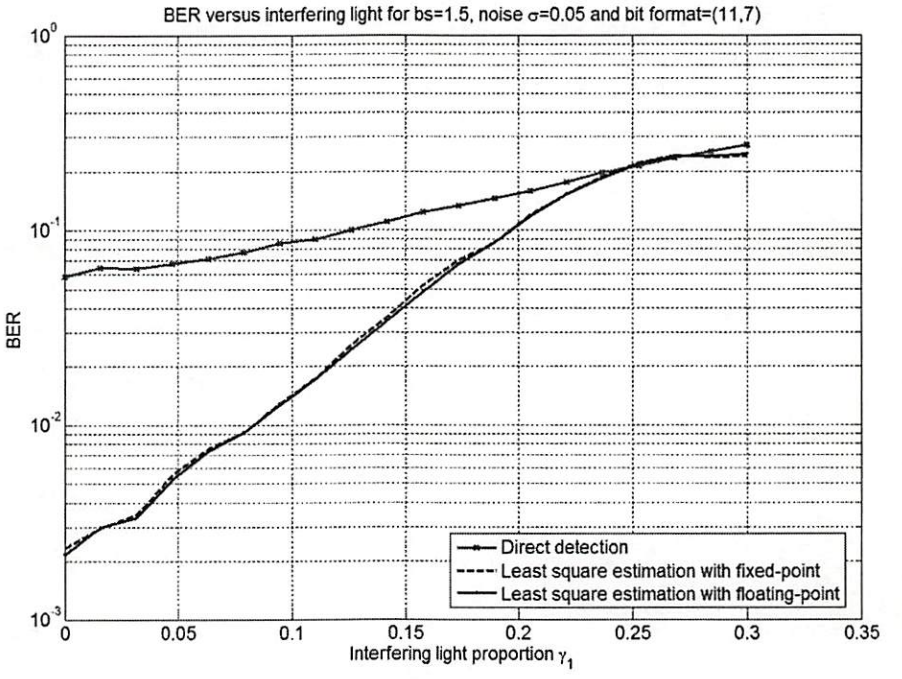



(b)

Figure 8.13. Fixed-point (10, 6) least square equalization simulation results on rectangular coordinates with $b_s$=1.5 and noise (a) $\sigma$ =0.02 (b) $\sigma$ =0.05.

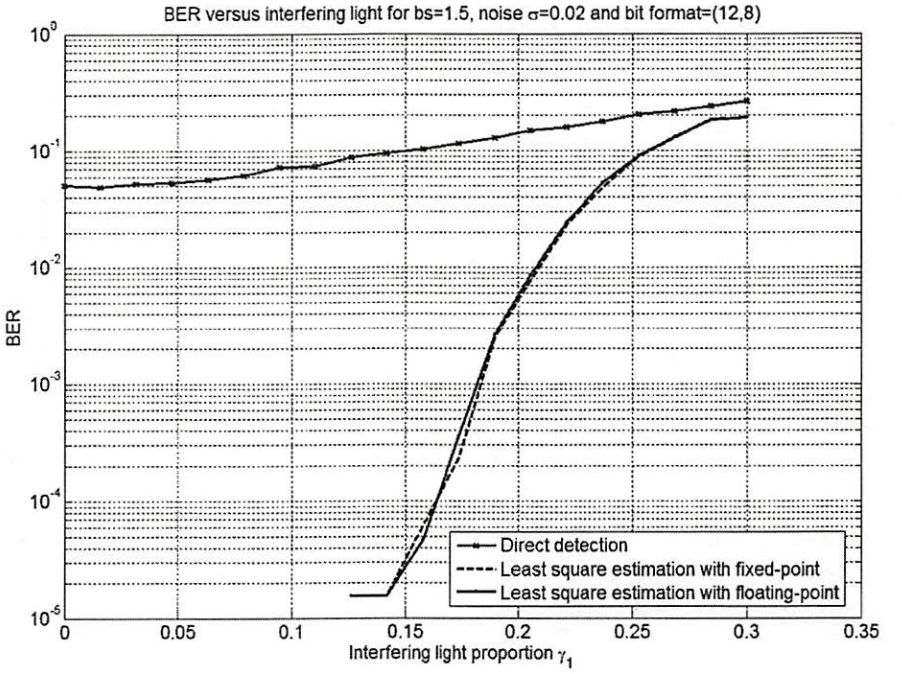



(a)

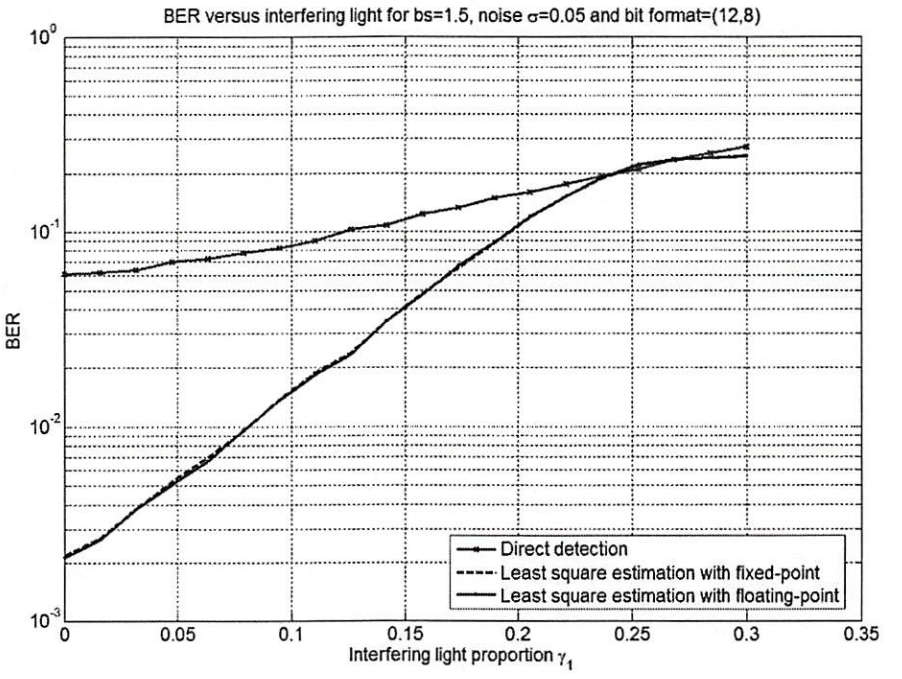



(b)

Figure 8.14. Fixed-point (11, 7) least square equalization simulation results on rectangular coordinates with $b_s$=1.5 and noise (a) $\sigma$ =0.02 (b) $\sigma$ =0.05.

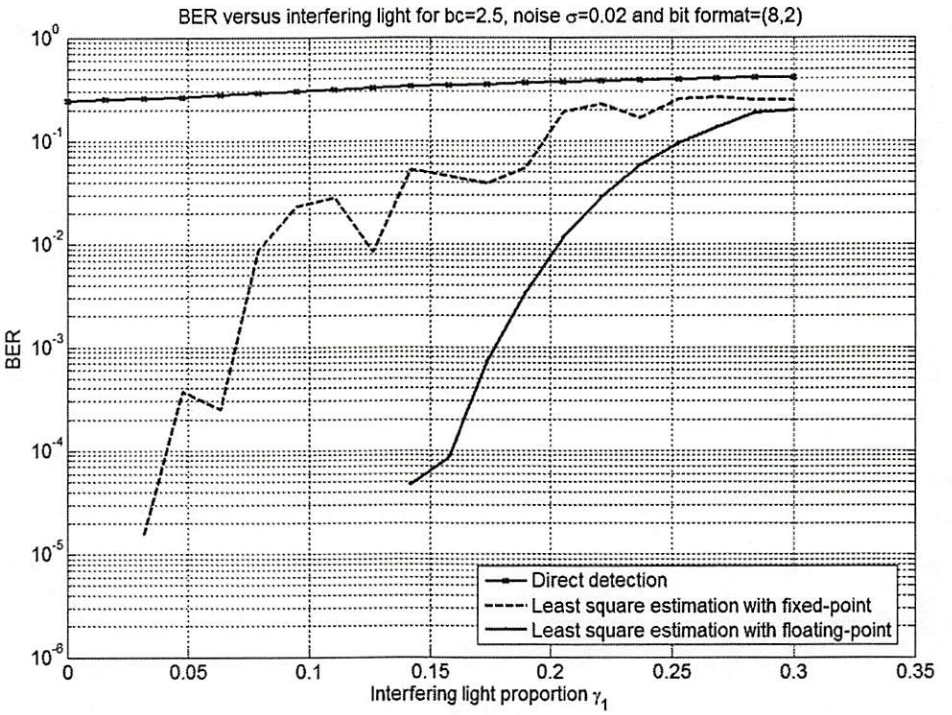



(a)

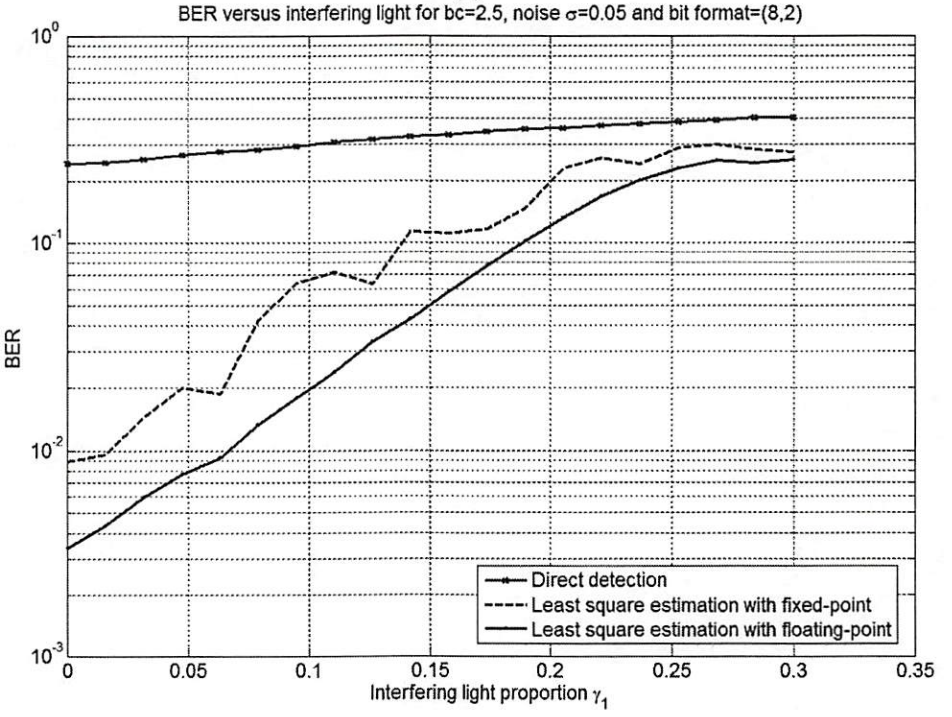



(b)

Figure 8.15. Fixed-point (12, 8) least square equalization simulation results on rectangular coordinates with $b_s$=1.5 and noise (a) $\sigma$ =0.02 (b) $\sigma$ =0.05.

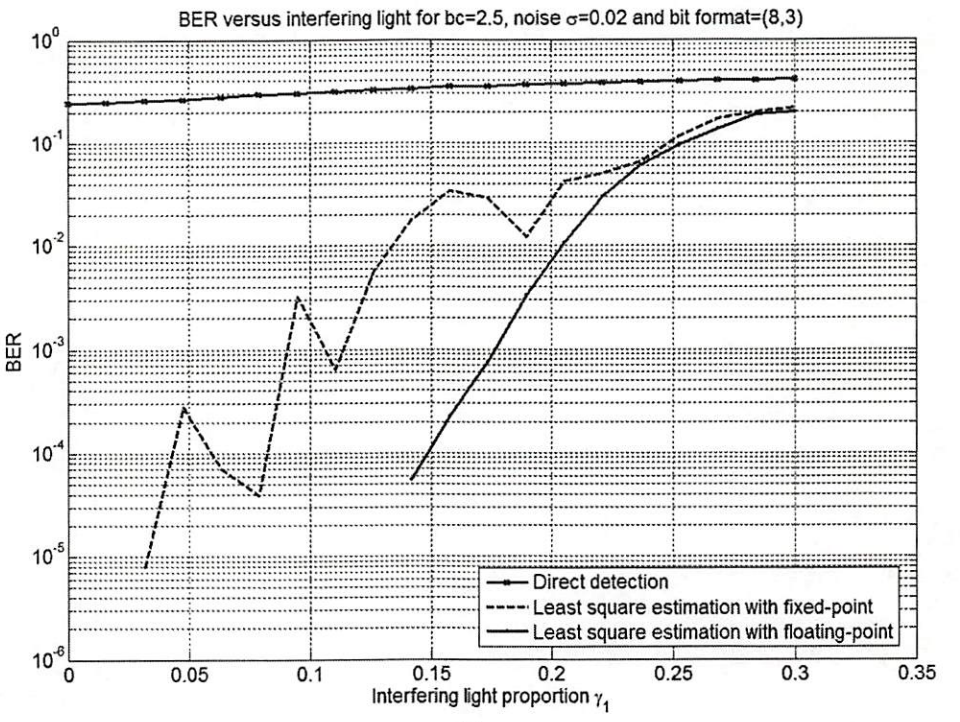



(a)

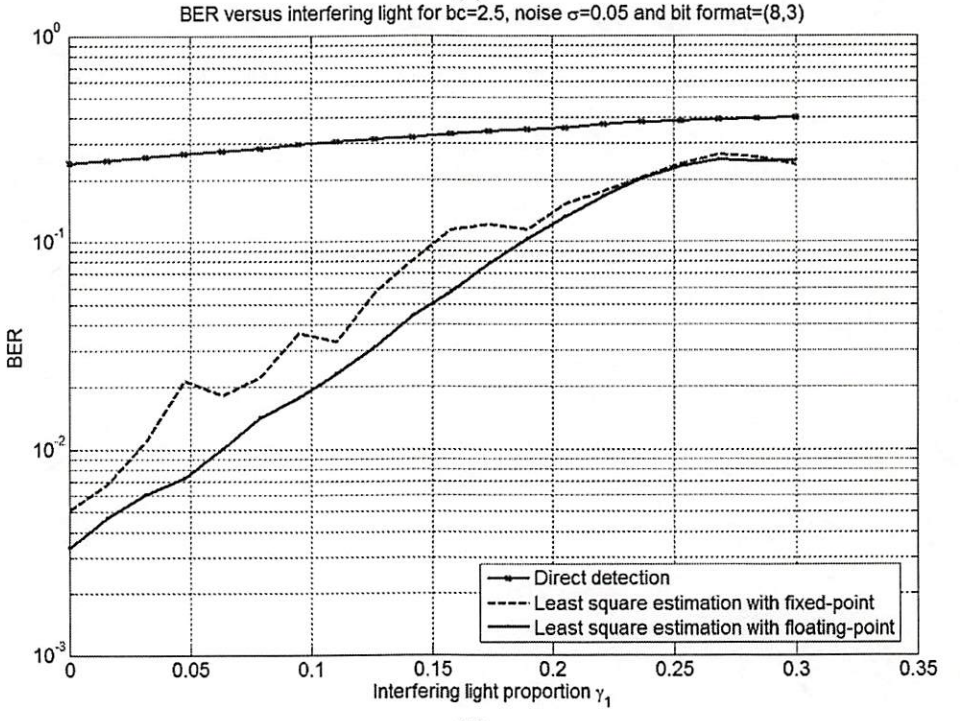



(b)

Figure 8.16. Fixed-point (8, 2) least square equalization simulation results on hexagonal coordinates with $b_c$=2.52 and noise (a) $\sigma$ =0.02 (b) $\sigma$ =0.05.

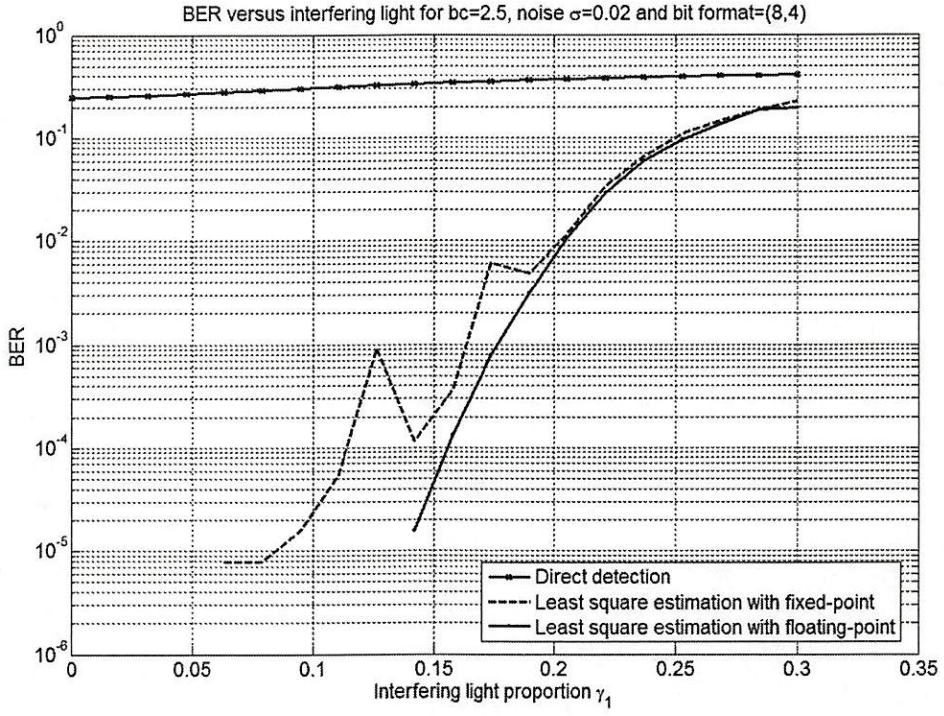



(a)

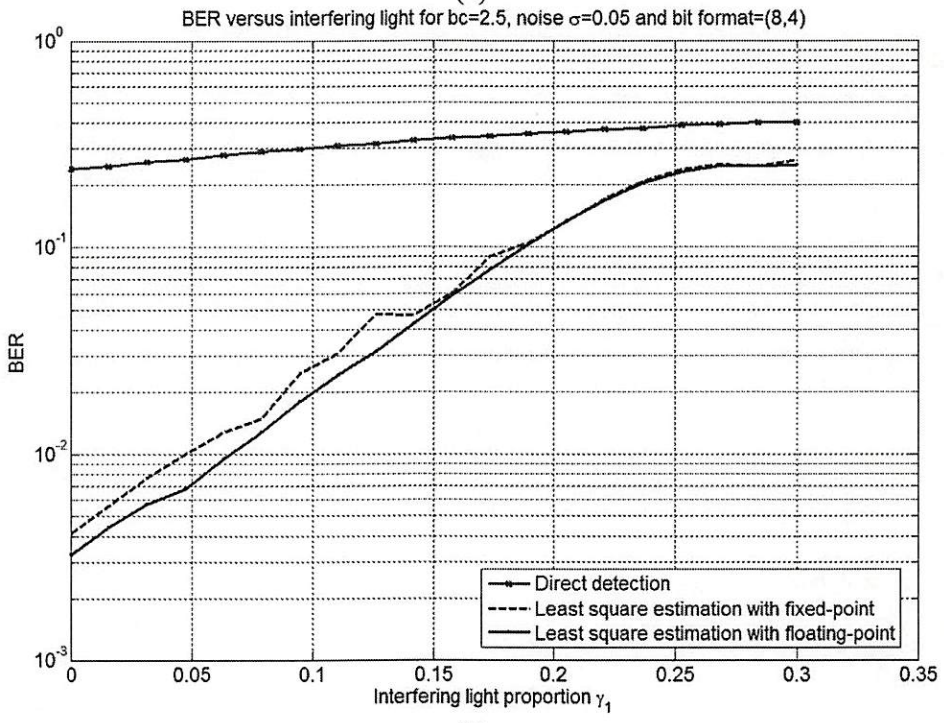



(b)

Figure 8.17. Fixed-point (8, 3) least square equalization simulation results on hexagonal coordinates with $b_c$=2.52 and noise (a) $\sigma$ =0.02 (b) $\sigma$ =0.05.

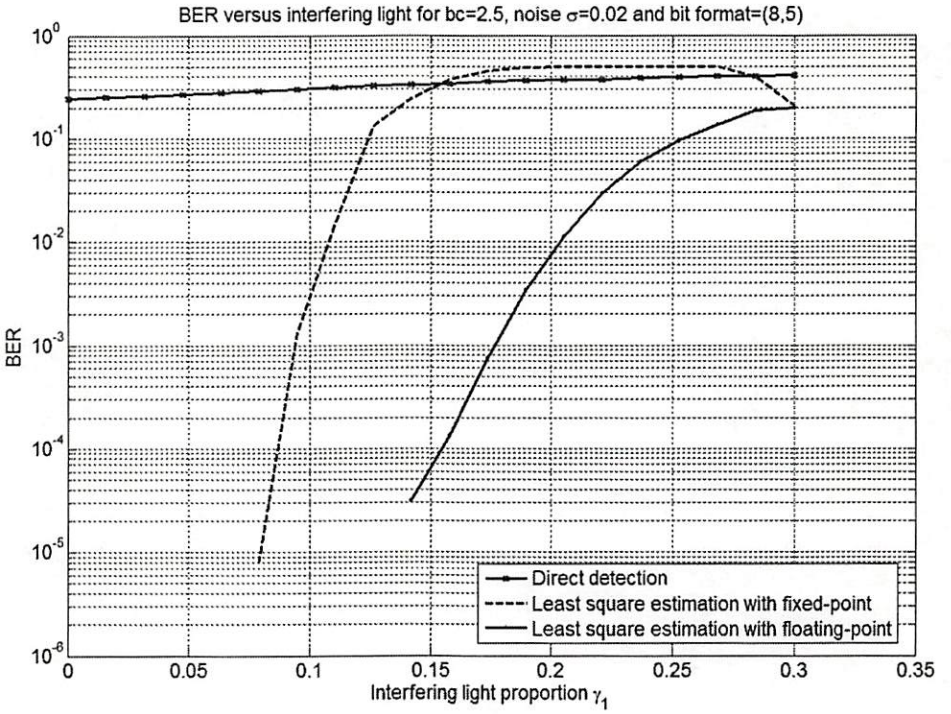



(a)

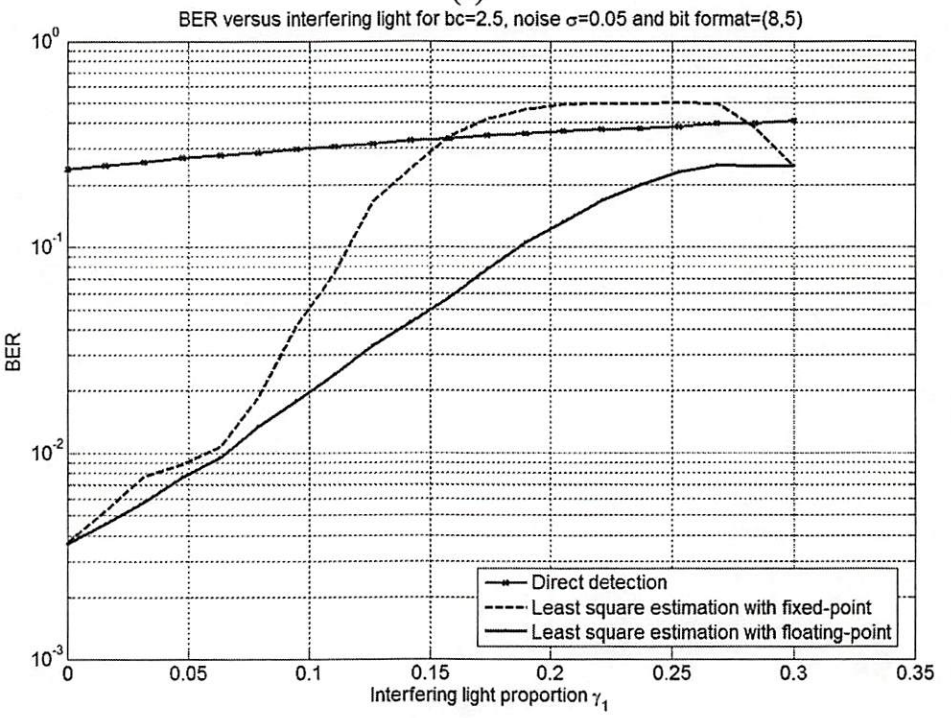



(b)

Figure 8.18. Fixed-point (8, 4) least square equalization simulation results on hexagonal coordinates with $b_c$=2.52 and noise (a) $\sigma$ =0.02 (b) $\sigma$ =0.05.

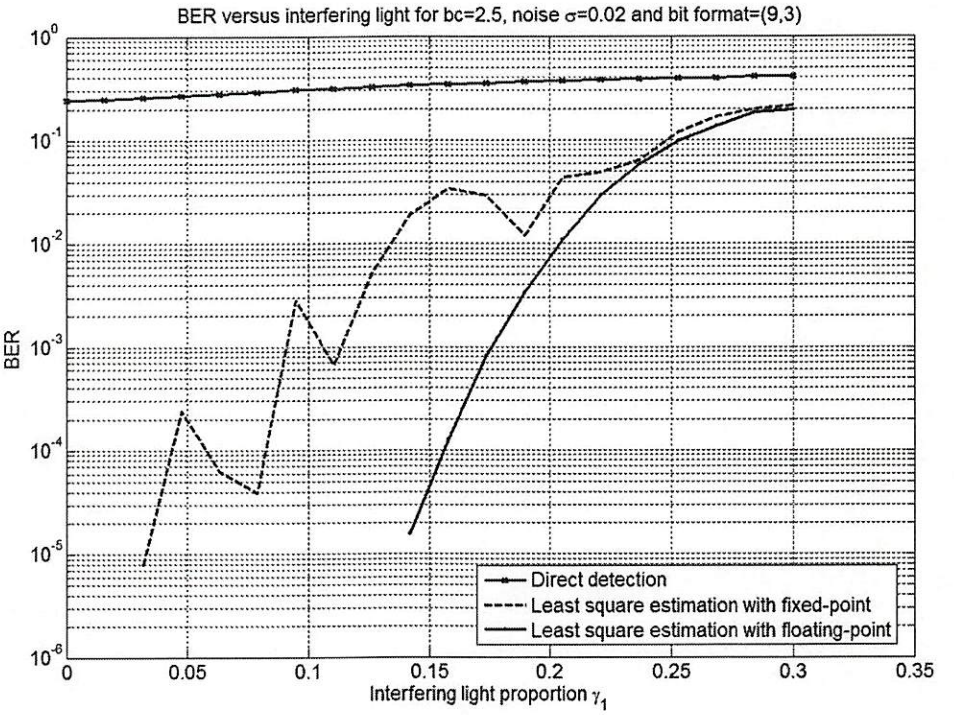



(a)

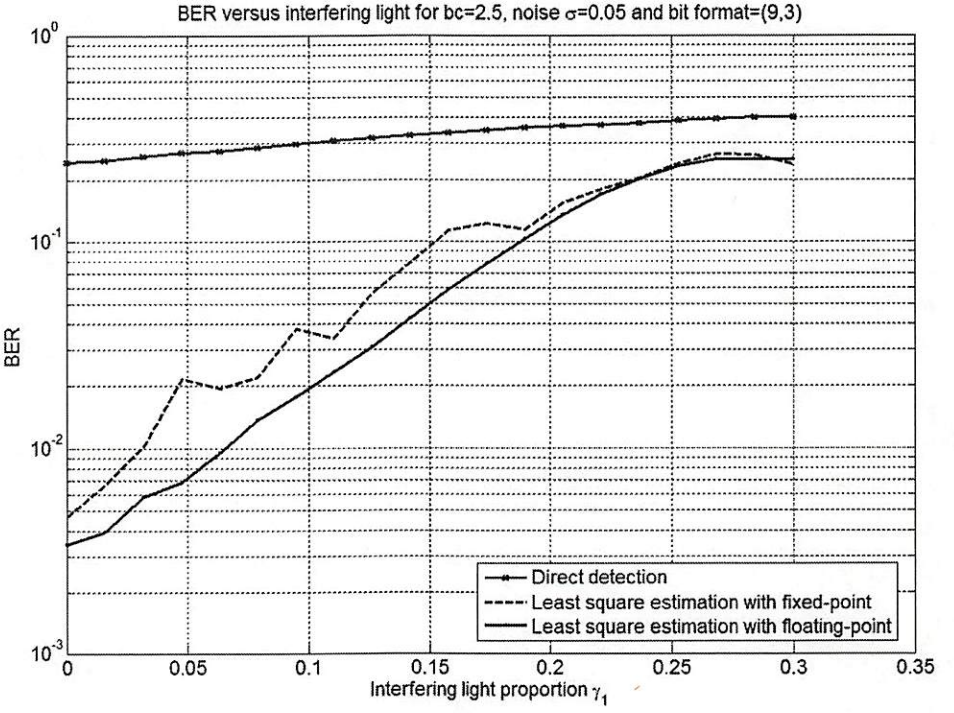



(b)

Figure 8.19. Fixed-point (8, 5) least square equalization simulation results on hexagonal coordinates with $b_c$=2.52 and noise (a) $\sigma$ =0.02 (b) $\sigma$ =0.05.

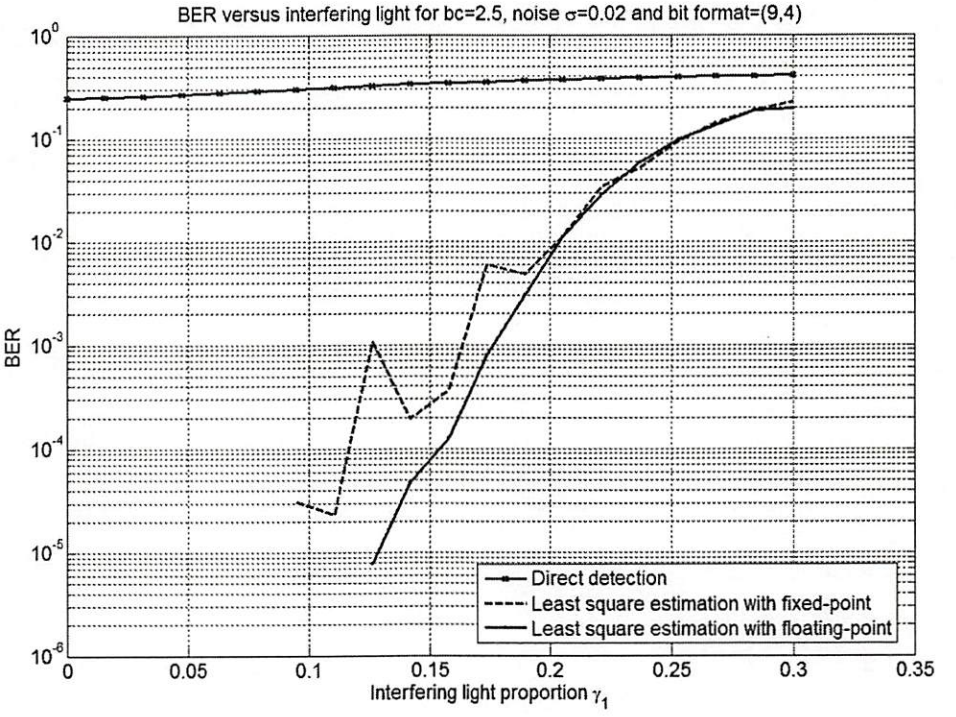



(a)

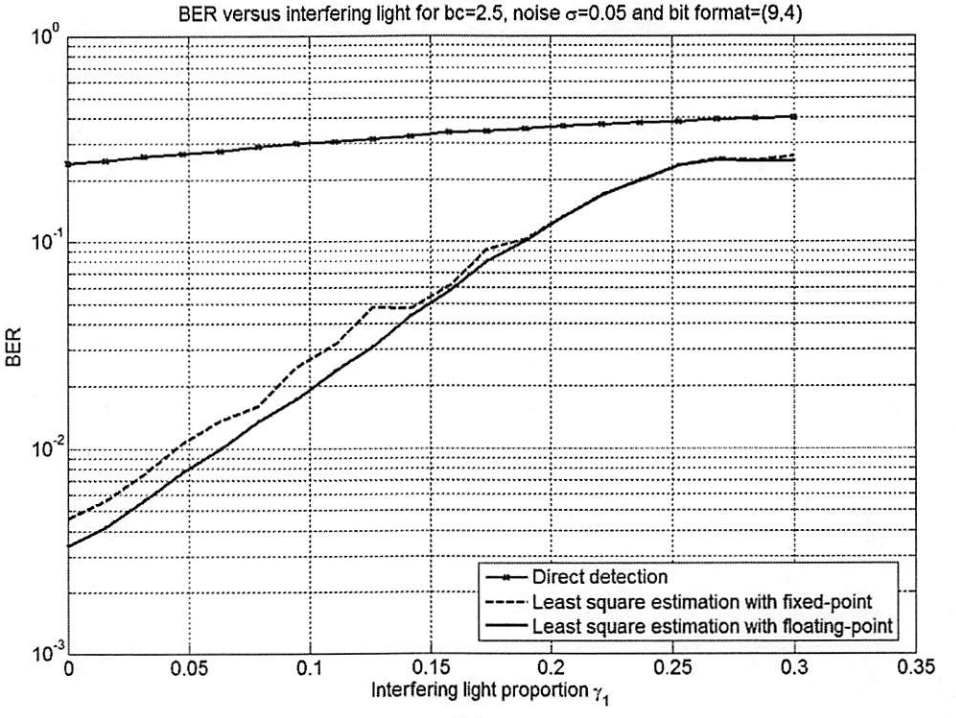



(b)

Figure 8.20. Fixed-point (9, 3) least square equalization simulation results on hexagonal coordinates with $b_c$=2.52 and noise (a) $\sigma$ =0.02 (b) $\sigma$ =0.05.

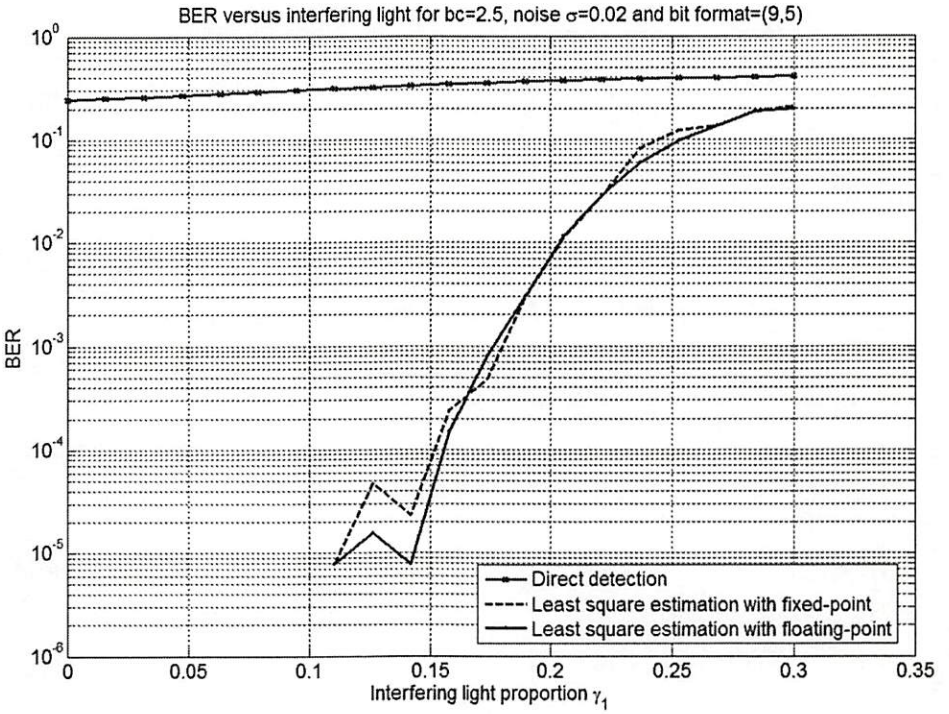



(a)

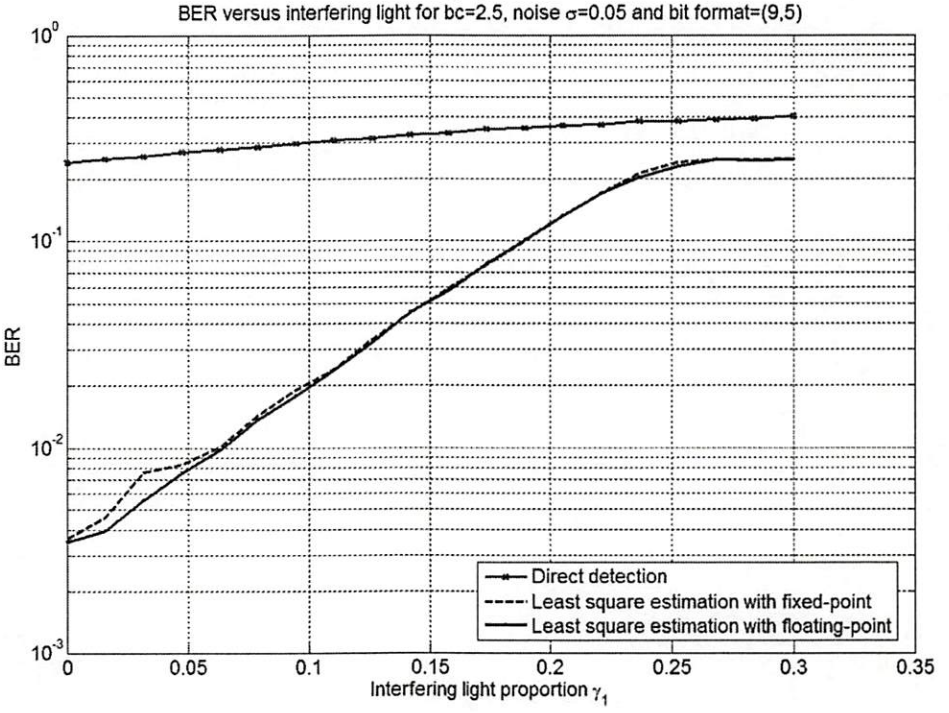



(b)

Figure 8.21. Fixed-point (9, 4) least square equalization simulation results on hexagonal coordinates with $b_c$=2.52 and noise (a) $\sigma$ =0.02 (b) $\sigma$ =0.05.

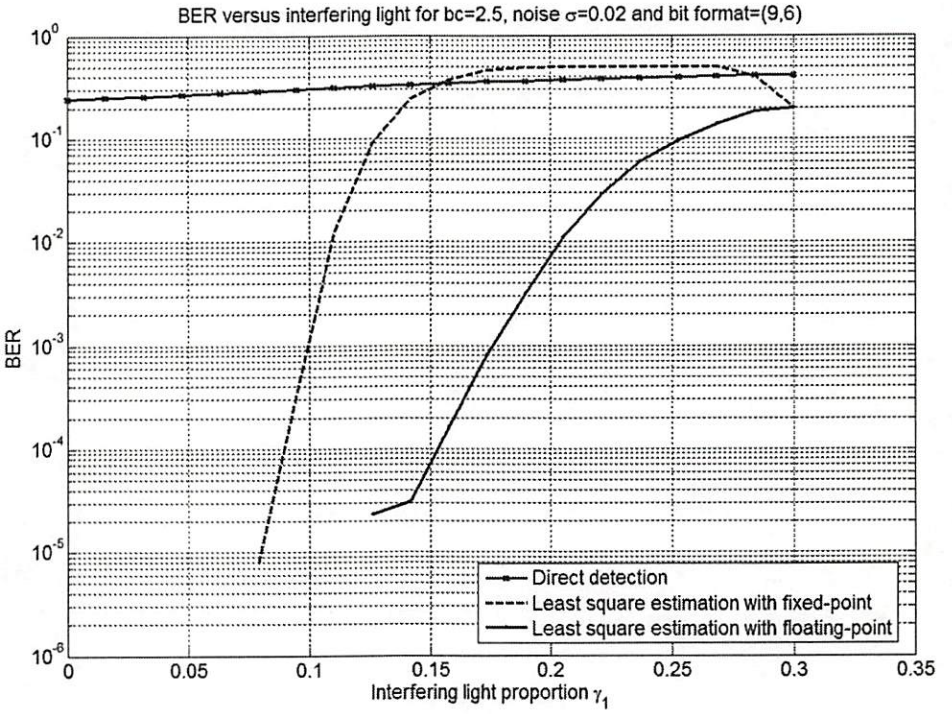



(a)

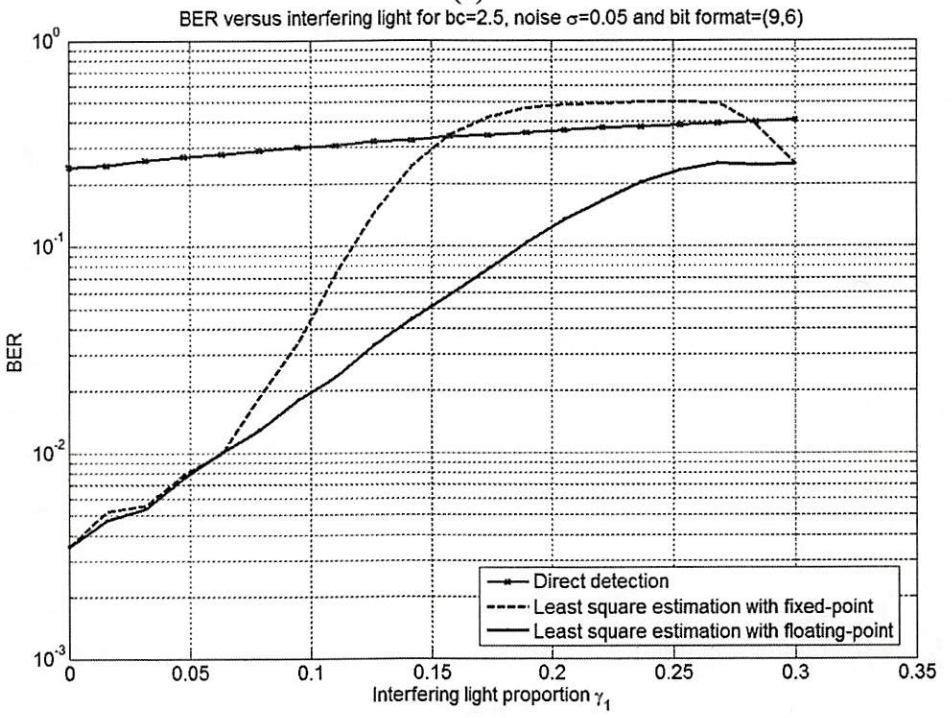



(b)

Figure 8.22. Fixed-point (9, 5) least square equalization simulation results on hexagonal coordinates with $b_c$=2.52 and noise (a) $\sigma$ =0.02 (b) $\sigma$ =0.05.

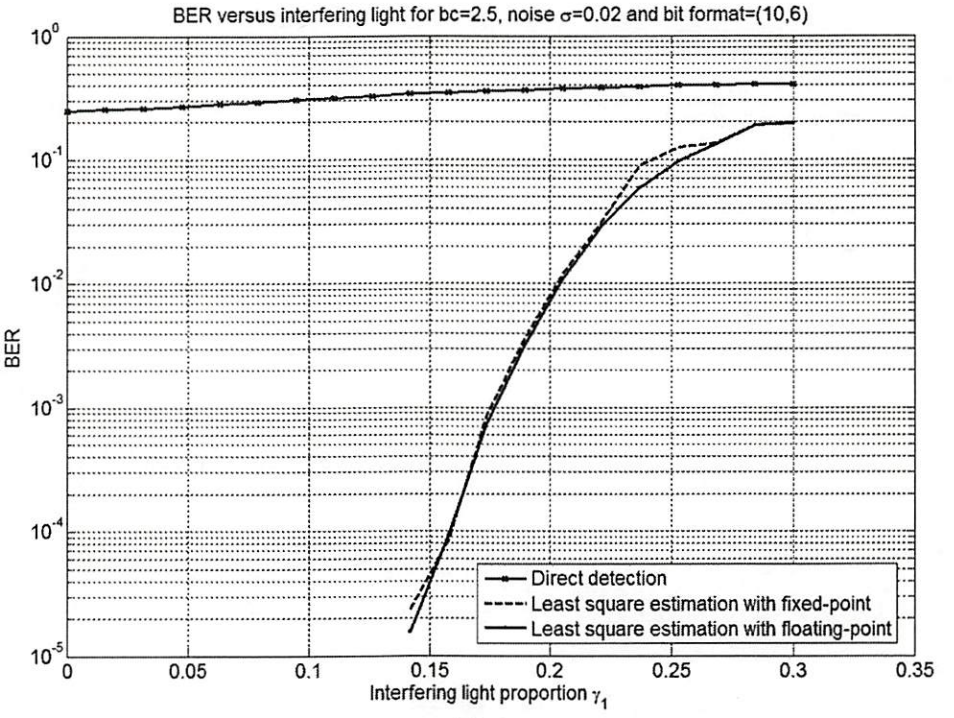



(a)

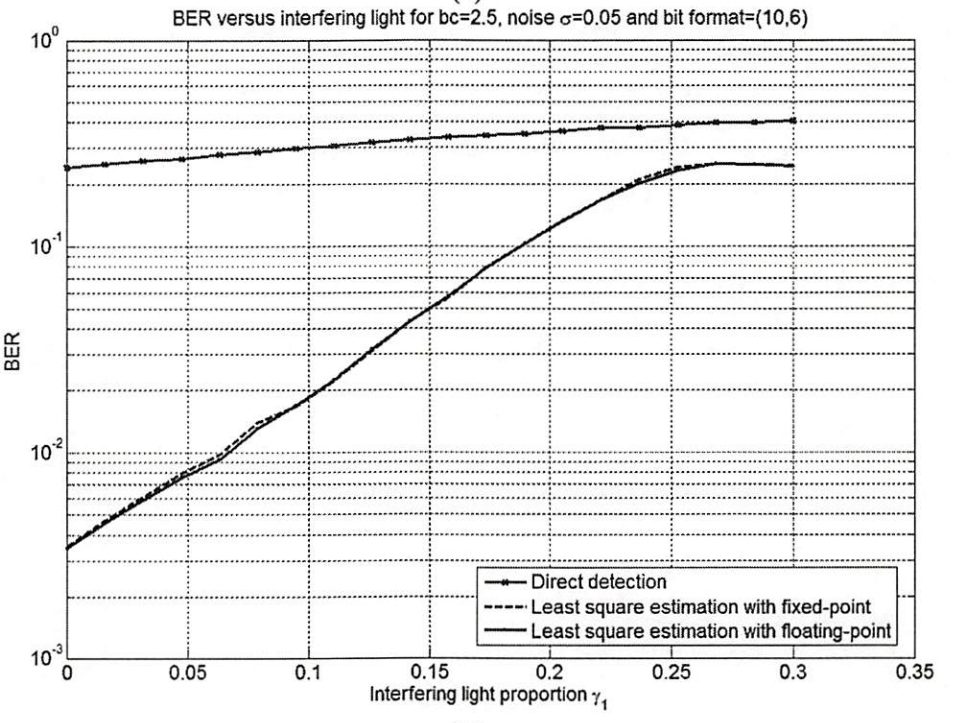



(b)

Figure 8.23. Fixed-point (9, 6) least square equalization simulation results on hexagonal coordinates with $b_c$=2.52 and noise (a) $\sigma$ =0.02 (b) $\sigma$ =0.05.

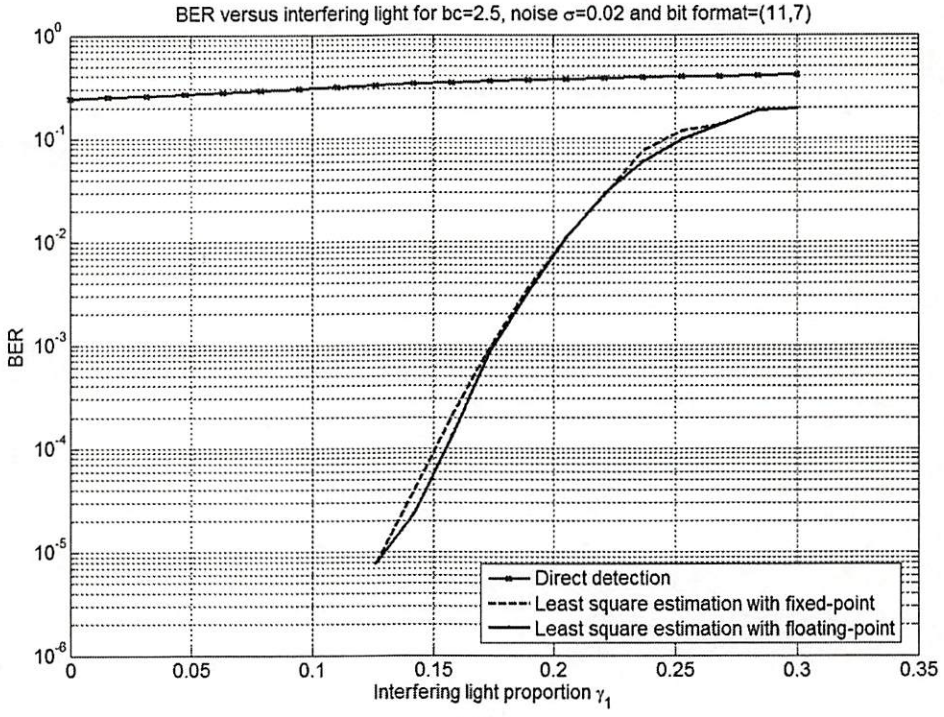



(a)

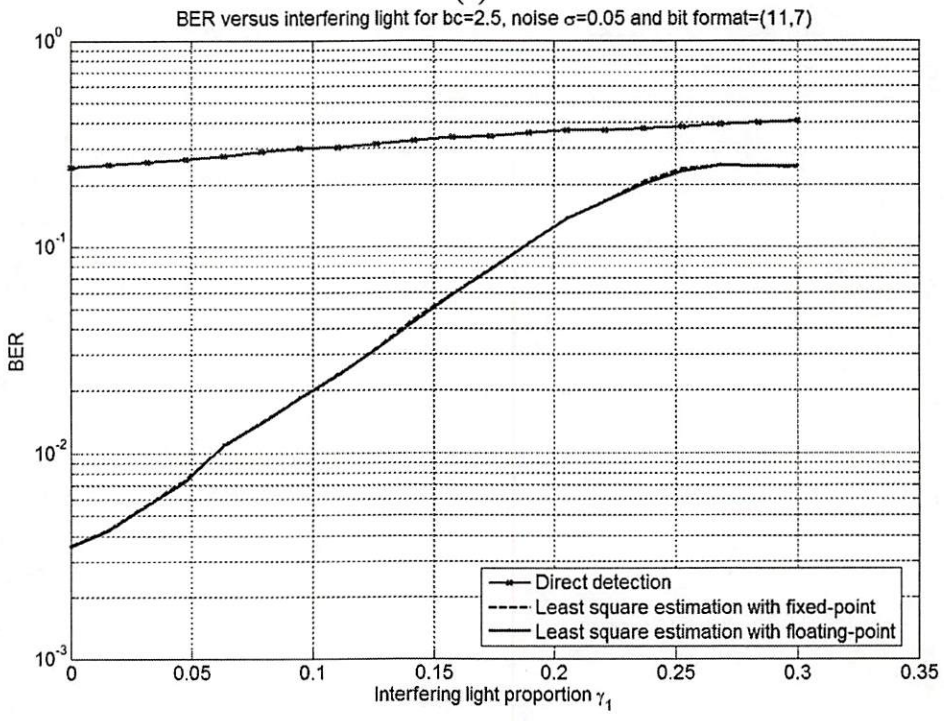



(b)

Figure 8.24. Fixed-point (10, 6) least square equalization simulation results on hexagonal coordinates with $b_c$=2.52 and noise (a) $\sigma$ =0.02 (b) $\sigma$ =0.05.

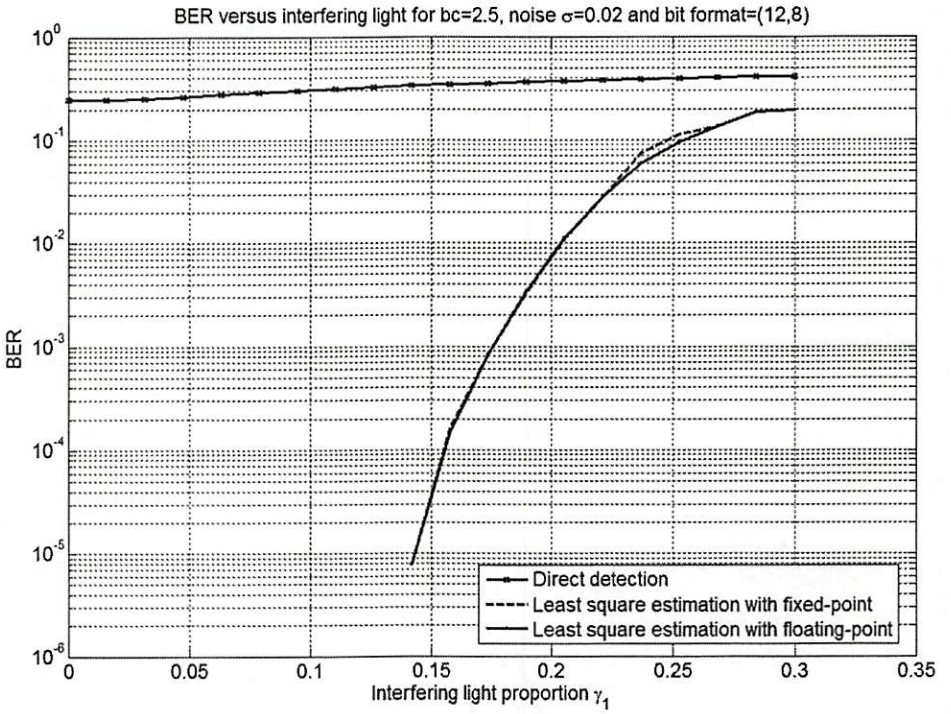



(a)

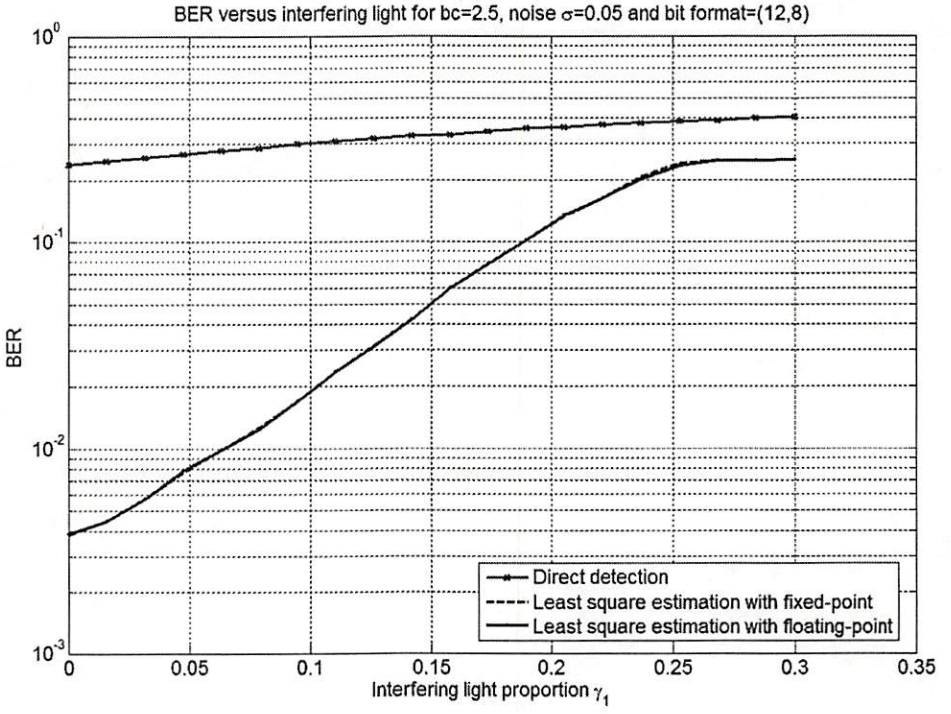



(b)

Figure 8.25. Fixed-point (11, 7) least square equalization simulation results on hexagonal coordinates with $b_c$=2.52 and noise (a) $\sigma$ =0.02 (b) $\sigma$ =0.05.




(a)




(b)

Figure 8.26. Fixed-point (12, 8) least square equalization simulation results on hexagonal coordinates with $b_c$=2.52 and noise (a) $\sigma$ =0.02 (b) $\sigma$ =0.05.

### 8.4.2 Fixed-Point Variable Threshold Method Simulation Results

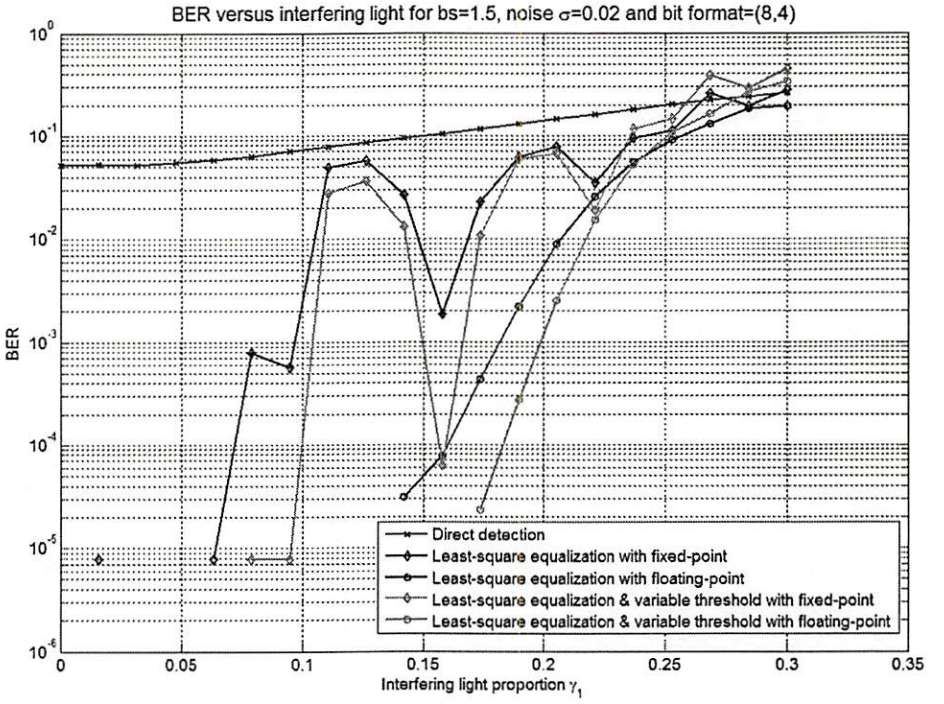
The simulation results for the fixed-point variable threshold method are shown in Figure 8.27 through Figure 8.36 for both rectangular and hexagonal coordinates. Since the quantization of least square equalization is the critical factor for the system performance, we use it as the controlled parameter in our simulations. From the fixed-point least square equalization simulation, we discover the best quantization parameters for the pseudo inverse matrix. We use those known parameters in the quantization process, which enables the least square equalization to generate reliable estimates.

In the simulation results of fixed-point formats (11, 7) and (12, 8), the fixed-point variable threshold method performs almost equally well as the floating-point counterpart. Therefore, using 11-bit fixed-point arithmetic for the variable threshold method is sufficient in terms of bit-error-rate performance. Since the 11-bit fixed-point arithmetic only uses about one sixth of the bits of the 64-bit floating-point arithmetic, it shows a significant saving in memory and computational complexity.

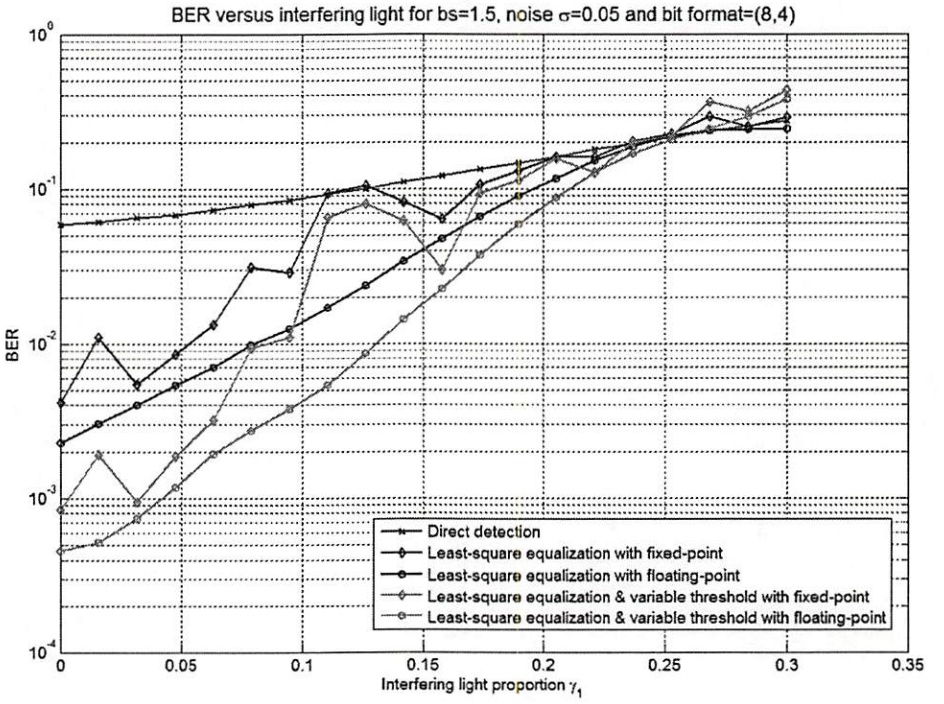
From the simulation results with rectangular coordinates, the (9, 5) fixed-point variable threshold method shows improvement over the 64-bit floating-point least square equalization. In the hexagonal coordinate simulation results, there is an improvement for variable threshold processing over ordinary least square processing even with the (8,4) fixed-point format. Hence, we conclude that the variable threshold method improves the BER almost under all circumstances. Even with the

limited 9-bit fixed-point arithmetic, the variable threshold method achieves better performance than the 64-bit floating-point least square equalization.

Comparing the results in rectangular and hexagonal coordinates, we notice that the system with hexagonal coordinates is more robust to quantization error than the system with rectangular coordinates as shown in the fixed-point least square equalization. Therefore, the hexagonal coordinate grid has potential to implement fixed-point algorithm with fewer bits. This translates into reduced memory requirements and hardware complexity.

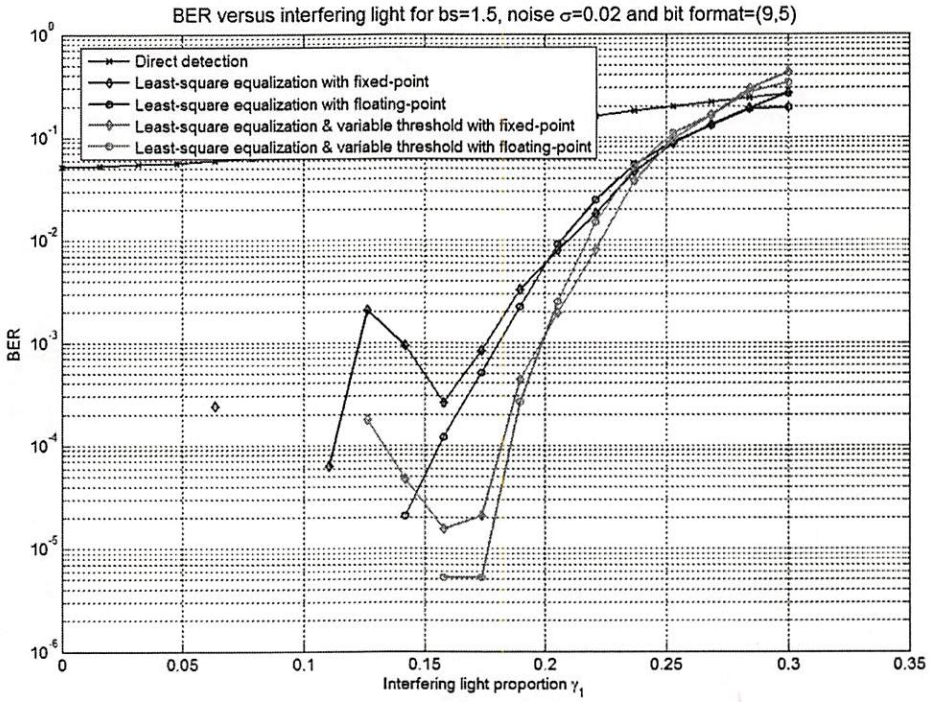



(a)

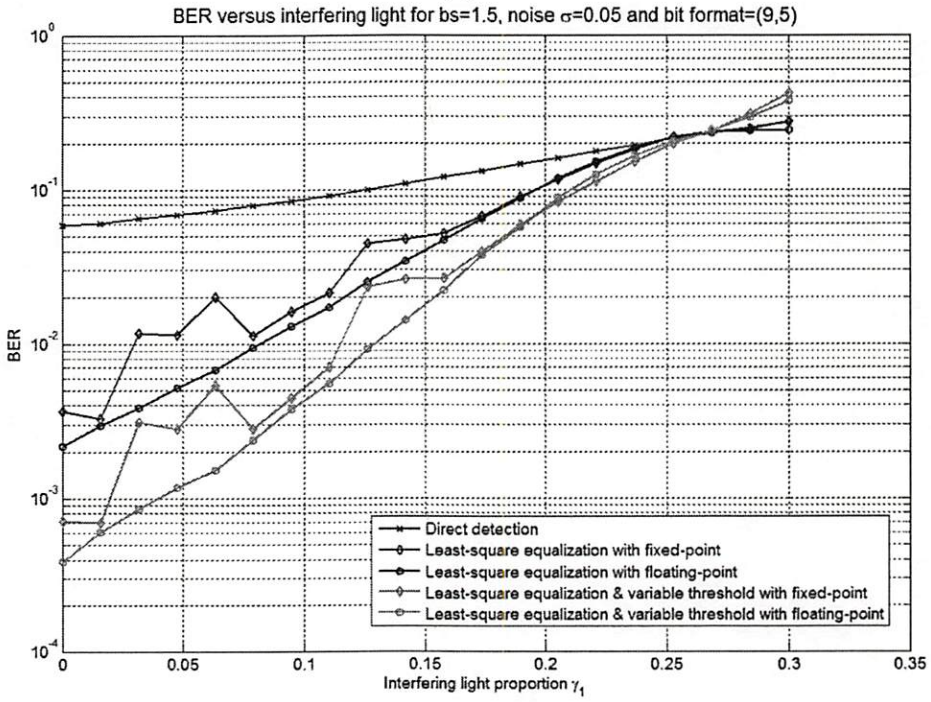



(b)

Figure 8.27. Fixed-point (8, 4) variable threshold method simulation results on rectangular coordinates with $b_s$=1.5 and noise (a) $\sigma$ =0.02 (b) $\sigma$ =0.05.

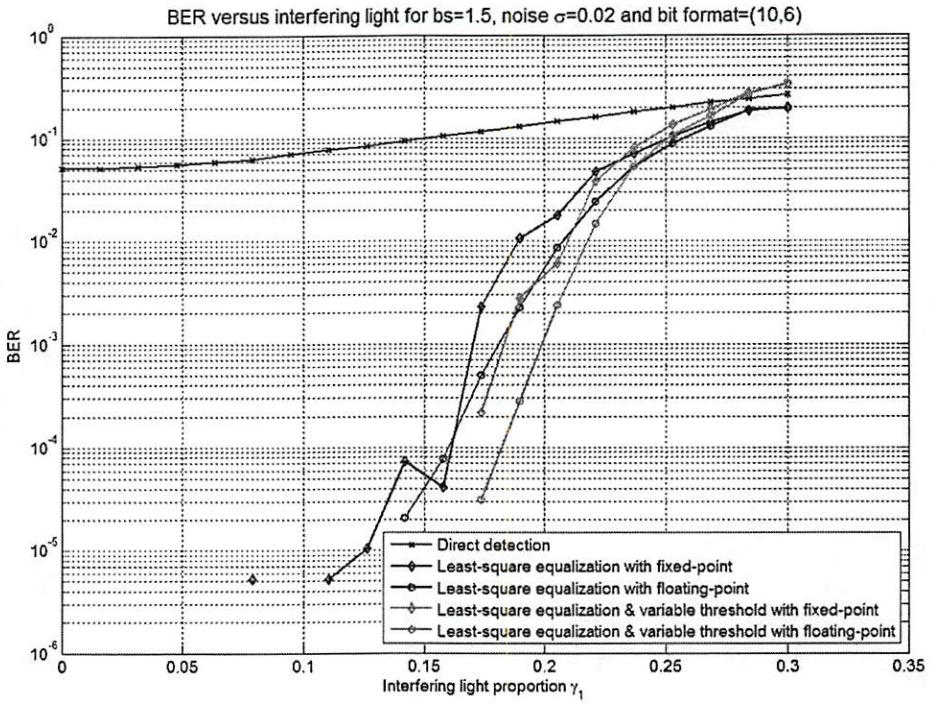



(a)

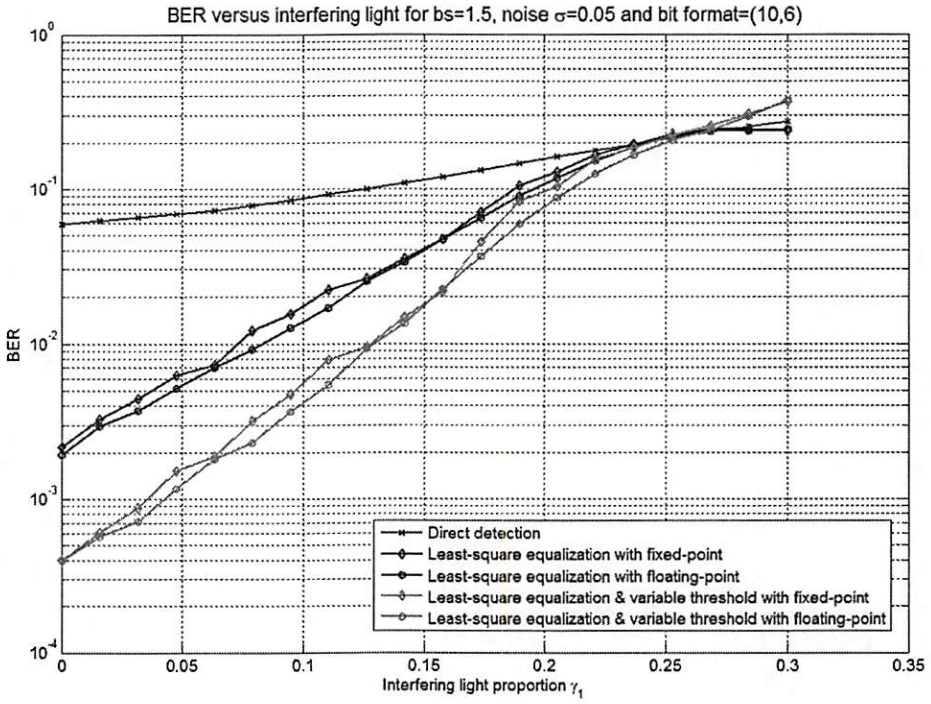



(b)

Figure 8.28. Fixed-point (9, 5) variable threshold method simulation results on rectangular coordinates with $b_s$=1.5 and noise (a) $\sigma$ =0.02 (b) $\sigma$ =0.05.

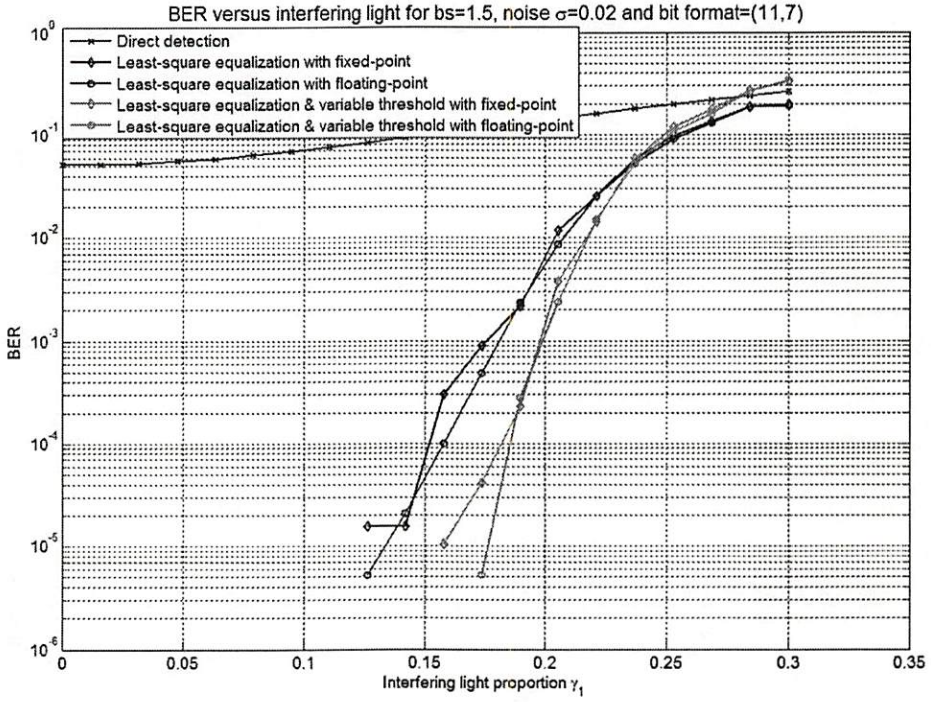



(a)

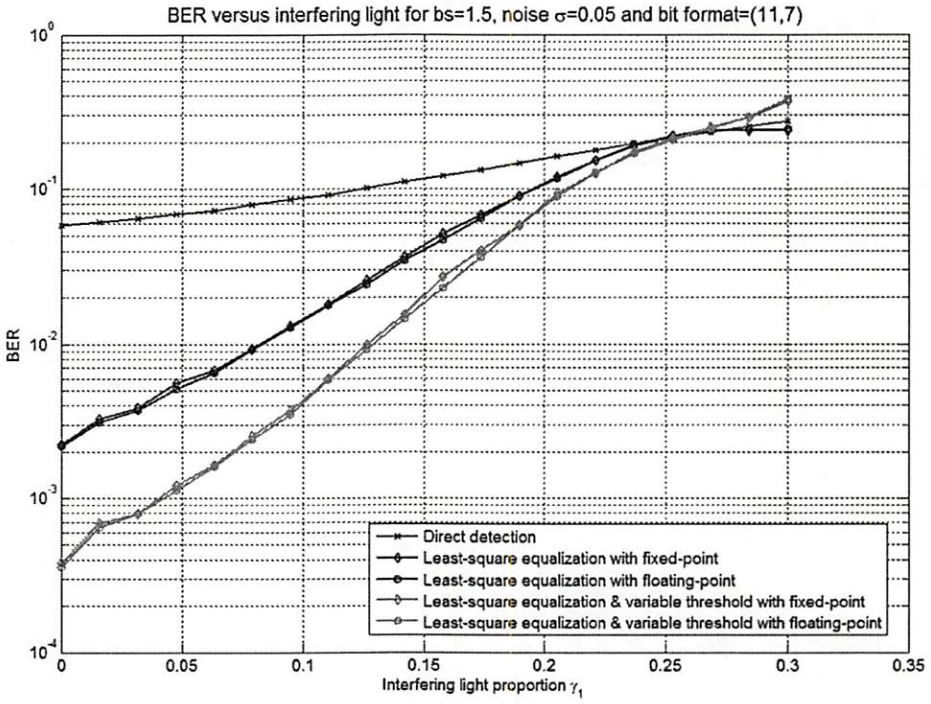



(b)

Figure 8.29. Fixed-point (10, 6) variable threshold method simulation results on rectangular coordinates with $b_s$=1.5 and noise (a) $\sigma$ =0.02 (b) $\sigma$ =0.05.

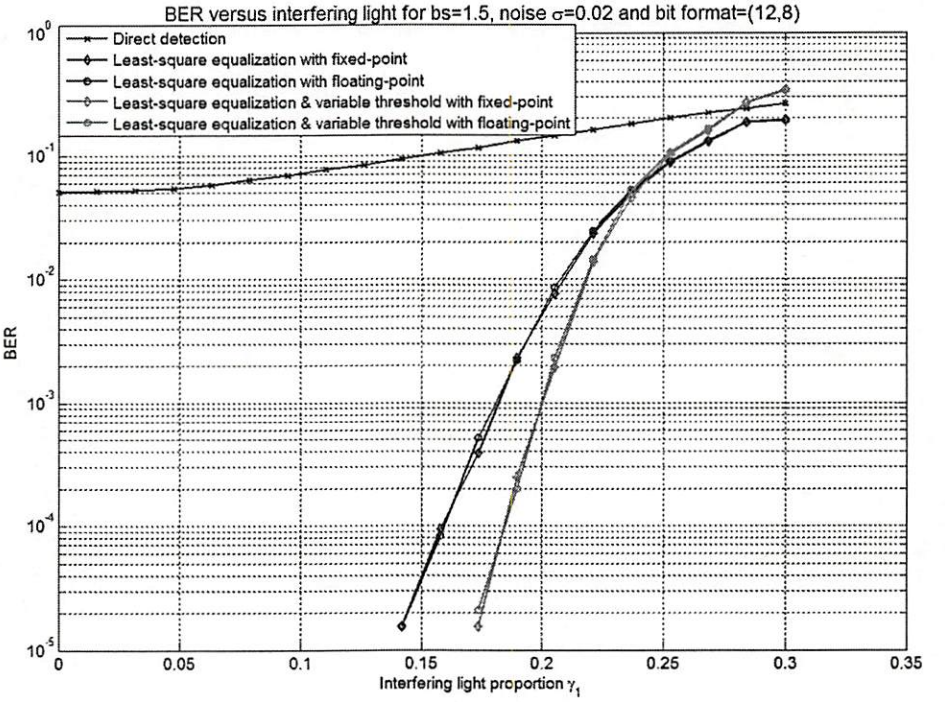



(a)

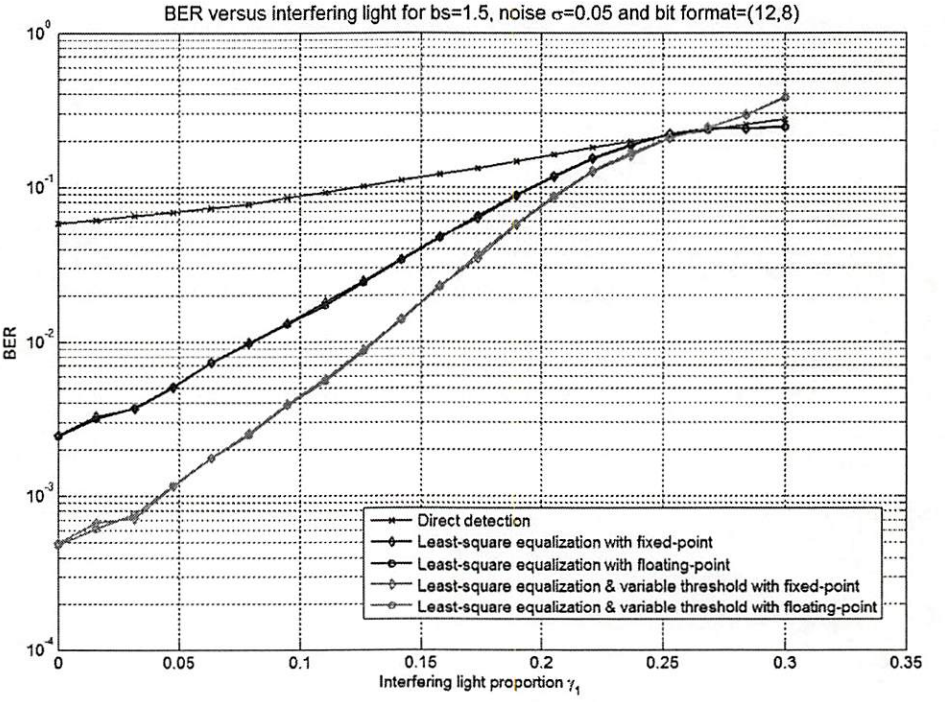



(b)

Figure 8.30. Fixed-point (11, 7) variable threshold method simulation results on rectangular coordinates with $b_s$=1.5 and noise (a) $\sigma$ =0.02 (b) $\sigma$ =0.05.

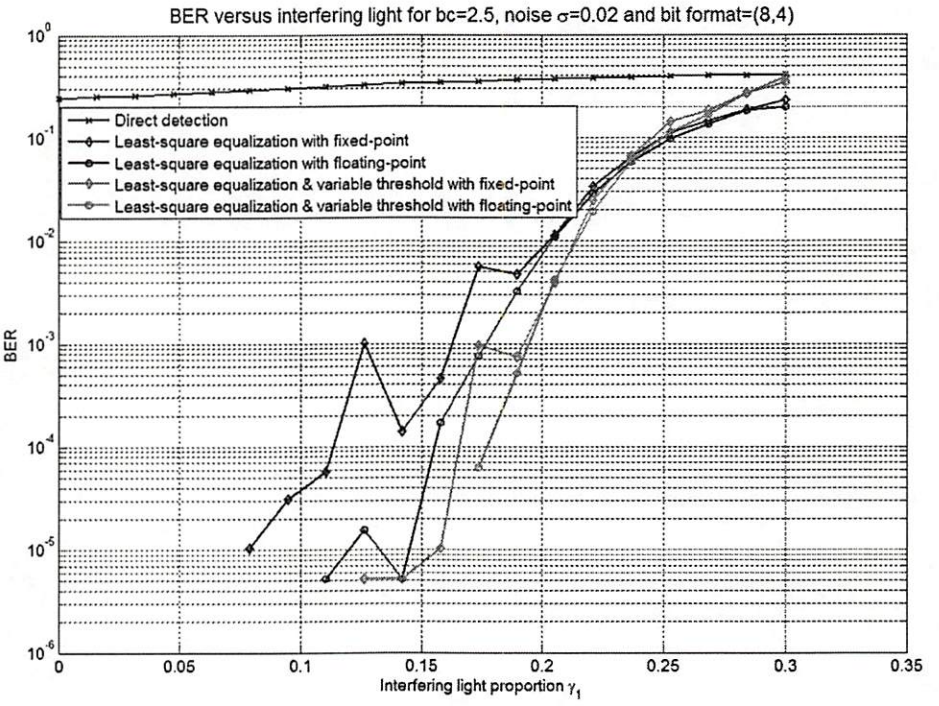



(a)

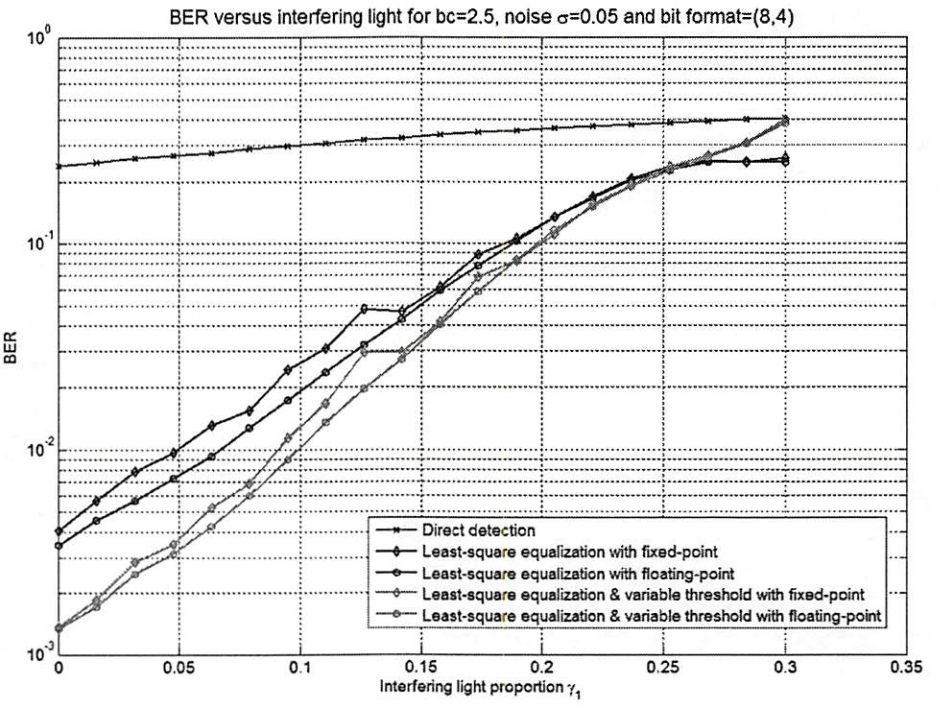



(b)

Figure 8.31. Fixed-point (12, 8) variable threshold method simulation results on rectangular coordinates with $b_s$=1.5 and noise (a) $\sigma$ =0.02 (b) $\sigma$ =0.05.

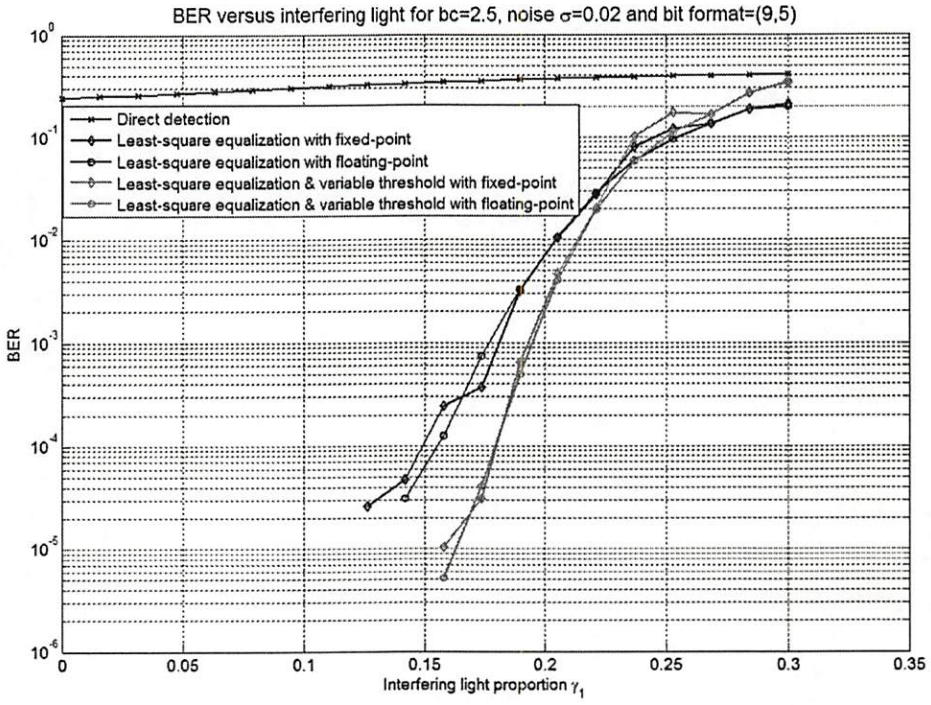



(a)

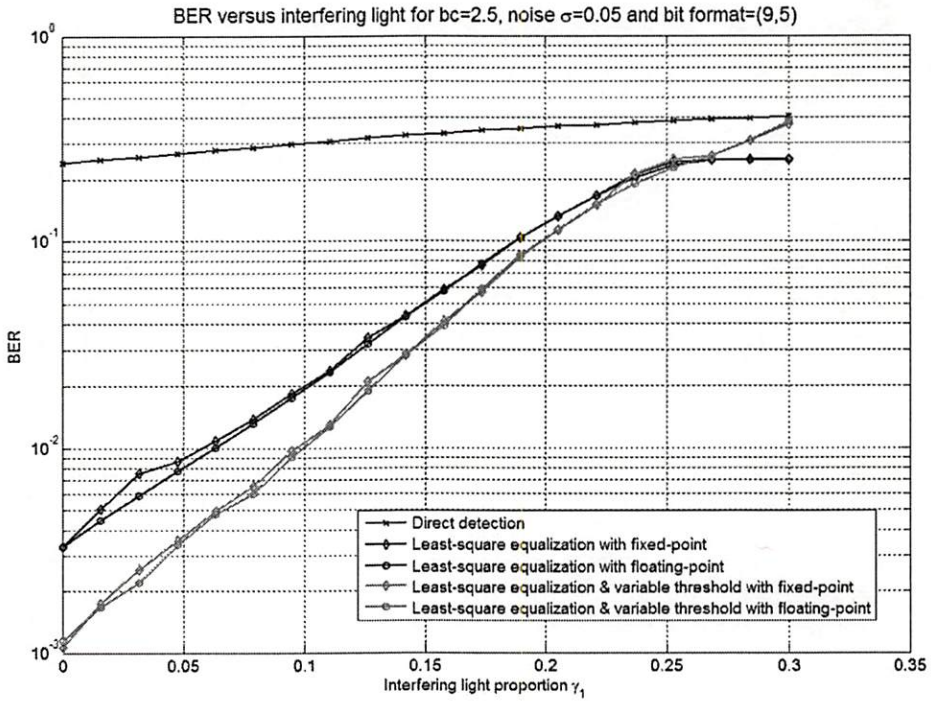



(b)

Figure 8.32. Fixed-point (8, 4) variable threshold method simulation results on hexagonal coordinates with $b_c$=2.52 and noise (a) $\sigma$=0.02 (b) $\sigma$=0.05.

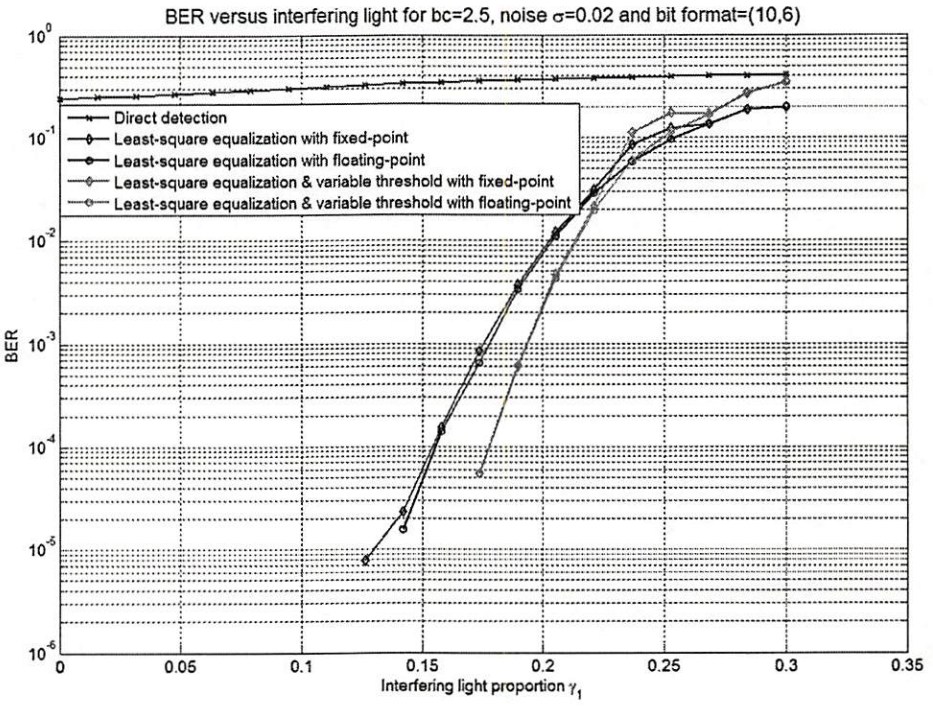



(a)

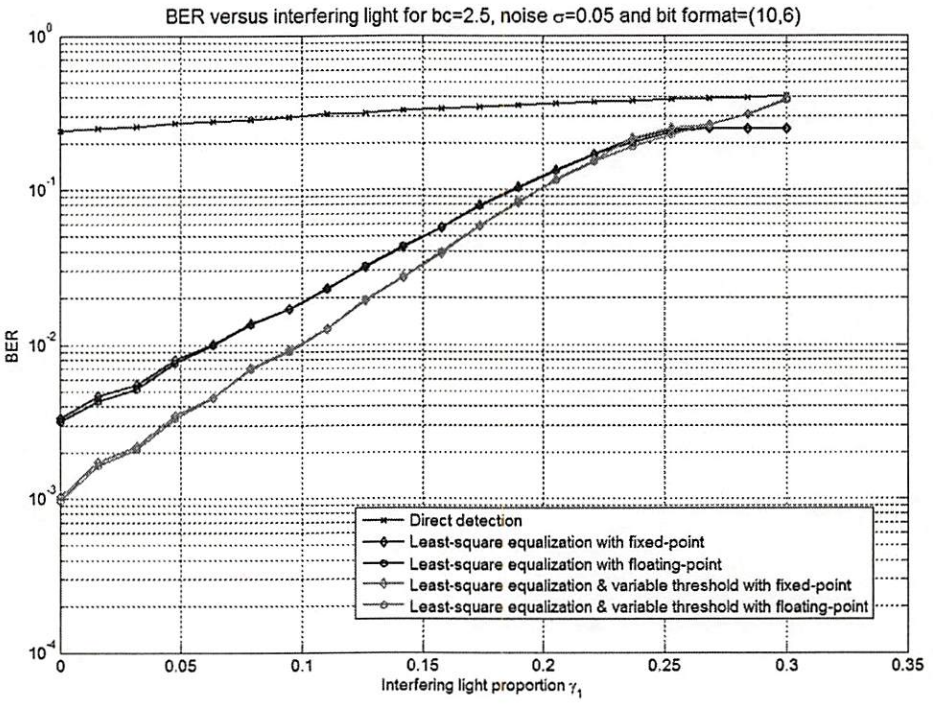



(b)

Figure 8.33. Fixed-point (9, 5) variable threshold method simulation results on hexagonal coordinates with $b_c$=2.52 and noise (a) $\sigma$ =0.02 (b) $\sigma$ =0.05.

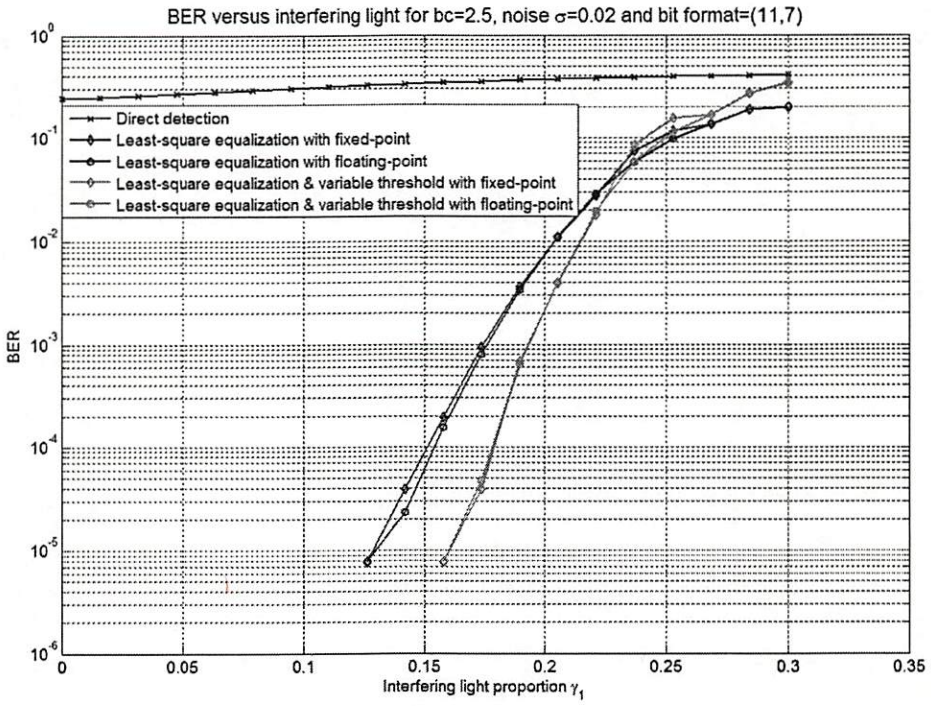



(a)

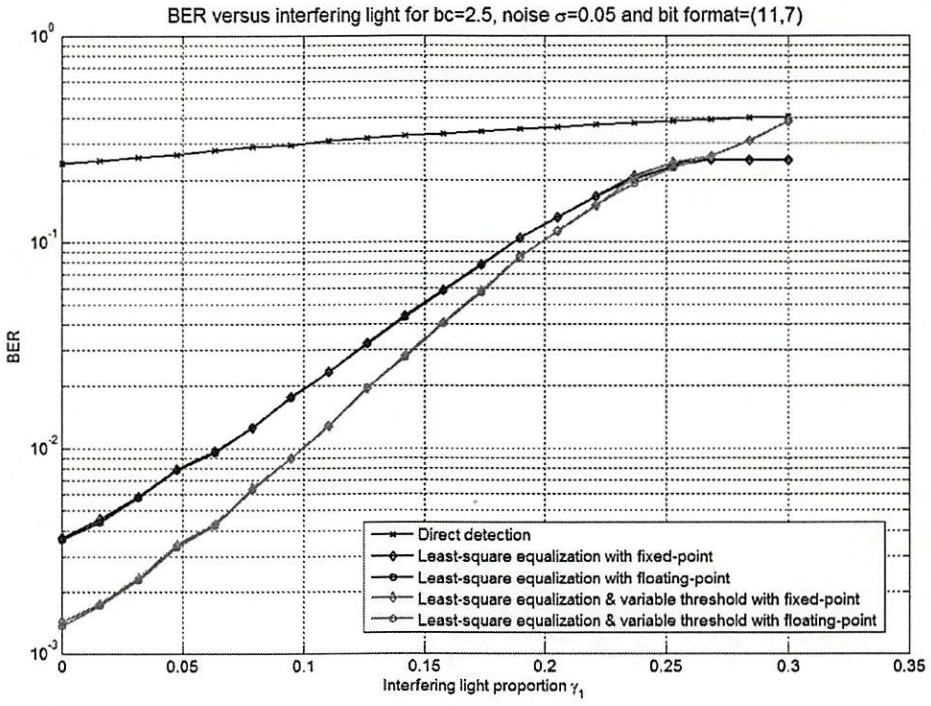



(b)

Figure 8.34. Fixed-point (10, 6) variable threshold method simulation results on hexagonal coordinates with $b_c$=2.52 and noise (a) $\sigma$ =0.02 (b) $\sigma$ =0.05.

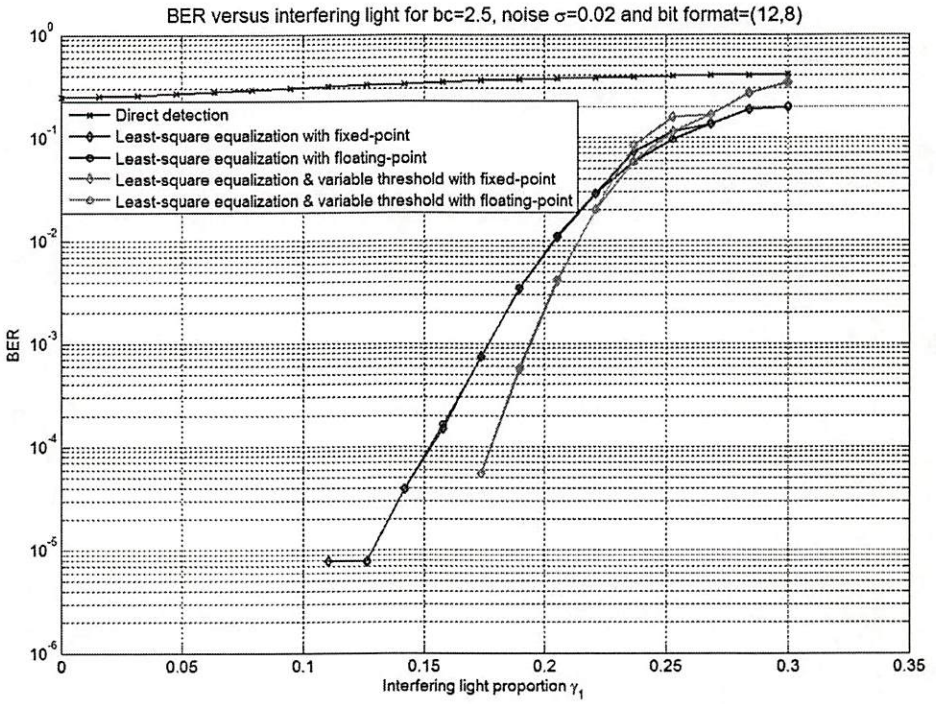



(a)

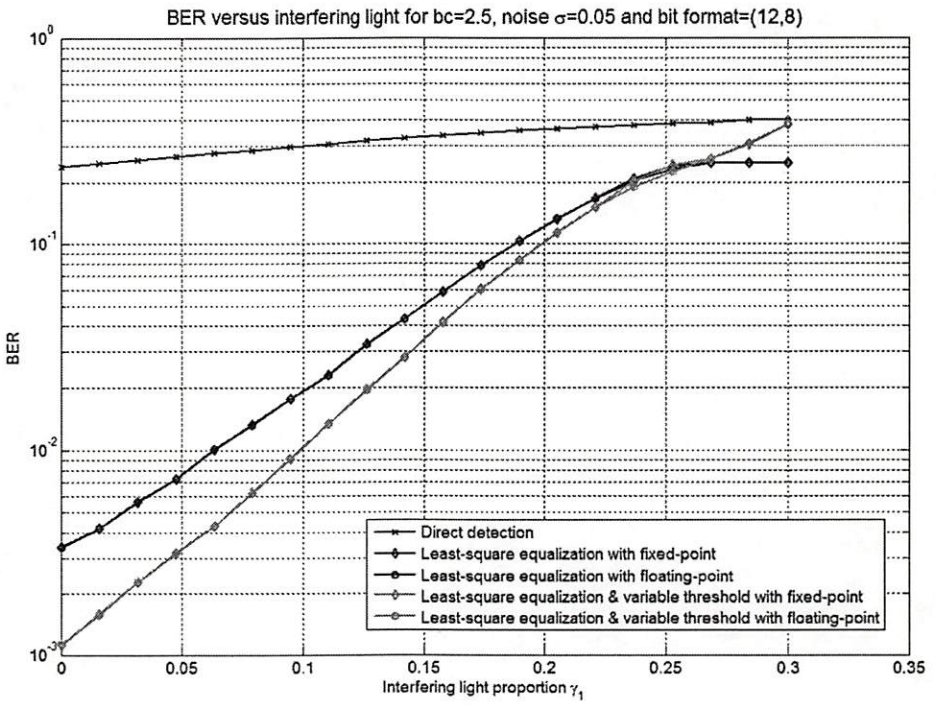



(b)

Figure 8.35. Fixed-point (11, 7) variable threshold method simulation results on hexagonal coordinates with $b_c$=2.52 and noise (a) $\sigma$ =0.02 (b) $\sigma$ =0.05.




(a)




(b)

Figure 8.36. Fixed-point (12, 8) variable threshold method simulation results on hexagonal coordinates with $b_c$=2.52 and noise (a) $\sigma$ =0.02 (b) $\sigma$ =0.05.

# Chapter 9 Pixel Misregistration in Three-Dimensional Page-Oriented Data Storage (3D PODS) System

## 9.1 Pixel Misregistration

In the readout system of the 3D PODS, any relative horizontal or vertical movement between the data page and the detector causes pixel misregistration. In previous chapters, we assume the readout system is perfectly aligned and that no any pixel misregistration exists. However, when pixel misregistration occurs, the received signal at the detector plane is partially lost. In addition, unwanted interference from surrounding neighbors further degrades the received signal. Therefore, pixel misregistration often causes severe signal degradation and erroneous data retrieval.

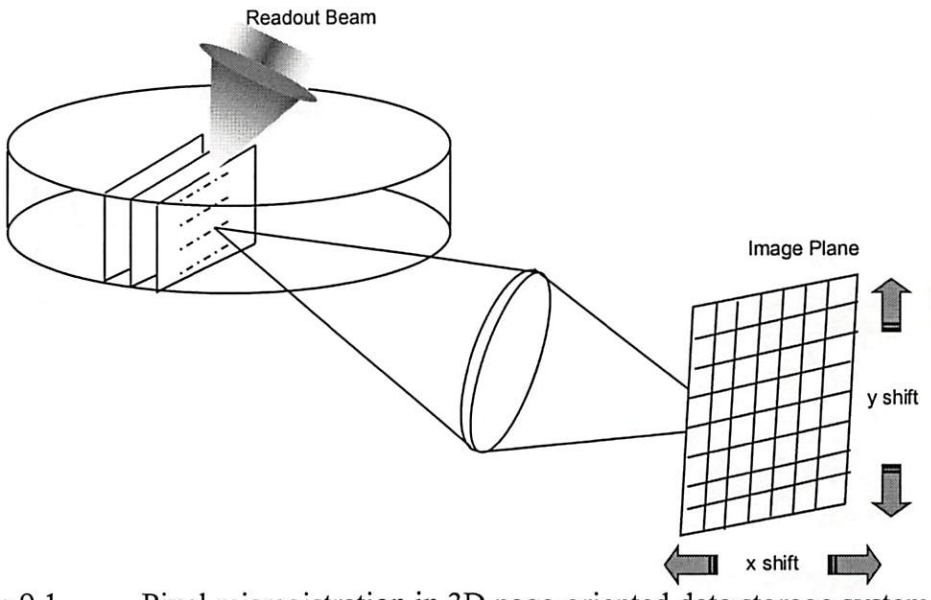



Figure 9.1. Pixel misregistration in 3D page-oriented data storage system.

Figure 9.1 shows the pixel misregistration in 3D page-oriented data storage system with both horizontal ($x$ direction) and vertical ($y$ direction) shift. The readout image of the image plane is simulated in Figure 9.2 for comparison. The image on right hand side in Figure 9.2 shows shift in both $x$ and $y$ directions. From Figure 9.2, we notice that most of the received signal lands outside the designated detector cell. In this case, the conventional fixed threshold method has difficulties in retrieving data and yields a high error rate.

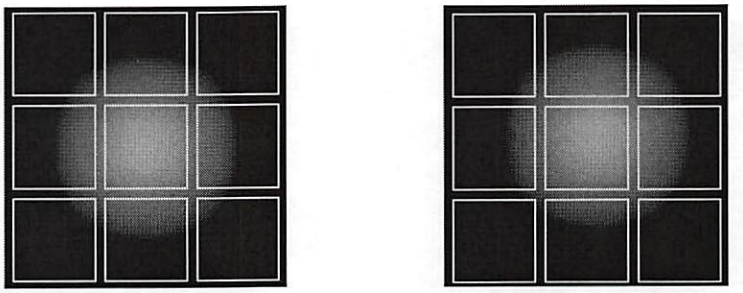



Figure 9.2. Comparison of detector plane images with and without pixel misregistration.

In this chapter, we apply least square equalization and variable threshold method to 3D PODS systems with pixel misregistration. In our method, we use knowledge of the PSF and the $(x,y)$ shift to formulate a new imaging model matrix. Then we substitute the new imaging model for the convolution matrix **H** in Eq. (3.19). However, due to the shift in pixels, matrix **H** loses symmetry property. We use this asymmetry matrix **H** directly in existing algorithm. Using this new PSF matrix, our

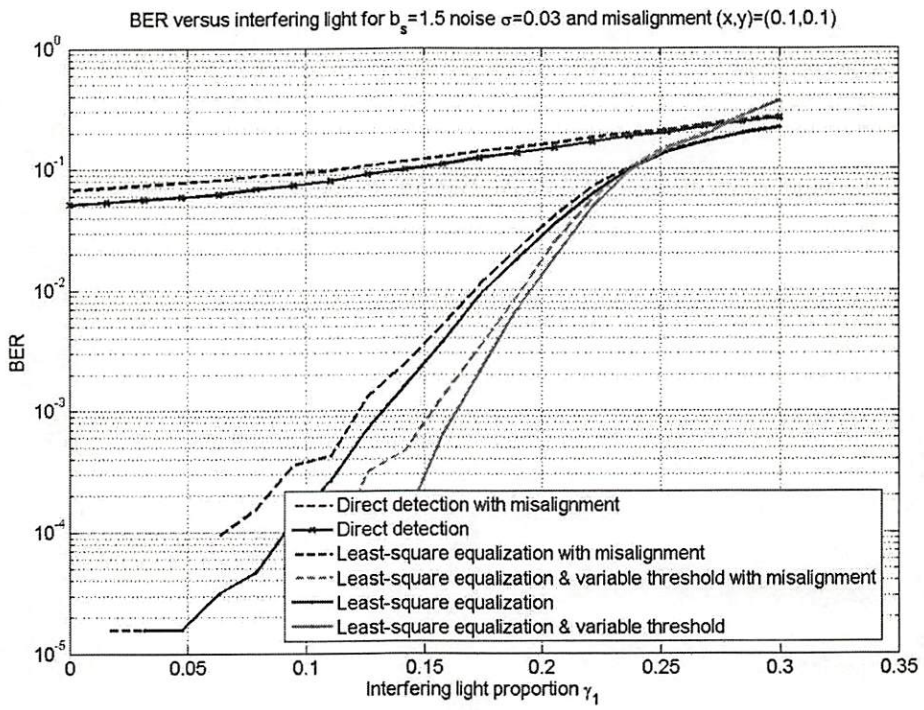
estimation methods are able to compensate for the misregistration effects and recover the data.

### 9.2 Simulation Results

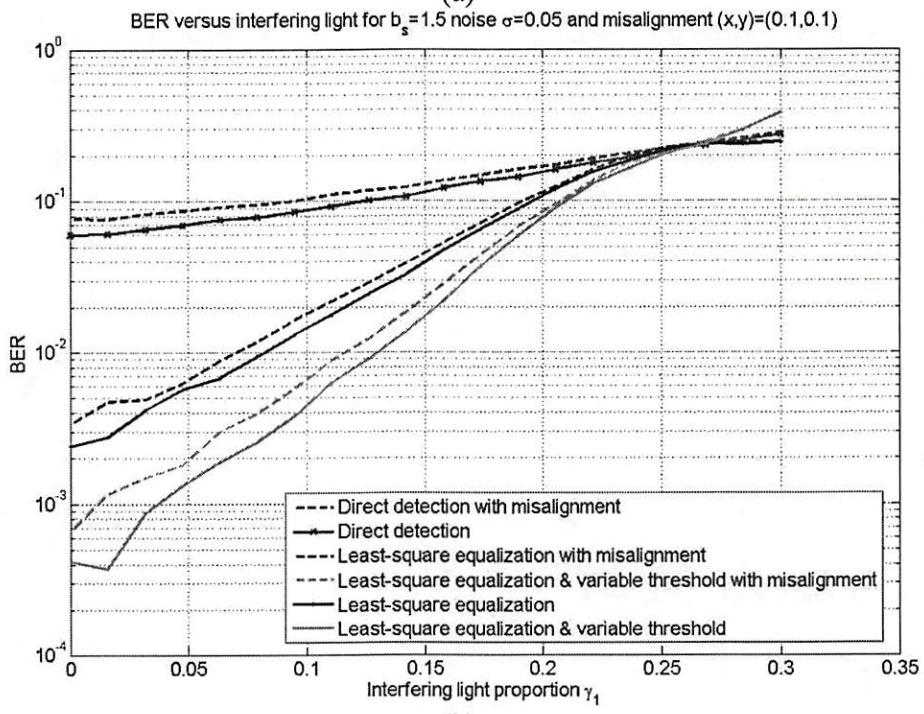
Figure 9.3 to Figure 9.7 show the simulation results of 3D PODS system using both least square equalization and variable threshold methods. The amount of shift is assumed equal in $x$ and $y$ and is expressed as a percentage of the unit spacing between detector cells. We show the results of different amounts of shift from 10% to 50%. Since all the detector cells are identical and located in a periodic array, any misregistration larger than 50% is the same as a misregistration of less than 50% relative to the closest cell.

From the simulation results, our methods show a significant improvement when the amount of shift is below 30%. Any practical system will be carefully designed and calibrated, so that the amount of misregistration shift is generally less than 30%. Our method reduces the effective misregistration to a much smaller amount, and further improves the system reliability in the presence of pixel misregistration.

Comparing the bit error rates of least square equalization and variable threshold methods, we notice that the variable threshold method performs well when the amount of misregistration is small. For larger amounts of shift, the least square equalization cannot provide a good data estimation for the variable threshold method, and it makes incorrect decisions based on erroneous information.

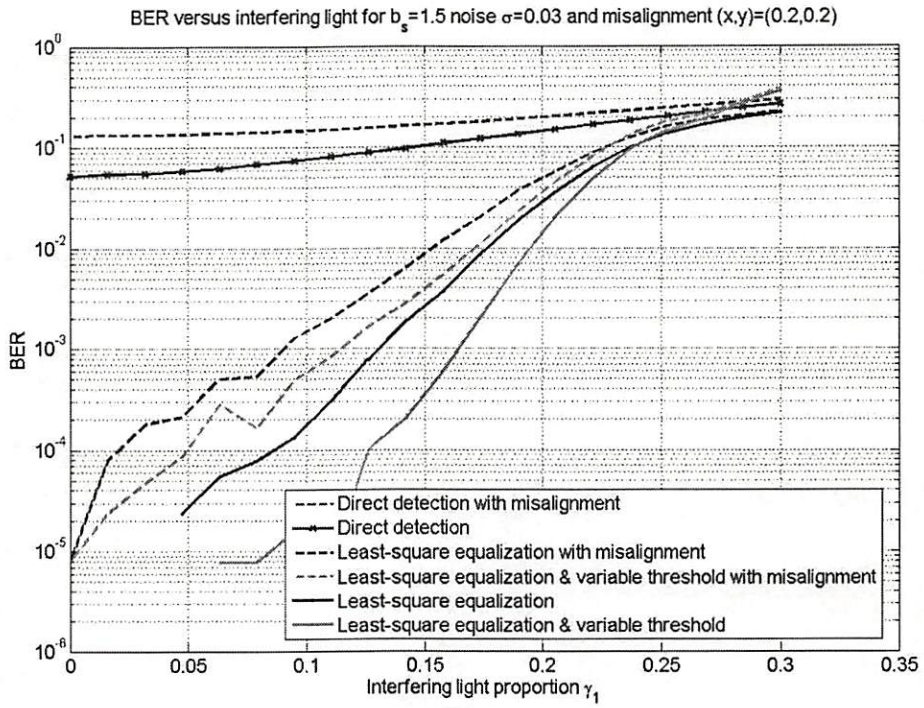



(a)

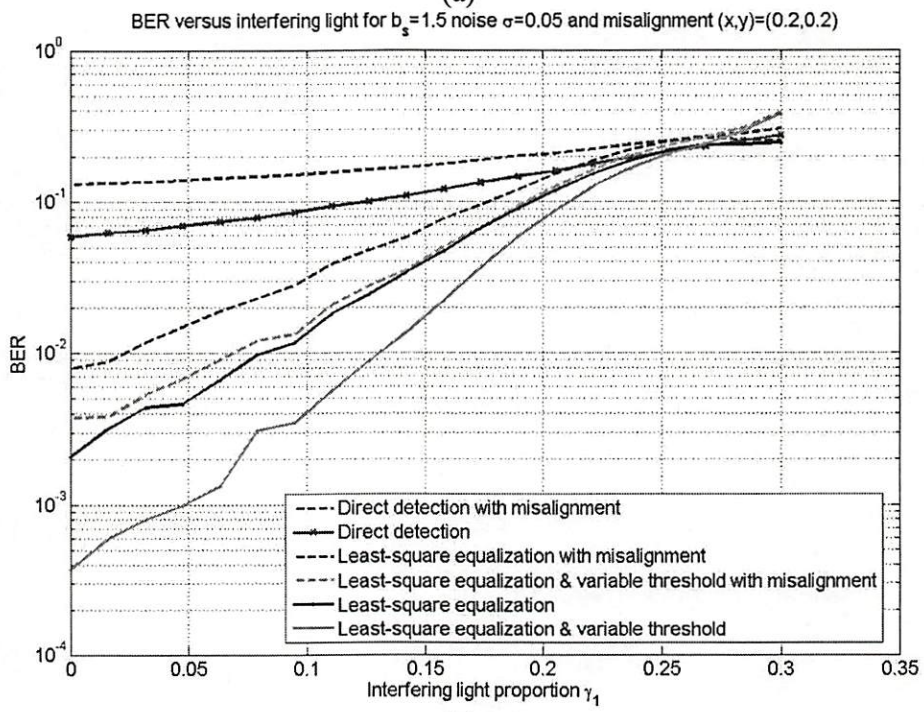



(b)

Figure 9.3. Pixel misregistration simulation results on rectangular coordinates with $b_s$=1.5, shift (x,y)=(0.1,0.1) and noise (a) $\sigma$=0.03 (b) $\sigma$=0.05.

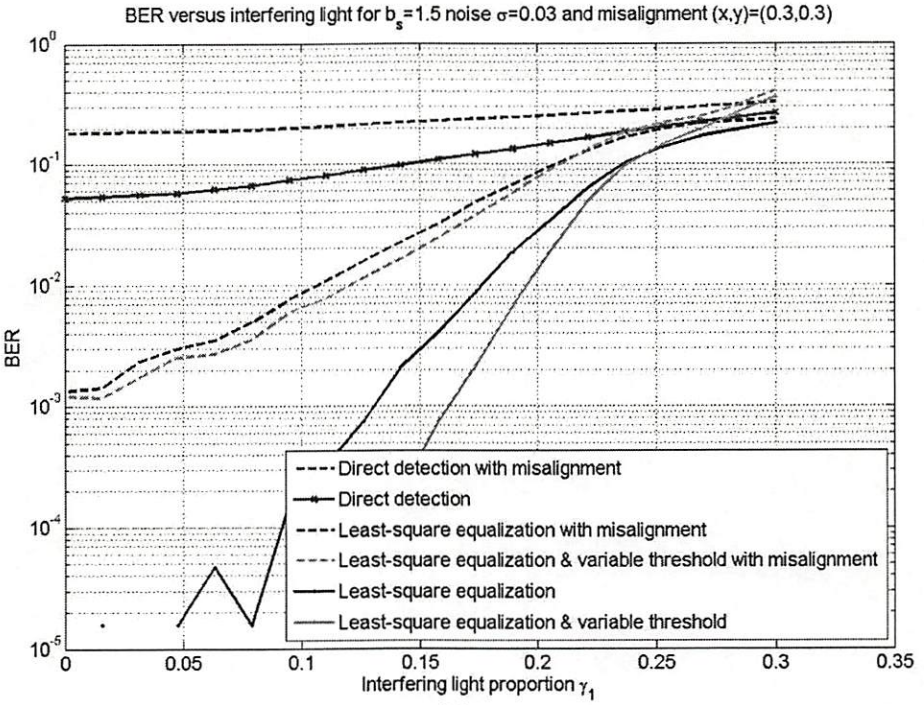



(a)

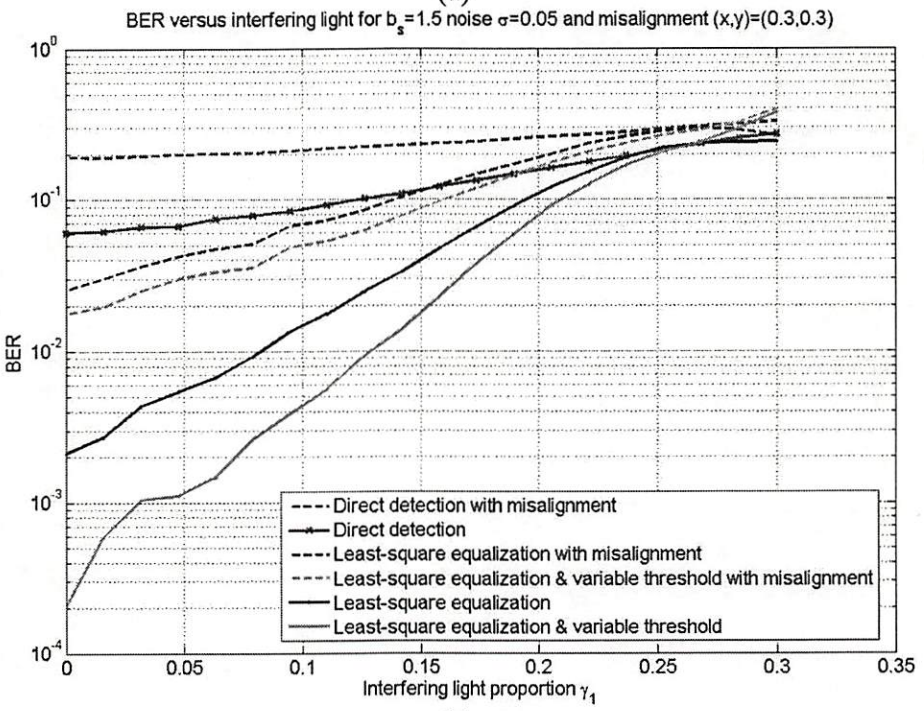



(b)

Figure 9.4. Pixel misregistration simulation results on rectangular coordinates with $b_s$=1.5, shift (x,y)=(0.2,0.2) and noise (a) $\sigma$=0.03 (b) $\sigma$=0.05.

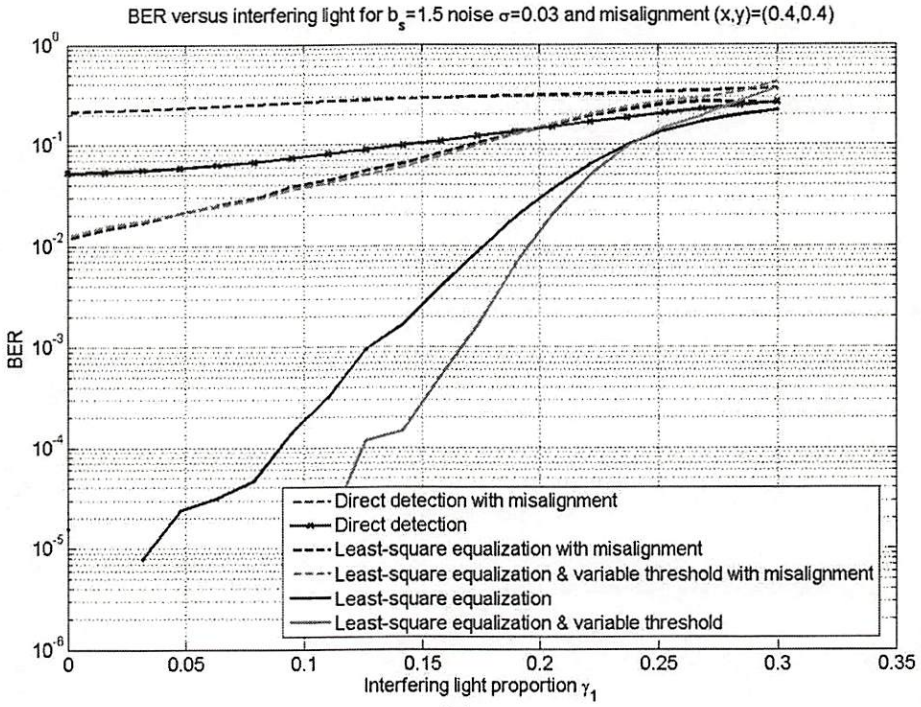



(a)

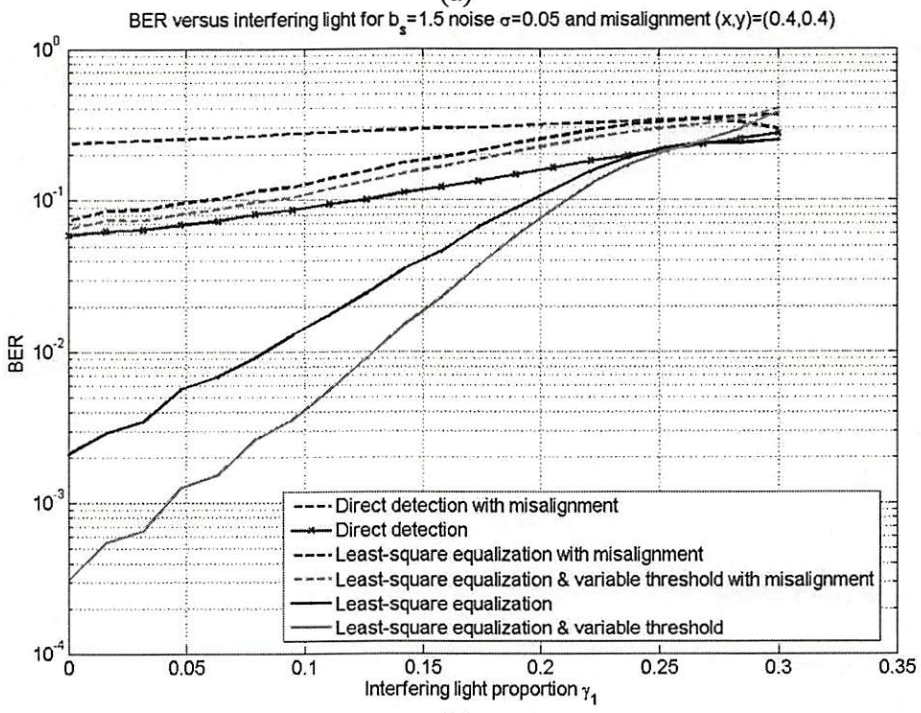



(b)

Figure 9.5. Pixel misregistration simulation results on rectangular coordinates with $b_s$=1.5, shift (x,y)=(0.3,0.3) and noise (a) $\sigma$=0.03 (b) $\sigma$=0.05.

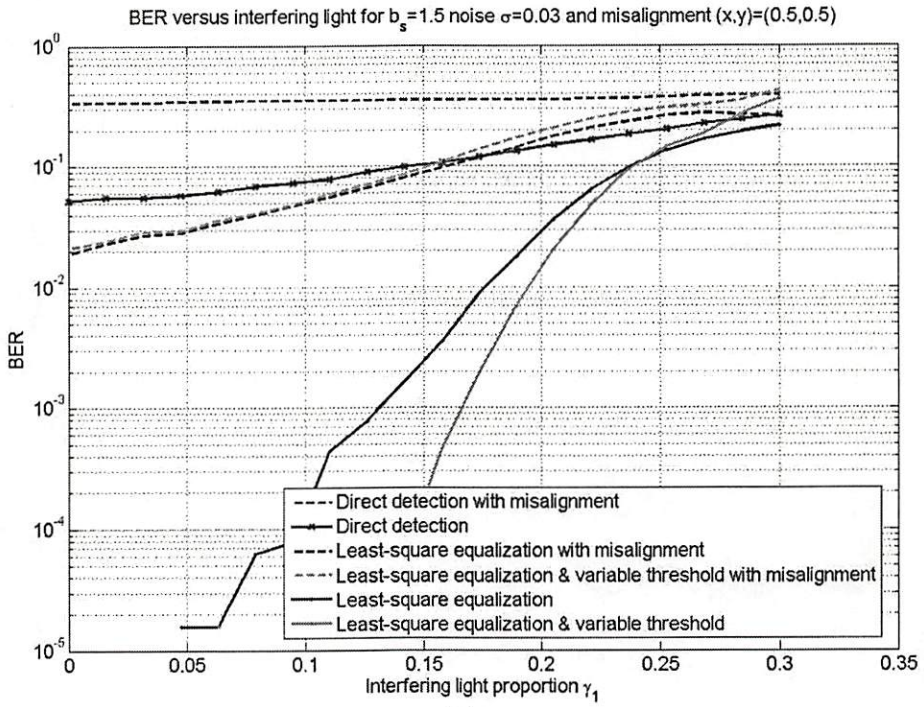



(a)

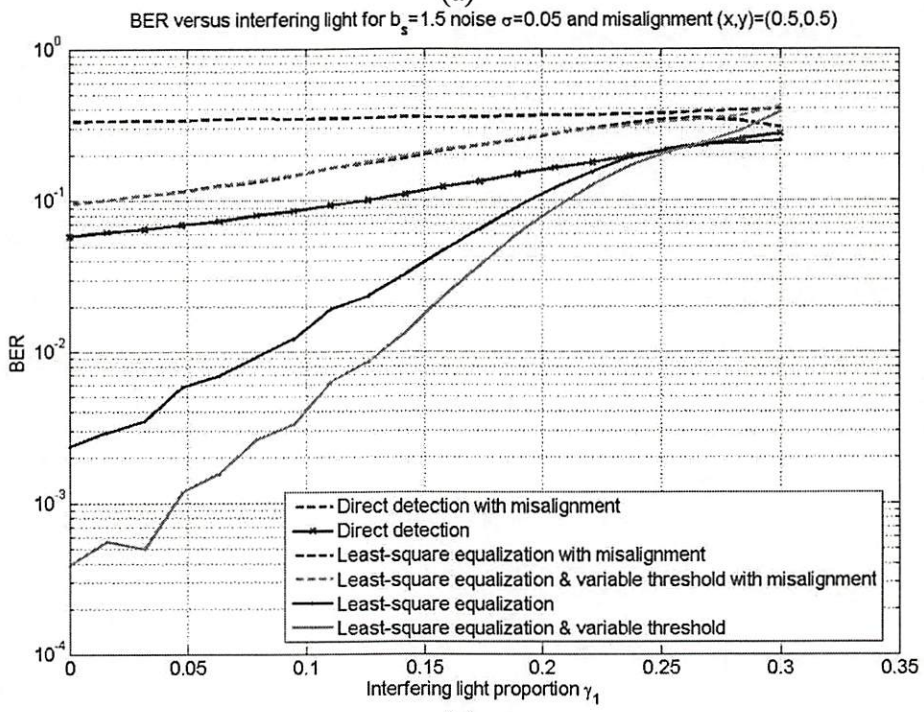



(b)

Figure 9.6. Pixel misregistration simulation results on rectangular coordinates with $b_s$=1.5, shift (x,y)=(0.4,0.4) and noise (a) $\sigma$ =0.03 (b) $\sigma$ =0.05.




(a)




(b)

Figure 9.7. Pixel misregistration simulation results on rectangular coordinates with $b_s$=1.5, shift (x,y)=(0.5,0.5) and noise (a) $\sigma$=0.03 (b) $\sigma$=0.05.

# Chapter 10 Summary, Future Work and Conclusions

## 10.1 Conclusions and Discussions

We have presented an application of least square equalization and variable threshold detection to three-dimensional page-oriented data storage (3D PODS) systems in order to improve the bit-error rate (BER) in the presence of ISI and IPI. We also described additional details of detection and noise reduction techniques that improve the performance of future multi-layer optical data storage systems with very high storage density. In Chapter 3, a detailed 3D PODS system mathematical model is introduced. We investigated inter-symbol interference (ISI) as well as inter-page interference (IPI) on multi-layer data storage systems. Based on an optical model, the effect of focusing error in the readout subsystem was examined. We also described the readout light intensity distribution for completeness. The novelty of this model is that it integrates the ISI and IPI effects of 3D PODS. In conventional PODS analyses, only the two-dimensional ISI is considered, and the ISI and IPI effects of 3D PODS with high data packing density are ignored. In the 3D PODS, the distance between data layers is shortened to increase the data packing density. Therefore, when the layer spacing is reduced, the effect of IPI cannot be neglected and must be taken into consideration with the effect of ISI.

With the 3D PODS model, we proposed a least square equalization detection algorithm for 3D PODS. Due to the inherent ISI and IPI of 3D PODS, our algorithm

can extend the accuracy of recorded data in the presence of these unwanted effects and the noise. Although we demonstrated least square equalization with three adjacent pages in an IPI model, it can be adapted easily to more general IPI effects. For different computational requirements, we described two modes of operation for least square equalization, i.e. the batch mode and the recursive mode. In batch mode operation, the data page is recovered when all the received signals containing that data page are available. On the other hand, some of the data page can be recovered when only part of these received signals are available in recursive mode operation. These two modes of operation give sufficient flexibility to our algorithm for different applications. It is also interesting to note that both of these operations generate the same detection results when given all the received signals.

In Chapter 5, we showed estimation and detection results based on our proposed algorithms. In the simulation, we used the physical parameters typical of an actual system. To have a fair comparison, we performed the simulations under different experimental settings. In the result figures, our detection algorithm achieved consistently better improvement than the conventional binary thresholding method. It is clear that the conventional thresholding method is not suitable for 3D PODS with both ISI and IPI effects.

Furthermore, it should be emphasized that while our least square equalization algorithm performs well in 3D PODS, it can also be used to recover the data with ISI effects only. In planar data storage systems, IPI effects are not present. Our proposed

algorithm can be employed in this situation with reduced complexity, i.e. reduced dimensionality in the matrix calculation.

To further enhance the data packing density, we introduced a multi-level data storage procedure in Chapter 6. It increases the number of signal levels encoded in the pixel, thus it can increase the amount of stored data without enlarging the area of the medium. Hence the data packing density is boosted by several times. However, multi-level system is more sensitive to additive noise at the detector plane. We evaluate two ways to achieve multi-level storage and compare their performance.

Based on the knowledge of the readout point-spread function, we proposed a variable threshold method in Chapter 7. The novelty of this detection method is that estimates the interference from surrounding neighbors and varies the threshold for each pixel. In this algorithm, the least square equalization is used to provide data estimation in computing the interference. The simulation results shows significant improvement over conventional fixed threshold method.

In Chapter 4 through Chapter 7, we assumed that the signal processing algorithms are implemented using floating-point DSPs with 32 bits of data (24 bits for the mantissa and 8 bits for the exponent). When the algorithm is implemented in hardware, it is important to know the effect of finite precision arithmetic. We analyzed our algorithms using the finite precision arithmetic in Chapter 8. We compared the performance with the use of fixed-point arithmetic and floating-point arithmetic. The fixed-point DSPs have better computing speed (higher MIPS rating) and cost less

than the floating-point DSPs. The tradeoff between the two techniques is an important design issue. This factor determines balance points between the system performance, computational complexity, and cost. From the simulation results, the fixed-point arithmetic uses only less than one sixth of the bits in 64-bit floating-point arithmetic with minor effects on the system performance.

In Chapter 9, we examined the effect of misalignment in the readout system and applied our algorithms to compensate it. Due to the pixel misregistration, the received signal is partially lost and with more unwanted interference. Our least square equalization and variable threshold detection methods successfully improve the performance with small amounts of misregistration.

Because we carefully examine ISI and IPI effects in our 3D PODS model, we have a basis for future research. With these mathematical models, one can try new algorithms and compare the performance. As the demand of data packing density keeps increasing, there will be increasing research on 3D PODS systems. Hence this thesis serves as pioneering work in this field.

## 10.2 Future Work

Because of their faster data transfer rate and higher data capacity, 3D PODS systems are a potential system for next generation data storage. Since 3D PODS systems are still in the early stages of development, there are still many possible areas of

improvement. The work described in this thesis should be useful for many future research possibilities. We list some interesting topics in this section.

### 10.2.1 Encoding Schemes for Multi-Level Data Storage Systems

We discussed two techniques for multi-level PODS data encoding. We plan to study the benefit of using different encoding schemes on multi-level systems. In one-dimensional communication systems, it is common and well developed to use multi-level encoding. Because of the two-dimension nature of 3D PODS, these encoding schemes must be modified before they can be applied.

### 10.2.2 Multi-Wavelength Readout Systems

In certain recording media, the readout data pattern is a function of the wavelength of the readout laser. For extremely high data packing densities, we can use different readout wavelengths in the same storage medium. Hence, we treat this multi-wavelength as another dimension to increase the data capacity. This topic is another solution to increase the capacity of future data storage systems.

# References


[1] M. J. Buckingham, *Noise in Electronic Devices and Systems* (Halsted Press, New York, 1983).

[2] G. W. Burr, H. Coufal, R. K. Grygier, J. A. Hoffnagle, and C. M. Jefferson, "Noise reduction of page-oriented data storage by inverse filtering during recording," Opt. Lett. **23**, 289-291 (1998).

[3] G. W. Burr, J. Ashley, H. Coufal, R. K. Grygier, J. A. Hoffnagle, C. M. Jefferson, and B. Marcus, "Modulation coding for pixel-matched holographic data storage," Opt. Lett. **22,** 639-641 (1997).

[4] Call/Recall Inc, San Diego, CA., "Fast Readout Optical Storage Technology Consortium," 2002, The Defense Advanced Research Projects Agency (DARPA).

[5] X. Chen, K. M. Chugg, and M. A. Neifeld, "Near-optimal parallel distributed data detection for page-oriented optical memories," IEEE J. Sel. Top. Quantum Electron. **4,** 866-879 (1998).

[6] K. M. Chugg, X. Chen, and M. A. Neifeld, "Two-dimensional equalization in coherent and incoherent page-oriented optical memory," J. Opt. Soc. Am. A **16,** 549-562 (1999).

[7] D. J. Costello, S. Lin, *Error Control Coding: Fundamentals and Applications*, 2nd Ed. (Prentice-Hall, New Jersey, August 2003).

[8] C. De Caro, A. Renn, and U. P. Wild, "Hole burning, Stark effect, and data storage:2:holographic recording and detection of spectral holes," Appl. Opt. **30,** 2890-2898 (1991).

[9] J. Dovic et al, "Multi-track DVD-ROM," in *Proceedings of Optical Data Storage Topical Meeting 2001*, Vol. 4342, (Optical Society of America, April 2001), pp. 112-114.

[10] A. S. Dvornikov, S. Esener, and P. M. Rentzepis, "Three-dimensional optical storage memory by means of two-photon interaction," in *Optical Computing Hardware*, J. Johns and S. H. Lee, eds. (Academic Press, Massachusetts, 1994), pp. 287-325.

[11] L. Fagoonee, W. M. J. Coene, A. Moinian, and B. Honary, "Nonlinear signal-processing model for signal generation in multilevel two-dimensional optical storage," Opt. Lett. **29**, 385-387, (2004).

[12] J. W. Goodman, *Introduction to Fourier Optics, 2nd Ed.* (McGraw-Hill, New York:, 1996).

[13] J. F. Heanue, M. C. Bashaw, and L. Hesselink, "Channel codes for digital holographic data storage," J. Opt. Soc. Am. A **12,** 2432-2439 (1995).

[14] J. F. Heanue, M. C. Bashaw, and L. Hesselink, "Volume holographic storage and retrieval of digital data," Science **265,** 749-752 (1994).

[15] T. Ho, N. Intharasombat, and A. A. Sawchuk, "Least Square Equalization and Variable Thresholding for 3D Page-Oriented Optical Data Storage Systems," in *SPIE Conference on Optical Data Storage* (Optical Society of America, April 2004).

[16] J. H. Hong, I. McMichael, T. Y. Chang, W. Christian, and E. G. Paek, "Volume holographic memory systems: techniques and architectures," Opt. Eng. **34,** 2193-2203 (1995).

[17] S. Hunter, F. Kiamilev, S. Esener, D. A. Parthenopoulos, and P. M. Rentzepis, "Potentials of two-photon based 3-D optical memories for high performance computing," Appl. Opt. **29,** 2058-2066 (1990).

[18] N. Intharasombat, T. Ho and A. A. Sawchuk, "Overcoming Inter-Page Interference (IPI) in 3D Optical Data Storage Systems," *European Optical Society Conference on Optics in Computing* (Optics in Computing, April 2004).

[19] M. Keskinoz and B. V. K. Vijaya Kumar, "Application of linear minimum mean-squared-error equalization for volume holographic data storage," Appl. Opt. **38,** 4387-4393 (1999).

[20] M. Keskinoz, B. V. K. V. Kumar, "Discrete magnitude-squared channel modeling, equalization, and detection for volume holographic storage channels," Appl. Opt. **43**, 1368-1378 (2004).

[21] B. M. King and M. A. Neifeld, "Sparse modulation coding for increased capacity in volume holographic storage," Appl. Opt. **39,** 6681-6688 (2000).

[22] B. Kohler, S. Bernet, A. Renn, and U. P. Wild, "Storage of 2000 holograms in a photochemical hole-burning system," Opt. Lett. **18,** 2144-2146 (1993).

[23] H-Y. S. Li and D. Psaltis, "Three dimensional holographic disk," Appl. Opt. **33,** 3764-3774 (1994).

[24] S. Lin, D.J. Costello Jr., *Error Control Coding* (Prentice-Hall, Englewood Cliffs, NJ, 1983).

[25] E. S. Maniloff, S. B. Altner, S. Bernet, F. R. Graf, A. Renn, and U. P. Wild, "Recording of 6000 holograms by use of spectral hole burning," Appl. Opt. **34,** 4140-4148 (1995).

[26] F. B. McCormick, "2-photon optical storage technology," in *Optical Processing and Computing*, Y. Fainman, ed., SPIE International Technical Working Group Newsletter, Special Issue on Optical Data Storage, Vol. 9, No. 1, pp. 1 (April 1998).

[27] J. M. Mendel, *Lessons in Estimation Theory for Signal Processing, Communications, and Control* (Prentice-Hall, Englewood Cliffs, NJ., 1995).

[28] D. E. Pansatiankul and A. A. Sawchuk, "Multi-dimensional modulation coding for page-oriented data storage systems", doctoral dissertation (Dept. Electrical Engineering, Univ. of Southern California, May 2002).

[29] D. A. Parthenopoulos and P. M. Rentzepis, "Three-dimensional optical storage memory," Science **245,** 843-845 (1989).

[30] W. K. Pratt, *Digital Image Processing, 3rd Ed.* (John Wiley and Sons, New York, 2001).

[31] D. Psaltis and G. W. Burr, "Holographic data storage," Computer **31**, Vol. 2, 52-60 (1998).

[32] D. Psaltis and F. Mok, "Holographic memories," Sci. Am. **273**, Vol. 5, 70-76 (1995).

[33] D. Psaltis, "Parallel optical memories," Byte **17**, Vol. 9, 179-182 (1992).

[34] A. Pu and D. Psaltis, "High-density recording in photopolymer-based holographic three-dimensional disks," Appl. Opt. **35**, 2389-2398 (1996).

[35] A. Renn and U. P. Wild, "Spectral hole burning and hologram storage," Appl. Opt. **26**, 4040-4042 (1987).

[36] B. E A. Saleh and M. C. Teich, *Fundamentals of Photonics* (Wiley, New York, 1991).

[37] A. A. Sawchuk, D. E. Pansatiankul, "Architectures and COTS Solutions Study for FROST Data Retrieval," in *Fast Readout Optical Storage Technology Consortium Quarterly Meeting* (December 12, 2000).

[38] F. M. Schellenberg, W. Lenth, and G. C. Bjorkland, "Technological aspects of frequency domain data storage using persistent spectral hole burning," Appl. Opt. **25**, 3207-3216 (1986).

[39] W. J. Smith, *Modern Optical Engineering, The Design of Optical Systems, 3rd Ed.* (McGraw-Hill, New York, 2000).

[40] D. A. Timugin, J. E. Walkup, and T. F. Krile, "Accuracy in analog optical processors: statistical analysis", J. Opt. Soc. Am. **11**, No. 2, pp 560-571 (February 1994).

[41] J. W. Toigo, "Avoiding a Data Crunch," Scientific American, (May 2000).

[42] V. Vadde and B. V. K. Vijaya Kumar, "Channel modeling and estimation for intrepage equalization in pixel-matched volume holographic data storage," Appl. Opt. **38**, 4374-4386 (1999).

[43] A. Vardy, M. Blaum, P. H. Siegel, and G. T. Sincerbox, "Conservative arrays: multidimensional modulation codes for holographic recording," IEEE Trans. Inf. Theory **42**, 227-230 (1996).

[44] M. M. Wang, S. C. Esener, F. B. McCormick, I. Cokgor, A. S. Dvornikov, and P. M. Rentzepis, "Experimental characterization of a two-photon memory," Opt. Lett. **22**, 558-560 (1997).

[45] J. Watkinson, *The Art of Data Recording* (Oxford, Butterworth-Heinemann Ltd., 1994).