國 立 交 通 大 學

電信工程學系

碩 士 論 文

運用支撐向量機技術之FHMA/MFSK訊號偵測

Detection of FHMA/MFSK Signals Based on SVM Techniques

研究生：劉人仰

指導教授：蘇育德 教授

中 華 民 國 九十四 年 七 月

運用支撐向量機技術之 FHMA/MFSK 訊號偵測

研究生：劉人仰　　指導教授：蘇育德 博士

國立交通大學電信工程研究所

中文摘要

本論文提供一個初步用來設計智慧型通訊處理器(intelligent communication processor)的整合架構。智慧型通訊處理器的一個重要特性是其可以利用離線式(off-line)的訓練時期所獲得的知識(knowledge)來做及時性的訊號偵測。雖然離線式的訓練可能需要大量的運算，但所獲得的資訊卻是很簡潔的，因此智慧型通訊處理器只需要小量的運算就可以做訊號偵測。利用這樣的概念，我們從機械學習的觀點來探討在多重存取環境中跳頻(frequency hopping)訊號偵測的問題。

相對於直接序列展頻(DS-CDMA)技術，跳頻技術是另一種吸引人的多重存取技術。

除了通道統計特性，跳頻多重存取系統的效能會由兩項主要考量所決定：波形(waveform)的設計和接收機的架構。在給定跳頻波形，我們仍然很難設計最大相似性(ML)接收機，其主要原因是我們對於通道統計特性的瞭解是不完整的，即使是完整的，通常相對應的條件機率分佈(conditional pdf)是沒有簡潔的表示式。

藉由把接收到的訊號視爲一個時間—頻率的圖形，我們可以把多使用者偵測的問題轉換成圖形辨識的問題，然後運用支撐向量機的技術來解決所對應產生的圖形辨識問題。利用適當的核心函數(kernel function)，支撐向量機技術把收到的訊號轉換到高維度的特徵空間。利用了序列最小優化(Sequential Minimization Optimization, SMO)和有向無迴圈圖(Direct Acyclic Graph)演算法來找尋在特徵空間中的最佳分離平面，我們提出了基於支撐向量機技術的接收機。模擬的結果顯示我們的設計具有強韌性和令人滿意的效能。

Detection of FHMA/MFSK Signals Based on SVM Techniques

Student : Jen-Yang Liu Advisor : Yu T. Su

Department of Communications Engineering

National Chiao Tung University

Abstract

This thesis documents an initial effort in establishing an unified framework for designing an intelligent communication processor (ICP). An important feature of a prototype ICP is its capability of applying the knowledge learned during an off-line training period to real-time signal detection. Although the off-line training might require very intensive computing power, the extracted information does has a concise representation, which then enables the corresponding ICP to detect a signal using only simple and low-power operations. As an application of such a concept, we revisit the problem of detecting frequency-hopped (FH) signals in a multiple access (MA) environment from a machine learning perspective.

Frequency-hopping is an attractive alternative multiple access technique for direct sequence based code division multiple access (CDMA) schemes. Other than the communication channel statistic, the capacity of an FHMA system is determined by two major related design concerns: waveform design and receiver structure. Given the FH waveform, one still has difficulty in designing an FHMA ML receiver due to the facts that our knowledge about the channel statistics is often incomplete and even if it is complete the associated conditional probability density function (pdf) does not render a closed-form expression.

Regarding the FHMA/MFSK waveform as a time-frequency pattern, we convert the mutiuser detection problem into a pattern classification problem and then resolve to the Support Vector Machine (SVM) approach for solving the resulting multiple-class classification problem. By using an appropriate kernel function, the SVM essentially transforms the received signal space into a higher dimension feature space. We propose a SVM-based FHMA/MFSK receiver by applying the Sequential Minimization Optimization (SMO) and Directed Acyclic Graph (DAG) algorithms to find the optimal separating hyperplanes in the feature space. Simulation results indicate that our design does yield robust and satisfactory performance.

致　　謝

首先感謝蘇育德老師兩年來的指導，不僅是學術上地帶領，對於生活上的學習，也受益非淺。也感謝家人的支持，由於你們默默的努力才會有今日的我。

實驗室的各位學長、同學、學弟、學妹，謝謝你們對於我平日的任性百般忍受，尤其是博班的各位學長，感謝你們平日的教誨，讓我對於其他的事物有更多的了解。

最後感謝我身邊的朋友們，因爲你們平日無形的幫助，讓我在學業和日常生活上能平順地渡過困境。要特別感謝的是繃帶小孩和冰牆。由於繃帶小孩的存在使我想起了許許多多要做的事情，而我也期望在有生之年能夠有缺陷地把它們做完。而冰牆也讓我體會到了人生矛盾的地方，或許我並沒有完全的體會到，但我相信那是我啓蒙的開端。

2005/8/28 劉人仰

Contents

List of Figures

Chapter 1

Introduction

Data detection has been casted as a special instant of the hypothesis testing problems. Given an observation vector $\mathbf{R}$, an optimal binary communication receiver makes a decision based on

$$\Lambda(\mathbf{R}) \stackrel{def}{=} \Gamma(\mathbf{R}) + \eta \stackrel{def}{=} \log \frac{p(\mathbf{R}|H_1)}{p(\mathbf{R}|H_0)} + \eta \tag{1.1}$$

where $\Gamma(\mathbf{R})$ is called the log-likelihood ratio, η is a constant and $p(\mathbf{R}|H_i), i = 0, 1$ is the conditional probability density function (pdf) of $\mathbf{R}$ given that the hypothesis H_i is true. For the channel model

$$H_i : \mathbf{R} = \mathbf{m}_i + \mathbf{W} \tag{1.2}$$

where $\mathbf{m}_i, i = 0, 1$, are constant vectors and $\mathbf{W}$ is a zero-mean white Gaussian random vectors, the decision variable $\Lambda(\mathbf{R})$ is of the form

$$\Lambda(\mathbf{R}) = a\left(\langle \mathbf{R}, \mathbf{m}_1 \rangle - \langle \mathbf{R}, \mathbf{m}_0 \rangle\right) + b \tag{1.3}$$

where a, b are constants. In case $\mathbf{m}_1 = -\mathbf{m}_0 = \mathbf{m}$, we have

$$\Lambda(\mathbf{R}) = 2a\langle \mathbf{R}, \mathbf{m} \rangle + b \tag{1.4}$$

The inner products $\langle \mathbf{R}, \mathbf{m}_i \rangle$ represent a *similarity measure* between the received vector $\mathbf{R}$ and the candidate signal $\mathbf{m}_i$. A binary decision based on (1.3) or (1.4) is also known as a binary classifier. In this sense a receiver of a binary communication system is a binary classifier and that of an M-ary communication system can be regarded as a multiple (M) class classifier.

In many real-world applications, however, the channel model (1.2) is invalid and the conditional pdfs $p(\mathbf{R}|H_i)$ do not have a closed-form expression. Moreover, in the space where the received vector $\mathbf{R}$ lies, there might not be a well-defined inner product. Fortunately, recent advances in machine learning have produced many elegant solutions to resolve the above difficulties for binary and M-ary classifiers. There is a major distinction between machine learning and signal detection in a digital communication receiver: the latter has very little latency tolerance and real-time processing is an absolute necessity while the former usually requires a reasonable learning period. The purpose of this thesis is to propose a plausible solution to this dilemma so that a receiver can take the advantage of the information extracted from sophisticated statistical learning while requires only simple operations for real-time detection.

The underlying philosophical guideline of our investigation is: we believe that the computing power of digital signal processors has steadily improved for the past two decades to the point that the incorporation of computational intelligence in a wireless communication device has become both feasible and desirable. We envision a communication receiver to possess, besides some stored intelligence, advanced on-line learning capability. This thesis represents a preliminary study towards establishing a theoretical framework for design such an intelligent communication processor.

To demonstrate the powerfulness of our approach, this thesis considers the detection of frequency-hopped multiple access (FHMA) signals in both AWGN and frequency nonselective channels as the sole application example. The same approach can be applied to the detection of a random signal in uncertain environments that do not yield analytical characterization on the associated probability distributions. Multiuser detection of wireless CDMA signals and demodulation of QAM signals in nonlinear satellite channels are but two other candidate applications.

FHMA techniques have been proposed and studied for more than two decades. Cooper and Nettleton [4] first proposed a frequency-hopped, phase-shift-keying, spread-

spectrum system for mobile communication applications. At about the same time Viterbi [22] suggested the use of multilevel FSK (MFSK) for low-rate multiple access mobile satellite systems. In 1979, Goodman *et al.* [9] studied the capability of a fast frequency-hopped MFSK with a hard-limited diversity combining receiver in additive, white Gaussian noise (AWGN) and Rayleigh fading channels.

Two major related design issues concerning the design of an FHMA system are the hopping pattern and the receiver structure. Solomon [15] and Einarsson [6] proposed FH pattern assignment schemes that render minimum multiple assess interference (MAI) in a chip-synchronous environment. Most of the earlier investigations on the noncoherent FHMA receiver design and performance analysis assume that: 1) a single-stage detector is employed by the receiver. 2) a chip(hop)-synchronous mode is maintained, i.e., all users switch their carrier frequencies simultaneously. The second assumption is a more practical assumption for the forward link than for the reverse link. The assumption also exempts the receiver from dealing with cochannel interference (CCI) and interchannel interference (ICI) if an appropriate channel separation is used by the system. Timor [20] proposed a two-stage chip-synchronous demodulator that is capable of eliminating most of the CCI if the thermal noised effect can be neglected by taking advantage of the inherent algebraic structure. He later [21] extended the two-stage demodulator to a multistage decoding algorithm to further improve the system performance. Recently, Su *et al.* [17] presented a nearly maximum likelihood (ML) diversity combiner for detecting asynchronous FHMA/MFSK signals in Rayleigh fading channels and analyze the performance of a close approximation of the ML receiver. They also discussed the effectiveness of a two-stage multiuser detector.

Since the relative timing offsets, the signal strengths, the hopping patterns for other system (undesired) users and the number of undesired users are usually unknown to the desired user, it is very difficult if not impossible to obtain a closed-form expression for the posteriori (conditional) probability density function (pdf) associated with a noncoherent

MFSK detector output. The derivation of a maximum likelihood detector is thus out of question under the said scenario. We try to solve the FHMA detection problem from a different perspective.

Regarding the FHMA/MFSK waveform observed at the output of a noncoherent matched-filter bank as a two-dimensional pattern, we then have a multiple-pattern classification problem. Recent advances in machine learning theory has offered a variety of approaches that give very impressive performance. In this thesis, we apply a nonlinear technique called support vector machine (SVM) to provide new solution to the problem of concern. SVM has been used in multiuser detection for CDMA systems [2, 8]. Gong [8] has shown that the SVM scheme has the advantage over many traditional adaptive learning approaches (e.g. the Least Mean Square (LMS) algorithm or the error back-propagation algorithm) in terms of performance, complexity and convergence rate. A major advantage of using SVM is that a nonlinear decision function can be constructed easily by using kernel functions, which is often less complex than nonlinear adaptive and neural network methods.

The rest of thesis is organized as follows. In Chapter 2, we give a brief description of a typical FHMA/MFSK system and present the corresponding signal, channel and receiver model. Chapter 3 contains an overview on the theory of Support Vector Machines(SVM) classifiers. In Chapter 4, we then apply the SVM method to the design of the FHMA/MFSK signal detector and provide some numerical examples of the behavior of our SVM-based receiver. Finally, in Chapter 5, we draw some conclusion remarks and suggest some further works.

Chapter 2

Review of FHMA/MFSK Systems

In this chapter, we examine of the basic building blocks of an FHMA/MFSK system and briefly review the performance analysis of two existing representative receivers. In particular, we consider the receiver proposed by Goodman *et al.* [9] and that by [17]. For the former case, we assume that: 1) a single-stage detector is employed by the receiver. 2) a chip(hop)-synchronous mode is maintained, i.e., all users switch their carrier frequencies simultaneously. Later, we discuss the signal model, which removes the assumption 2). This chapter is organized as followed. In Section 2.1, we describe the system architecture proposed by Goodman *et al.* and in Section 2.2, we discuss the corresponding error rate performance. We then consider a modified receiver model [17] that takes into account the effects of ICI and CCI in Section 2.3.

2.1 Basic system building blocks

A block diagram for the transmitter of FHMA/MFSK system is given in Figs. 2.1-2.2. The data source produces a data bit stream of rate $R_b = \frac{K}{T}$ bit/s which is then converted into a K-bit word sequence $\{X_m\}$. The subscript m denotes mth link (user) in a multiuser system. Each user is assigned an unique signature sequence $\{a_{m1}, a_{m2}, \cdots, a_{mL}\}$ of rate $\tau = \frac{T}{L}$, where $a_{mi} \in \{0, 1, \cdots, M-1\}$ are $k-$bit symbols and each information word of T seconds duration is added modulo-$2^K = M$ by the unique $L-$symbol signature

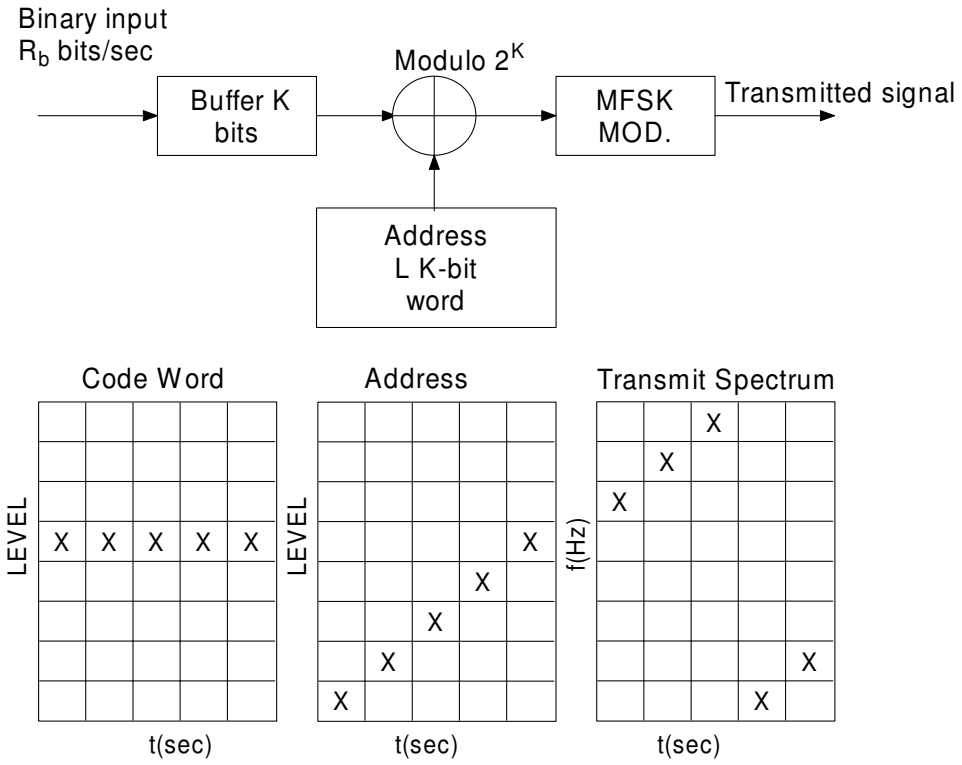


Figure 2.1: Transmitter block diagram and signal matrices. The matrices show sequences of logic levels (code word, address) or frequencies (transmitted spectrum) within the transmitter.

sequence of the same length. The resulting spread symbol sequence $\{Y_{ml}\}$ is given by

$$Y_{ml} = X_m \oplus a_{ml} \tag{2.1}$$

which is then used to select transmitter carrier frequencies.

At the receiving end of the system, demodulation and modulo-M *subtraction* are performed, yielding

$$Z_{ml} = Y_{ml} \ominus a_{ml} = X_m. \tag{2.2}$$

If there are other users sharing the same frequency band, the received signals could be scattered over different frequency bands at the (desired) mth user's receiver time-frequency filter outputs. The interference caused by the nth user is

$$Z'_{ml} = X_n \oplus a_{nl} \ominus a_{ml} \tag{2.3}$$

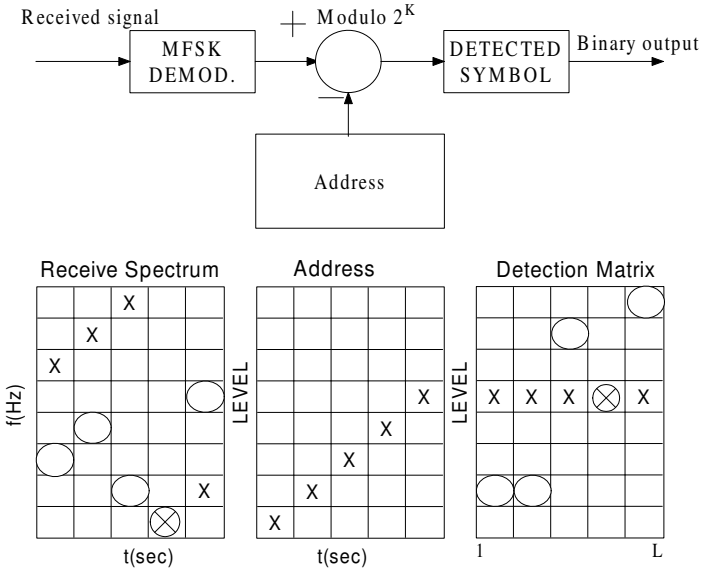


Figure 2.2: Receiver block diagram and signal matrices. The matrices show reception of the frequencies in Fig.2.1 (X) and reception of a set of tones (O) transmitted by one other user.

If the a_{nl} and a_{ml} are different, $Z_{ml} \neq Z'_{ml}$, i.e., there will be no interference in the desired band, see Fig. 2.2 in which crosses denote tones of the mth user while circles denote interfering tones. Using a proper family of signature sequences there will be minimum interfering tones coincided with the desired signal allowing multiple users to access the same frequency band. A block diagram for a complete FHMA/MFSK system is depicted in 2.3. We now address the issue the detecting FHMA/MFSK signals in a multiuser environment. We start with a generic nonlinear diversity-combining receiver as that shown in Fig. 2.4. The received waveform is first dehopped and then detected by a bank of energy detectors before passing through a memoryless nonlinearity $g(\cdot)$. For the one proposed by Goodman *et al.* [9], $g(\cdot)$ is simply a hard-limiter. Two kinds of error might occur, false alarm and miss. A false alarm occurs when the detected energy is greater than the hard-limiter threshold so that the detector produces an output indicating the presence of a signal tone while there is in fact none. A miss refers to the opposite case when an energy detector produces a "no signal" indicator while signal is actually

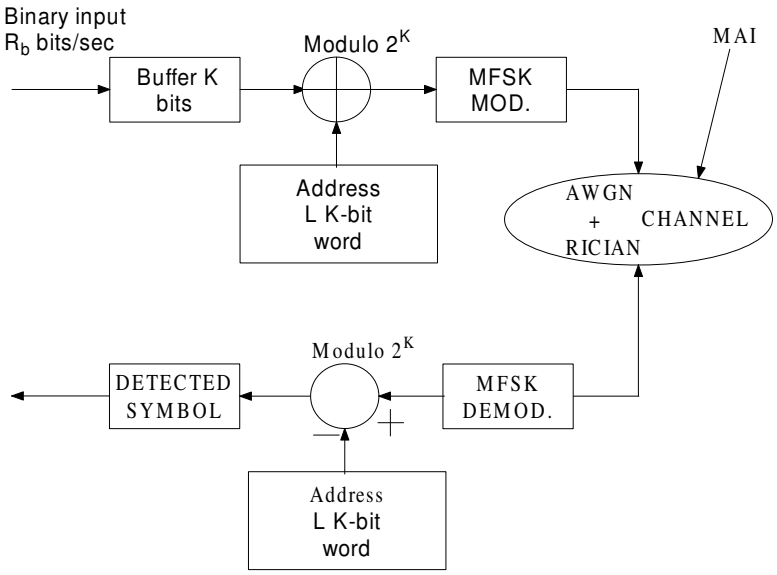


Figure 2.3: A block diagram for the FHMA/MFSK system under investigation.

present. We illustrate the effects of false alarms and misses in Fig. 2.5. The sole square indicates a missing tone in the transmitted code word and the filled circle indicates a false alarm. It is obvious that there is no complete row of "signal present" indicators in the detection matrix. Hence, it is not a good strategy to insist on choosing a complete row as a decision. To allow for this possibility, Goodman *et al.* use the majority logic decision rule: Choose the code word associated with the row containing the greatest number of "signal present" entries. Under this decision rule, an error will occur when insertions (tones due to other users and false alarms) form a row with more or the same number of entries than the row corresponding to the transmitted code word.

2.2 Error rate analysis

This section mainly follows the derivations of [9]. Table 2.1 contains a set of formulae which give a tight upper bound on the average bit error rate as a function of the following quantities:

(1) K bits per code word.

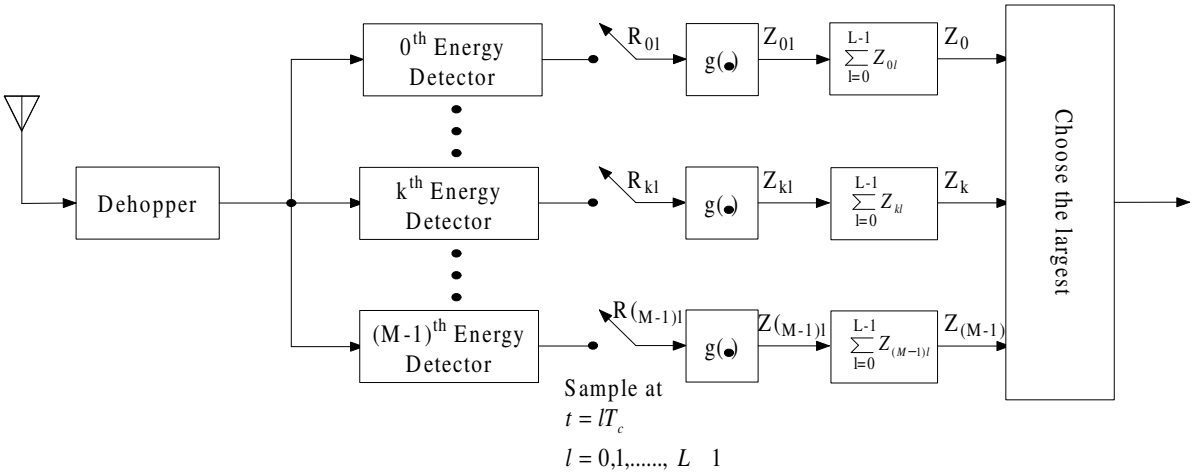


Figure 2.4: A nonlinear diversity-combining receiver for noncoherent FHMA/MFSK signals.

(2) L hopped times per code word.

(3) U simultaneous users.

(4) p_D miss probability.

(5) p_F false alarm probability.

A key assumption in the derivation of these formulas is that the signature sequence of each user is chosen at random. That is, each a_{ml} is uniformly distributed between 0 and $2^K - 1$ and is independent of all other address word. For the detail of derivation, please refer to [9].

2.3 A nonlinear diversity-combining detector

In this section, we introduce another receiver structure [17] that works for both the chip(hop)-synchronous and chip asynchronous conditions. Without the chip-synchronous assumption different FHMA codewords are no longer orthogonal and interfering system users cause only not co-channel interference (CCI) but also inter-channel interference

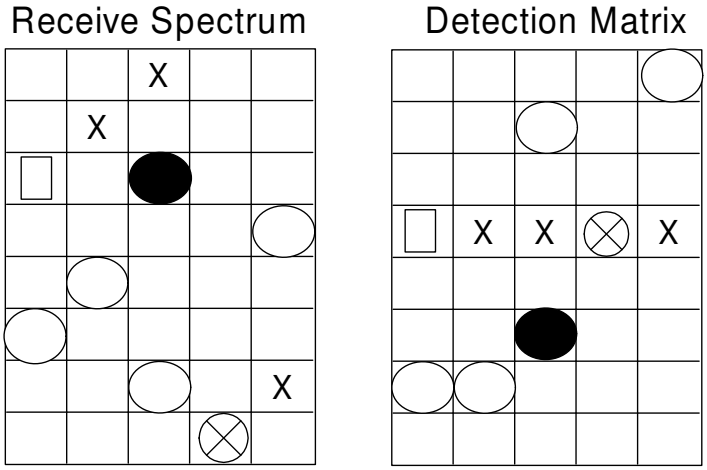


Figure 2.5: The effects of a false alarm and a miss on signal in the receiver of 2.2.

(ICI). In other words, even if an interfering user does not use the same tone during a given hop interval, it still contribute to the desired user's energy detector output.

Let the L-tuple, $\mathbf{a} = (a_0, a_1, \ldots, a_{L-1})$, be a signature sequence, where, as before, $a_i \in G = \{0, 1, \ldots, M-1\}$, represents a candidate FSK tone of duration T_c seconds. The actual transmitted frequencies within a symbol interval is

$$\mathbf{f}_m = \mathbf{f}_0 + (b_{m0}, b_{m1}, \ldots, b_{m(L-1)})\triangle f = \mathbf{f}_0 + (\mathbf{m} \oplus \mathbf{a})\triangle f,$$

where $\mathbf{m} = (m, m, \ldots, m)$ is a constant L-tuple representing a K bits message, $\oplus$ denotes modulo-M addition, $\mathbf{f}_0$ is the carry frequency, $\triangle f$ is the tone space and $b_{ml} \in G$. Define a set of basis waveforms

$$x_{nl}(t) = \sqrt{S} P_{T_c}(t - lT_c) e^{i[2\pi(f_0 + b_{ml}\triangle f)t]} \tag{2.4}$$

where S is the signal power, $E_c = ST_c$ is the average signal energy per chip, $T_c = T_s/L$ is the chip duration and

$$P_{T_c}(t) = \begin{cases} 1, & 0 \le t < T_c \\ 0, & \text{otherwise.} \end{cases} \tag{2.5}$$

The received waveform, corresponding to the desired user, consists of three components

$$\widehat{r}(t) = \widehat{c}_m(t) + \sum_{j=1}^{J} \widehat{I}_j(t) + n(t) \tag{2.6}$$

Probability of insertion due to interference:

$$p = (1 - (1 - 2^{-K})^{U-1})(1 - p_D)$$

Probability of insertion due to interference or false alarm:

$$p_I = p + p_F - pp_F$$

Probability of m entries in a spurious row:

$$P_S(m) = \begin{pmatrix} L \\ m \end{pmatrix} p_I^m (1 - p_I)^{L-m}.$$

Probability that no unwanted row has as many as n entries:

$$P(n, 0) = \left(\sum_{m=0}^{n-1} P_S(m) \right)^{2^K - 1} ; \; n > 0$$

Probability that n is the maximum number of entries in an unwanted row and only one unwanted row has n entries:

$$P(n, 1) = (2^K - 1) P_S(n) \left(\sum_{m=0}^{n-1} P_S(m) \right)^{2^K - 2} ; \; n = 1, 2, \ldots, L$$

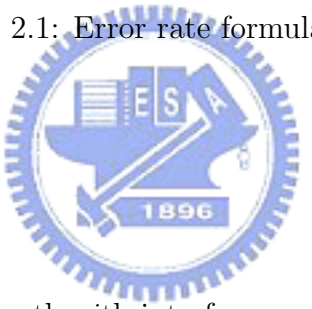
Probability of i entries in the correct row:

$$P_C(i) = \begin{pmatrix} L \\ i \end{pmatrix} (1 - p_D)^i p_D^{L-i}$$

Upper bound on bit-error rate:

$$P_B \approx \frac{2^{K-1}}{2^K - 1} \left(1 - \sum_{i=1}^{L} P_C(i)[P(i, 0) + \tfrac{1}{2} P(i, 1)] \right)$$

Table 2.1: Error rate formulas.



where

$\widehat{c}_m(t)$ = desired signal;

$\widehat{I}_j(t)$ = interference caused by the jth interferer;

J = number of interferers (hence there are $K = J + 1$ users in the system);

$n(t)$ = additive white Gaussian noise (AWGN) whose one-sided power spectral density is N_0 $\mathrm{W/H_Z}$.

We assume that different MFSK tones at the same chip experience independent and identically (i.i.d) frequency nonselective Rician fading and statistics of a user signal at different chips are i.i.d. as well. Ignoring the propagation delay, the desired signal can be written as

$$\widehat{c}_m(t) = \sum_{l=1}^{L-1} \alpha_{ml} \sqrt{S} P_{T_c}(t - lT_c) e^{i[2\pi(f_0 + b_{ml} \triangle f)t + \theta_{ml}]} \tag{2.7}$$

where $\{\theta_{ml}\}$ are random variables $(r.v.s)$ uniformly distributed within $[0, 2\pi)$ and $\{\alpha_{ml}\}$ are i.i.d. Rician $(r.v.s)$. Note that $\{\theta_{ml}\}$ represents the noncoherent receiver's uncertainty about the phase of the received waveform. Let Γ be the ratio of the power a^2 in the specular component to the power, $2\sigma_f^2$, in the diffused component. Using the normalization, $E(\alpha_{ml}^2) = a^2 + 2\sigma_f^2 = 1$, we obtain $a^2 = \Gamma/(1+\Gamma)$ and $2\sigma_f^2 = 1/(1+\Gamma)$, respectively. The jth interfering signal, being transmitted through a similar but independent Rician fading channel, has the form

$$\widehat{I}_j(t) = \sum_{k=0}^{M-1} \sum_{l=0}^{L-1} \alpha_{jl} c_{jkl} \sqrt{S} P_{T_c}[t - (l - \varepsilon_j)T_c] e^{i[2\pi(f_0 + k\triangle f)t + \theta_{jl}]} \tag{2.8}$$

where α_{jl} and θ_{jl} are independent random variables having the same distribution as α_{ml} and θ_{ml} of the desired signal. However, for each (j, l) pair only one c_{jkl} is nonzero, i.e., $c_{jkl} = \delta(k - b_{jl})$, where $\delta(x) = \begin{cases} 1 & \text{if } x = 0 \\ 0 & \text{otherwise.} \end{cases}$ and b_{jl} is the frequency tone used by the jth interferer during the lth chip interval. ε_j is the corresponding normalized timing misalignment with respect to the desired signal and is assumed to be uniformly distributed with in $[0, 1]$. With the received waveform given by (2.6)-(2.8), the dehopped signal becomes

$$r(t) = c_m(t) + \sum_{j=1}^{J} I_j(t) + n(t) \tag{2.9}$$

where

$$c_m(t) = \sum_{l=1}^{L-1} \alpha_{ml} \sqrt{S} P_{T_c}(t - lT_c) e^{i[2\pi(f_0 + m\triangle f)t + \phi_{ml}]} \tag{2.10}$$

$$\begin{aligned} I_j(t) = \sum_{k=0}^{M-1} \sum_{l=0}^{L-1} \{ & \alpha_{jl} c_{jkl} \sqrt{S} P_{T_c}[t - (l - \varepsilon_j)T_c] e^{i[2\pi(f_0 + g_{kl}\triangle f)t + \phi_{jl}]} \\ & + \alpha_{j(l+1)} c_{jk(l+1)} \sqrt{S} P_{T_c}[t - (l + 1 - \varepsilon_j)T_c] e^{i[2\pi(f_0 + g_{kl}\triangle f)t + \phi'_{jl}]} \} \end{aligned} \tag{2.11}$$

and $g_{kl} = (k - a_l) \bmod M$. Since the phase of the local frequency synthesizer is not continuous (noncoherent) when it hops from one band to another, i.e., the "local phases" at different hops are independent, the dehopped random phases, $\phi_{ml}, \phi_{jl}, \phi'_{jl}$ are i.i.d. $r.v.s$ that are uniformly distributed within $[0, 2\pi]$. Notice that the equations (2.9)-(2.11)

imply that perfect power control has been assumed so that all the received signals have the same average strength. Now, we consider the complex random variable U_{nl}, which is given by

$$U_{nl} = \frac{1}{E_c}\int_{lT_c}^{(l+1)T_c} r(t)x_{nl}^*(t)dt \stackrel{def}{=} s_{nl} + \sum_{j=1}^{J} I_{jnl} + z_{nl} \tag{2.12}$$

where $r(t)$ is given by (2.9), $x_{nl}^*(t)$ is the complex conjugate of $x_{nl}(t)$ defined by (2.4), and s_{nl}, I_{jnl}, z_{nl} are the components contributed by the desired signal, the jth interferer and AWGN, respectively.

Denote the output of the nth energy detector at the jth diversity branch (hop) during a given symbol interval by R_{nl} (see Fig. 2.4). Then, it can be shown that $R_{nl} = \|U_{nl}\|^2$. In particular, Z_{nl} is a zero-mean complex Gaussian random variable whose real and imaginary parts have the same variance $\sigma_0^2 = (2E_c/N_0)^{-1}$. Substituting (2.4) and (2.9)-(2.11) into the upper part of (2.12), we immediately have

$$s_{nl} = \alpha_{ml}e^{i\phi_{ml}}sinc[(m-n)\eta]e^{i[2\pi(m-n)(l+\frac{1}{2})\eta]} \tag{2.13}$$

where $\mathrm{sinc}(x) \equiv \sin(\pi x)/(\pi x)$, $\eta = \triangle f/R_c$, and

$$\begin{aligned} I_{jnl} &= \sum_{k=0}^{M-1}\left\{\alpha_{jl}c_{jkl}e^{i\phi_{jl}}(1-\epsilon_j)\mathrm{sinc}[g_{kln}(1-\epsilon_j)\eta]e^{i[2\pi g_{kln}(l+\frac{1-\varepsilon_j}{2})\eta]}\right\} \\ &+ \sum_{k=0}^{M-1}\left\{\alpha_{j(l+1)}c_{jk(l+1)}e^{i\phi'_{j(l+1)}}\epsilon_j\mathrm{sinc}[g_{kln}\epsilon_j\eta]e^{i[2\pi g_{kln}(l+1-\frac{\varepsilon_j}{2})\eta]}\right\} \end{aligned} \tag{2.14}$$

with $g_{kln} = g_{kl} - n = [(k-a_l)\mathrm{mod}\ M] - n$. Although the channel separation g_{kln} can be as large as $M-1$, the corresponding ICI is a decreasing function of the parameter. Hence we will only consider $2B$ adjacent channels around the desired tone. The magnitude of the product $g_{kln}\triangle f$ represents the frequency separation between the dehopped jth interferer at the lth hop and the nth energy detector. The term $\mathrm{sinc}[g_{kln}(1-\varepsilon_j)\eta]$ represents the contribution to the output of the nth energy detector associated with the kth MFSK tone, which is generated by the jth interferer in the lth chip interval. As shown by (2.14), the orthogonality among the MFSK tones can no longer be maintained,

since the $\mathrm{sinc}[g_{kln}(1-\varepsilon_j)\eta]$ or $\mathrm{sinc}[g_{kln}\varepsilon_j\eta]$ is nonzero unless $\varepsilon_j = 0$ or 1 for all j (chip-synchronization), and, simultaneously $\eta = \triangle f/R_c$ is an integer ($R_c = 1/T_c$ is the chip rate).

2.4 A suboptimal realization

In this section, we discuss the suboptimal receiver proposed by [17]. According to the Bayes' decision rule, the noncoherent receiver is depicted in Fig.2.4 and the nonlinearity $g(x)$ is given by

$$g(x) = \frac{R_{1/2}[g_{\mathrm{opt}}(x) - g_{\mathrm{opt}}(0)]}{g_{\mathrm{opt}}(R_{1/2}) - g_{\mathrm{opt}}(0)} \tag{2.15}$$

where

$$g_{\mathrm{opt}}(x) \equiv \ln f_{R_{ml}}(x|m) - f_{R_{ml}}(x|k) \tag{2.16}$$

and $f_{R_{nl}}(x|m)$ is the probability density function under the mth chip is transmitted. It can be shown that

$$f_{R_{nl}}(x|m) = \frac{1}{2}\int_0^\infty \lambda J_0(\sqrt{x}\lambda)\Phi_{X_{nl}}(\lambda|m)d\lambda \tag{2.17}$$

where $X_{nl}^2 = R_{nl}$ and $\Phi_{X_{nl}}(\lambda|m)$ is the characteristic function of X_{nl}, which can be given by

$$\Phi_{X_{nl}}(\lambda|m) = e^{-\lambda^2\sigma_0^2/2}M(\lambda,m)\prod_{\rho=-B}^{B}[(1-\mu)^2 + 2\mu(1-\mu)H(\lambda,\rho) + \mu^2 G(\lambda,\rho)]^J \tag{2.18}$$

where δ_{nm} is the Kronecker delta function,

$$M(\lambda,m) = \begin{cases} \prod_{\rho=-B,\rho\neq 0}^{B}\left\{1-\mu+\mu e^{-\mathrm{sinc}^2(\rho\eta)\lambda^2\sigma_f^2/2}J_0[a\mathrm{sinc}(\rho\eta)\lambda]\right\}, & m\neq n \\ e^{-\lambda^2\sigma_f^2/2}J_(a\lambda), & m = n \end{cases}$$

$$H(\lambda,\rho) = \int_0^1 e^{-\epsilon^2\mathrm{sinc}^2(\rho\epsilon\eta)\lambda^2\sigma_f^2/2}J_0[a\epsilon\mathrm{sinc}(\rho\epsilon\eta)\lambda]d\epsilon \tag{2.19}$$

$$\begin{aligned} G(\lambda,\rho) &= \int_0^1 e^{-\{[1-\epsilon]^2\mathrm{sinc}^2[\rho(1-\epsilon)\eta]+\epsilon^2\mathrm{sinc}^2(\rho\epsilon\eta)\lambda^2\sigma_f^2/2\}}J_0(a\epsilon\mathrm{sinc}[\rho\epsilon\eta]\lambda) \\ &\quad J_0(a[1-\epsilon]\mathrm{sinc}[\rho(1-\epsilon)\eta]\lambda)d\epsilon \end{aligned} \tag{2.20}$$

Furthermore, $R_{1/2}$ is derived according to $g_{\rm opt}(R_{1/2}) = 0.5 g_{\rm opt}(R_{\rm sat})$ and the saturation point $R_{\rm sat}$ is defined as $R_{\rm sat} = \min\{x : g'_{\rm opt}(x) = 0\}$. Since $R_{\rm sat}$ is often difficult to locate, we may choose $R'_{\rm sat} = 100d$, where $d \equiv \frac{N_0}{E_c}$, as the reference point and define $R_{\rm sat}$ as the value such that $g_{\rm opt}(R_{1/2}) = 0.5 g_{\rm opt}(R'_{\rm sat})$.

However, we can further approximate the soft-limiter by a q-level quantizer, $\widetilde{g}(\cdot)$, with a uniform step size of $\nu = \Upsilon/(q-1) \equiv R'_{\rm sat}/(q-1)$, where Υ is the upper threshold of the quantizer. In this case, the probability mass function of a uniform quantizer's outputs $\{Z_{nl} = \widetilde{g}(R_{nl})\}$ can be expressed as

$$\begin{aligned} P_{Z_{nl}}(z) &= \sum_{k=0}^{q-1} P_r(Z_{nl} = k\nu)\delta(z - k\nu) \\ &\equiv \sum_{k=0}^{q-1} V_{nl}(k)\delta(z - k\nu) \end{aligned}$$

where the probabilities $V_{nl}(k) = P_r(Z_{nl} = k\nu) = P_r(k\nu \le R_{nl} < (k+1)\nu)$ are given by

$$V_{nl}(k) = \begin{cases} F_{R_{nl}}((k+1)\nu|m) - F_{R_{nl}}(k)\nu|m), & k = 0, 1, \ldots, q-2 \\ 1 - F_{R_{nl}}(\Upsilon|m), & k = q-1 \end{cases}$$

assuming that the mth chip has been transmitted. The cumulative distribution function $F_{R_{nl}}(R|m)$ can be given by

$$F_{R_{nl}}(R|m) = \sqrt{R}\int_0^\infty J_1(\sqrt{R}\lambda)\Phi_{X_{nl}}(\lambda|m)d\lambda \tag{2.21}$$

Define

$$\begin{aligned} C_{nl}(k) &= P_r\{\text{the } n\text{th chip output}=k\nu, \text{ when the number of diversity branches} \\ &\qquad \text{used for combining is } l+1 \mid \text{the } m\text{th chip is transmitted}\} \end{aligned} \tag{2.22}$$

Then, we have

$$C_{n(L-1)}(k) = P_r(Z_n = k\nu| \text{ the } m\text{th bin is transmitted}) \tag{2.23}$$

where $Z_n = \sum_{l=0}^{L-1} Z_{nl}$ is the output of an L-diversity combiner. The conditional probability mass function of Z_n is given by

$$P_{Z_n} = \sum_{k=0}^{L(q-1)} C_{n(L-1)}(k)\delta(z - k\nu) \tag{2.24}$$

With those above equations and the assumption that $\{R_{nl}\}$ are approximately i.i.d. $r.v.$s, the symbol error probability can be expressed as

$$\begin{aligned} P_s(M;L) &= 1 - P_r(\text{correct symbol decision}) \\ &= 1 - \sum_{k=1}^{L(q-1)} C_{m(L-1)}(k) \sum_{i=0}^{M-1} \binom{M-1}{i} \frac{[C_{n(L-1)}(k)]^i}{i+1} \left[\sum_{l=0}^{k-1} C_{n(L-1)}(l)\right]^{M-i-1} \\ &\quad + \frac{C_{m(L-1)}(0)}{M} \left[C_{n(L-1)}(0)\right]^{M-1} \end{aligned} \tag{2.25}$$

Chapter 3

Support Vector Machine Classifiers

In this chapter, we give a brief overview on the Support Vector Machines (SVM) classifiers. We discuss linear SVM classifiers with separable and non-separable cases and then extend the results to nonlinear SVM classifiers by using the "kernel trick". This chapter is organized as follows [19]. After introducing the notion of *maximal margin*, which is an important concept for understanding SVM, we discuss the linear SVM classifiers with separable and non-separable cases. The ensuing section introduces the kernel trick and Mercer condition–a key idea to extend the results from linear to nonlinear. Finally, we explain how to apply the kernel trick to design nonlinear SVM classifiers. For the further readings, please refer to [1, 5, 18, 19]

3.1 Maximal margin

Fig. 3.1 shows a separable classification problem in a two-dimensional space. One can see that there exist several separating lines that completely separate the data of the two classes (labelled by 'x' and '+', respectively). It is natural to ask "among these separating lines, which one is the best?" In order to obtain an unique hyperplane, Vapnik suggested rescaling of the problem such that the points closest to the hyperplane satisfy $|\mathbf{w}^T\mathbf{x}_k + b| = 1$, where $\mathbf{w}$ is the normal vector of the hyperplane and $\mathbf{x}_k$ are the closest points to the hyperplane in the space. The problem can thus be formulated as

$$\max_{\mathbf{w}\in R^n, b\in R} \min\{\|\mathbf{x}-\mathbf{x}_i\| \ |\mathbf{x}, \mathbf{x}_i \in R^n, \ < \mathbf{w}\cdot\mathbf{x} > +b = 0, \ i = 1, \cdots, m\}$$

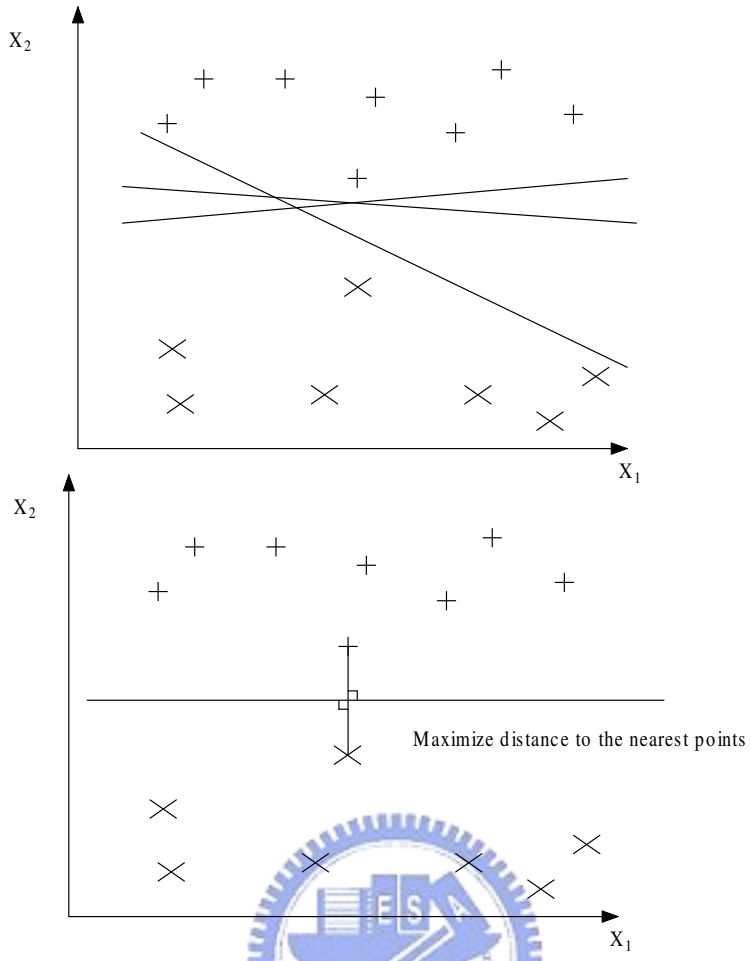


Figure 3.1: (Top) the separating hyperplane is not unique; (Bottom) one define a unique hyperplane corresponding to a maximal distance between the nearest point of the two classes.

subject to the constraint

$$\begin{cases} \mathbf{w}^T\mathbf{x} + b \geqslant +1 & \text{if} \quad y_k = +1 \\ \mathbf{w}^T\mathbf{x} + b \leqslant -1 & \text{if} \quad y_k = -1 \end{cases}$$

where $y_k \in \{-1, +1\}$ is the output decision or the class label. The constraint can be rewritten as

$$y_k(\mathbf{w}^T\mathbf{x} + b) \geqslant +1$$

Notice that the margin equals $2/\|\mathbf{w}\|^2$, as illustrated in Fig. 3.2 ($\|\mathbf{w}\|^2 = \mathbf{w}^T\mathbf{w}$). Hence, the points closest to the hyperplane have a distance $1/\|\mathbf{w}\|$ and the decision boundary

is the hyperplane that maximizes the distance to the nearest points of the two classes. It is obviously that to maximize the distance is equivalent to minimizing $\|\mathbf{w}\|^2$ subject to the constraint $y_k(\mathbf{w}^T\mathbf{x} + b) \geqslant +1$, which is a convex optimization problem. Such an optimization problem is mathematical tractable and there exist several well-known algorithms, e.g., the interior point algorithms.

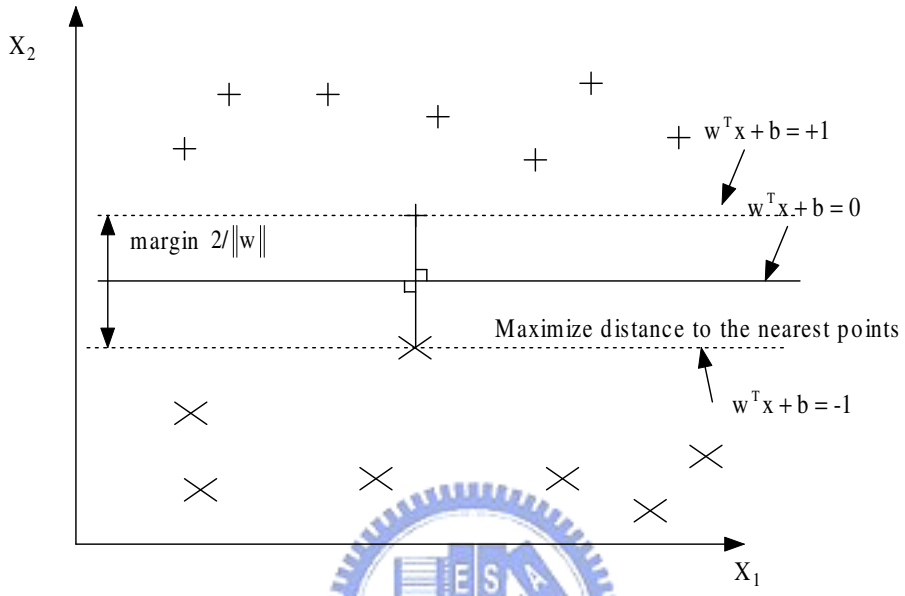


Figure 3.2: The margin is the distance between the dashed lines, which is equal to $2/\|\mathbf{w}\|^2$.

3.2 Linear SVM classifiers with separable case

Consider a given training set $\{\mathbf{x}_k, y_k\}_{k=1}^N$ with input data $\mathbf{x}_k \in \mathbb{R}^n$, output data $y_k \in \{-1, +1\}$ and N is the cardinality of the training point set. As discussed above, a classifier can be regarded as a hyperplane satisfying the maximal margin. The linear classifier can be written as:

$$y(\mathbf{x}) = \text{sign}(\mathbf{w}^T\mathbf{x} + b) \tag{3.1}$$

One would like to have $\mathbf{w}$ and b that maximize the margin subject to the constraint that all training samples are correctly classified. This gives the following primal problem

$$\min_{\mathbf{w},b} J_P(\mathbf{w}) = \tfrac{1}{2}\mathbf{w}^T\mathbf{w}$$
$$\text{subject to} \quad y_k(\mathbf{w}^T\mathbf{x} + b) \geqslant +1,\ k = 1, \ldots N$$

To solve this problem, one introduces the Lagrangian multipliers $\alpha_k \geq 0$ for $k = 1, \ldots, N$

$$\mathcal{L}(\mathbf{w}, b; \alpha) = \frac{1}{2}\mathbf{w}^T\mathbf{w} - \sum_{k=1}^{N} \alpha_k(y_k(\mathbf{w}^T\mathbf{x}_k + b) - 1) \tag{3.2}$$

The solution is given by the saddle point of the Lagrangian

$$\max_{\alpha} \min_{\mathbf{w},b} \mathcal{L}(\mathbf{w}, b; \alpha).$$

Hence one obtains

$$\frac{\partial \mathcal{L}}{\partial \mathbf{w}} = 0 \quad \longrightarrow \quad \mathbf{w} = \sum_{k=1}^{N} \alpha_k y_k \mathbf{x}_k \tag{3.3}$$

$$\frac{\partial \mathcal{L}}{\partial \mathbf{b}} = 0 \quad \longrightarrow \quad \sum_{k=1}^{N} \alpha_k y_k = 0 \tag{3.4}$$

Substituting (3.3) into (3.1), one get

$$y(\mathbf{x}) = \text{sign}(\sum_{k=1}^{N} \alpha_k y_k \mathbf{x}_k^T \mathbf{x} + b) \tag{3.5}$$

By replacing the expression for $\mathbf{w}$ (3.3) in (3.2) and using (3.4), one obtains the following Quadratic Programming (QP) problem as the dual problem in the Lagrange multipliers α_k

$$\max_{\alpha} J_D(\alpha) = -\tfrac{1}{2} \sum_{k,l=1}^{N} y_k y_l \mathbf{x}_k^T \mathbf{x}_l \alpha_k \alpha_l + \sum_{k=1}^{N} \alpha_k$$
$$\text{subject to} \sum_{k=1}^{N} \alpha_k y_k = 0$$

This QP problem has a number of interesting properties:

- Sparseness:

  An interesting property is that many of the resulting α_k values are equal to zero. Hence the obtained solution vector is sparse. This means that in the resulting classifier the sum should be taken only over the non-zero α_k values instead of all training data points:

$$y(\mathbf{x}) = \text{sign}(\sum_{k=1}^{\sharp SV} \alpha_k y_k \mathbf{x}_k^T \mathbf{x} + b)$$

where the index k runs over the number of support vectors, which are the training points corresponding to *non-zero* α_k.

- Global and unique solution:

  The matrix related to α in this quadratic form is positive definite or positive semidefinite. If the matrix is positive definite (all eigenvalues are strictly positive), the solution α to this QP problem is global and unique. When the matrix is positive semidefinite (all eigenvalues are positive or zero), the solution is global but not necessarily unique. There is another interesting fact: the solution of ($\mathbf{w}$,b) is unique but α may be not. In terms of $\mathbf{w} = \sum_{k=1}^{N} \alpha_k y_k \mathbf{x}_k$, this seems to mean that there may exist equivalent expansions of $\mathbf{w}$ which require fewer support vectors.

- Geometrical meaning of support vectors:

  The support vectors obtained from the QP problem are located close to the decision boundary. Notice that the training points located close to the decision boundary may not be support vectors.

- The physical meaning of the support vector [18]

  If we assume that each support vector $\mathbf{x}_i$ exerts a perpendicular force of size α_i and direction $y_j \cdot \mathbf{w}/\|\mathbf{w}\|$ on a solid plane sheet lying along the hyperplane, then the solution satisfies the requirements for mechanical stability. The constraint (3.4) states that the forces on the sheet sum to zero, and (3.3) implies that the torques also sum to zero, via $\sum_i \mathbf{x}_i \times y_i \alpha_i \mathbf{w}/\|\mathbf{w}\| = \mathbf{w} \times \mathbf{w}/\|\text{b}fw\|$

3.3 Linear SVM classifiers with nonseparable case

Most real-world problems that one encounters are nonseparable either in a linear sense or nonlinear sense. For example, consider the classical detection problem: one transmits

a signal $x \in \{-1, +1\}$ and the received signal is $y = x + w$, where w is AWGN. It is well-known that the optimal decision boundary is a linear separating hyperplane according to Bayesian decision theory; see Chapter 1. Linear non-separability is most likely a result of overlapping distributions. For the class of linearly nonseparable cases (with overlapping distributions between the two classes), one is forced to *tolerate misclassifications*. It is done by taking additional slack variables in the problem formulation. In order to tolerate misclassifications, one modifies the set of inequalities as

$$y_k(\mathbf{w}^T\mathbf{x} + b) \geqslant 1 - \xi_k, \quad k = 1, \ldots, N$$

with the slack variables $\xi_k \geq 0$. When $\xi_k > 1$, the kth inequality becomes violated in comparison with the corresponding inequality of the linearly separable case; that is, it is a misclassified point. In the primal weight space the optimization problem (primal problem) becomes:

$$\begin{aligned} &\min_{\mathbf{w},b,\xi} J_P(\mathbf{w}, \xi) = \tfrac{1}{2}\mathbf{w}^T\mathbf{w} + C\sum_{k=1}^{N}\xi_k \\ &\text{subject to } y_k(\mathbf{w}^T\mathbf{x}_k + b) \geq 1 - \xi_k, \quad \xi_k \geq 0 \quad k = 1, \ldots, N \end{aligned}$$

where C is a positive real constant. The following Lagrangian are to be considered

$$\mathcal{L}(\mathbf{w}, b, \xi; \alpha, \nu) = J_P(\mathbf{w}, \xi) - \sum_{k=1}^{N}\alpha_k(y_k(\mathbf{w}^T\mathbf{x}_k + b) - 1 + \xi_k) - \sum_{k=1}^{N}\nu_k\xi_k \tag{3.6}$$

with Lagrange multipliers $\alpha_k \geq 0, \nu_k \geq 0$ for $k = 1, \ldots, N$. The second set of Lagrange multipliers is needed due to the additional slack variables ξ_k. The solution is given by the saddle point of the Lagrangian:

$$\max_{\alpha,\nu}\min_{\mathbf{w},b,\xi}\mathcal{L}(\mathbf{w}, b, \xi; \alpha, \nu)$$

One obtains

$$\frac{\partial\mathcal{L}}{\partial\mathbf{w}} = 0 \quad \longrightarrow \quad \mathbf{w} = \sum_{k=1}^{N}\alpha_k y_k \mathbf{x}_k \tag{3.7}$$

$$\frac{\partial\mathcal{L}}{\partial\mathbf{b}} = 0 \quad \longrightarrow \quad \sum_{k=1}^{N}\alpha_k y_k = 0 \tag{3.8}$$

$$\frac{\partial\mathcal{L}}{\partial\xi_k} = 0 \quad \longrightarrow \quad 0 \leq \alpha_k \leq c, \qquad k = 1, \ldots, N \tag{3.9}$$

By replacing the expression for $\mathbf{w}$ (3.7) in (3.6) and using (3.8), one obtains the following dual QP problem:

$$\max_{\alpha} J_D(\alpha) = -\tfrac{1}{2} \sum_{k,l=1}^{N} y_k y_l \mathbf{x}_k^T \mathbf{x}_l \alpha_k \alpha_l + \sum_{k=1}^{N} \alpha_k$$

subject to

$$\begin{aligned} \sum_{k=1}^{N} & \alpha_k y_k = 0 \\ 0 & \leq \alpha_k \leq c, \quad k = 1, \ldots, N \end{aligned}$$

3.4 Kernel, Kernel trick and Mercer condition

In order to extend the linear method to deal with more general cases which often require nonlinear classifiers, one maps an input data vector $\mathbf{x}$ into $\varphi(\mathbf{x})$ in a high (can be infinite) dimensional feature space $\mathcal{H}$. It is hopped that the transformed data will be linearly separable in the high dimensional feature space. The constructed separating hyperplane in $\mathcal{H}$ may be a nonlinear in the original input space (Fig. 3.3).

It is indeed possible that no explicit specification of the nonlinear mapping $\varphi(\mathbf{x})$ is needed when constructing the separating hyperlane. This is due to the following theorem which leads to the concept of kernel trick.

Theorem 1 *For any symmetric, continuous function $K(\boldsymbol{x}, \boldsymbol{z})$ satisfying the Mercer's condition*

$$\int_{\mathbf{Z}} K(\boldsymbol{x}, \boldsymbol{z}) g(\boldsymbol{x}) g(\boldsymbol{z}) d\boldsymbol{x} d\boldsymbol{z} \geq 0 \tag{3.10}$$

for any square integrable function $g(\boldsymbol{x})$, where $\mathbf{Z}$ is a compact subset of $\mathbb{R}^n$, there exists a Hilbert space $\mathcal{H}$, a map $\varphi : \mathbb{R}^n \longrightarrow \mathcal{H}$ and positive numbers λ_i such that

$$K(\boldsymbol{x}, \boldsymbol{z}) = \sum_{i=1}^{n_{\mathcal{H}}} \lambda_i \varphi_i(\boldsymbol{x}) \varphi_i(\boldsymbol{z}) \tag{3.11}$$

where $\boldsymbol{x}, \boldsymbol{z} \in \mathbb{R}^n$ and $n_{\mathcal{H}} \leq \infty$ is the dimension of $\mathcal{H}$.

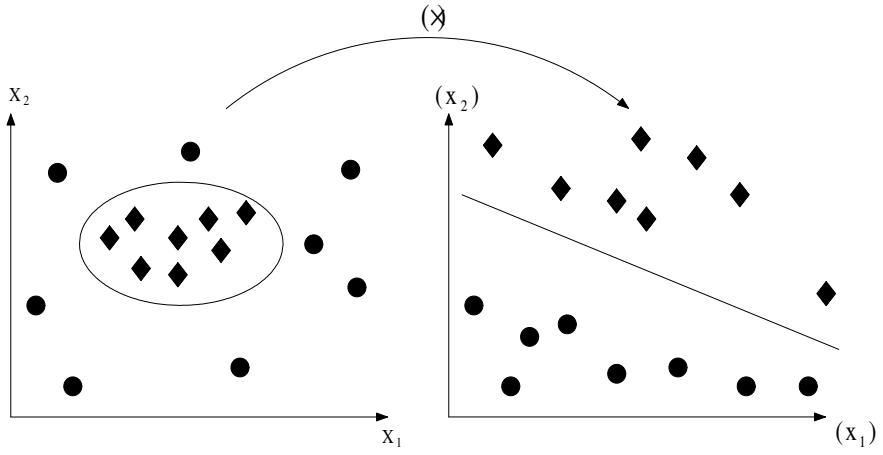


Figure 3.3: The relation between input space and feature space.

Notice that (3.11) can be rewritten as

$$K(\mathbf{x}, \mathbf{z}) = \sum_{i=1}^{n_{\mathcal{H}}} \sqrt{\lambda_i}\varphi_i(\mathbf{x})\sqrt{\lambda_i}\varphi_i(\mathbf{z}) \tag{3.12}$$

One can define $\tilde{\varphi}_i(\mathbf{x}) = \sqrt{\lambda_i}\varphi_i(\mathbf{x})$ and $\tilde{\varphi}_i(\mathbf{z}) = \sqrt{\lambda_i}\varphi_i(\mathbf{z})$ so that an inner product in the feature space is equivalent to the kernel function $K(\mathbf{x}, \mathbf{z})$ in the input space

$$\langle \tilde{\varphi}(\mathbf{x}), \tilde{\varphi}(\mathbf{z}) \rangle = \tilde{\varphi}(\mathbf{x})^T \tilde{\varphi}(\mathbf{z}) = K(\mathbf{x}, \mathbf{z}). \tag{3.13}$$

Hence, having a positive semidefinite kernel (one satisfies (3.10) is a condition to guarantee that one can compute the kernel of the input data vectors $\mathbf{x}, \mathbf{z}$ without the need to construct $\varphi(\cdot)$. In other words, it is possible to find the separating hyperplane without explicitly performing the nonlinear mapping. The application of (3.13) is often called the *kernel trick*. It enables one to work in high dimensional feature spaces without actually having to do explicit computations in this space.

There are several choices for the kernel $K(\cdot, \cdot)$. Some typical choices are

$$K(\mathbf{x}, \mathbf{x}_k) = \mathbf{x}^T\mathbf{x}_k \text{ (linear SVM)} \tag{3.14a}$$

$$K(\mathbf{x}, \mathbf{x}_k) = (\tau + \mathbf{x}^T\mathbf{x}_k)^d \text{ (polynomial SVM of degree } d) \tag{3.14b}$$

$$K(\mathbf{x}, \mathbf{x}_k) = \exp(-\|\mathbf{x} - \mathbf{x}_k\|^2/\sigma^2) \text{ (RBF kernel)} \tag{3.14c}$$

$$K(\mathbf{x}, \mathbf{x}_k) = \tanh(\kappa_1\mathbf{x}^T\mathbf{x}_k + \kappa_2) \text{ (MLP kernel)} \tag{3.14d}$$

The Mercer condition holds for all σ in the RBF kernel case and all $\tau > 0$ in the polynomial case but not for all possible choices of κ_1, κ_2 in the MLP kernel case.

It is worth noticing that one can create kernels from kernels. The main idea is to perform some operations on positive definite kernels such that the results still remain positive definite. For symmetric positive definite kernels the following operations are allowed [5]:

- $K(\mathbf{x}, \mathbf{z}) = aK_1(\mathbf{x}, \mathbf{z})$ $(a > 0)$
- $K(\mathbf{x}, \mathbf{z}) = K_1(\mathbf{x}, \mathbf{z}) + b$ $(b > 0)$
- $K(\mathbf{x}, \mathbf{z}) = \mathbf{x}^T P \mathbf{z}$ $(P = P^T > 0)$
- $K(\mathbf{x}, \mathbf{z}) = K_1(\mathbf{x}, \mathbf{z}) + K_2(\mathbf{x}, \mathbf{z})$
- $K(\mathbf{x}, \mathbf{z}) = K_1(\mathbf{x}, \mathbf{z})K_2(\mathbf{x}, \mathbf{z})$
- $K(\mathbf{x}, \mathbf{z}) = f(\mathbf{x})f(\mathbf{z})$
- $K(\mathbf{x}, \mathbf{z}) = K_1(\phi(\mathbf{x}), \phi(\mathbf{z}))$
- $K(\mathbf{x}, \mathbf{z}) = p_+(K_1(\mathbf{x}, \mathbf{z}))$
- $K(\mathbf{x}, \mathbf{z}) = \exp(K_1(\mathbf{x}, \mathbf{z}))$

where $a, b \in \mathbb{R}^+$ and K_1, K_2 are symmetric positive definite kernel functions, $f(\cdot) : \mathbb{R}^n \to \mathbb{R}$, $\phi(\cdot) : \mathbb{R}^n \to \mathbb{R}^{n_{\mathcal{H}}}$ and $p_+(\cdot)$ is a polynomial with positive coefficients.

One may also normalize kernels:

$$\widetilde{K}(\mathbf{x}, \mathbf{z}) = \frac{K(\mathbf{x}, \mathbf{z})}{\sqrt{K(\mathbf{x}, \mathbf{x})}\sqrt{K(\mathbf{z}, \mathbf{z})}} = \cos(\theta_{\varphi(\mathbf{x}),\varphi(\mathbf{z})}) \tag{3.15}$$

where $\widetilde{K}$ is the normalized kernel and $\theta_{\varphi(\mathbf{x}),\varphi(\mathbf{z})}$ is the angle between $\varphi(\mathbf{x}), \varphi(\mathbf{z})$ in the feature space.

3.5 Nonlinear SVM classifiers

The extension from linear SVM classifiers to nonlinear SVM classifiers is straightforward. As discussed in section 3.4, one maps the input data into a high dimensional feature space. It means that one replaces $\mathbf{x}$ by $\varphi(\mathbf{x})$. Therefore, the primal problem for nonlinear SVM classifiers with nonseparable case is:

$$\min_{\mathbf{w},b,\xi} J_P(\mathbf{w},\xi) = \tfrac{1}{2}\mathbf{w}^T\mathbf{w} + C\sum_{k=1}^{N}\xi_k$$

subject to

$$y_k(\mathbf{w}^T\varphi(\mathbf{x}_k) + b) \geq 1 - \xi_k \tag{3.16}$$

$$\xi_k \geq 0 \quad k = 1,\ldots,N \tag{3.17}$$

and the representation of nonlinear classifier is :

$$y(\mathbf{x}) = \text{sign}(\mathbf{w}^T\varphi(\mathbf{x}) + b) \tag{3.18}$$

One should note that the dimension of $\mathbf{w}$ is $n_{\mathcal{H}}$, not n (the dimension of the input space). As discuss in section 3.4, $n_{\mathcal{H}}$ can be infinite in general. It means that it is hard or even impossible to solve the primal problem for large $n_{\mathcal{H}}$. Fortunately, one can solve this problem by considering the dual problem. Similar to the work in section 3.3, one introduces the Lagrangian multipliers:

$$\mathcal{L}(\mathbf{w},b,\xi;\alpha,\nu) = J_P(\mathbf{w},\xi) - \sum_{k=1}^{N}\nu_k\xi_k - \sum_{k=1}^{N}\alpha_k(y_k(\mathbf{w}^T\varphi(\mathbf{x}_k)+b) - 1 + \xi_k) \tag{3.19}$$

with Lagrange multipliers $\alpha_k \geq 0, \nu_k \geq 0$ for $k = 1,\ldots,N$. The solution is given by the saddle point of the Lagrangian:

$$\max_{\alpha,\nu}\min_{\mathbf{w},b,\xi}\mathcal{L}(\mathbf{w},b,\xi;\alpha,\nu)$$

One obtains

$$\frac{\partial \mathcal{L}}{\partial \mathbf{w}} = 0 \quad \longrightarrow \quad \mathbf{w} = \sum_{k=1}^{N} \alpha_k y_k \varphi(\mathbf{x}_k) \tag{3.20}$$

$$\frac{\partial \mathcal{L}}{\partial \mathbf{b}} = 0 \quad \longrightarrow \quad \sum_{k=1}^{N} \alpha_k y_k = 0 \tag{3.21}$$

$$\frac{\partial \mathcal{L}}{\partial \xi_k} = 0 \quad \longrightarrow \quad 0 \leq \alpha_k \leq c, \qquad k = 1, \ldots, N \tag{3.22}$$

By replacing the expression for $\mathbf{w}$(3.20) in (3.19) and using (3.21), one obtains the following dual QP problem:

$$\max_{\alpha} J_D(\alpha) = -\frac{1}{2} \sum_{k,l=1}^{N} y_k y_l K(\mathbf{x}_k, \mathbf{x}_l) \alpha_k \alpha_l + \sum_{k=1}^{N} \alpha_k \tag{3.23}$$

subject to

$$\begin{aligned} \sum_{k=1}^{N} \quad & \alpha_k y_k = 0 \\ 0 \quad & \leq \alpha_k \leq c, \qquad k = 1, \ldots, N \end{aligned}$$

Then, the nonlinear classifier becomes (substitute (3.20) into (3.18)):

$$y(\mathbf{x}) = \text{sign}(\sum_{k=1}^{N} \alpha_k y_k K(\mathbf{x}_k, \mathbf{x}) + b)$$

Notice that the computation is in the input space, and the dimension of the input space is usually finite and not very large. Therefore, one can solve this problem in dual space while can not in primal space.

As denoted in section 3.2, this QP problem has a number of interesting properties:

- Sparseness:

  As discussed before, many α_k values are equal to zero. Hence the obtained solution vector is sparse. This means that in the resulting classifier the sum should be taken only over the non-zero α_k values instead of all training data points as in the linear case:

$$y(\mathbf{x}) = \text{sign}(\sum_{k=1}^{\sharp SV} \alpha_k y_k K(\mathbf{x}_k, \mathbf{x}) + b)$$

where the index k runs over the number of support vectors.

- Global and unique solution:

  As in the linear SVM case, the solution to the convex QP problem is global and unique if one choose a positive definite kernel. This choice guarantees that the matrix involved in the QP problem is positive definite and that one can apply kernel trick as well.

- Geometrical meaning of support vectors:

  The support vectors obtained from the QP problem are located close to the decision boundary. Notice that the training points located close to the decision boundary may not be support vectors.

Chapter 4

Detector Structures Based on SVM Techniques

Two critical issues arise when applying the SVM scheme to solve our detection problem, namely, the training speed and the multi-class classification issues. We will describe a fast algorithm for training support vector machine–the Sequential Minimal Optimization (SMO) algorithm. This algorithm breaks a large QP optimization problem into a series of smallest possible QP tasks. These small QP tasks are solved analytically, which avoids using a time-consuming numerical QP optimization as an inner loop. We then discuss the Directed Acyclic Graph (DAG) method that combines many two-class classifiers into a multi-class classifier. For an N-class problem, the DAG contains $N(N-1)/2$ classifiers, one for each pair of classes.

By applying the DAG algorithm to classify the computer generated training samples and using the SMO algorithm, we obtain the support vectors for binary classification. Given these support vectors, we then apply the DAG method for real-time signal detection. Thus the design of a SVM-based FHMA/MFSK detector requires an off-line (non-real time) training period to find the associated support vectors. Once these support vectors become available, we place them in the detector memory and on-line detection just follows a predetermined DAG algorithm using the known support vectors. Before giving numerical application examples, we summarize the major attributes of SMO and DAG algorithms in the following sections.

4.1 The SMO Algorithm

The architecture of the SMO algorithm can be summarized as follows.

M1. The outer loop: select the first Lagrange multipliers to be optimized.

M2. The inner loop: select the second Lagrange multipliers to be optimized.

M3. Solving for two selected Lagrange multipliers analytically.

We now proceed to address [SMO3] and then [SMO1] and [SMO2], respectively.

4.1.1 Analytical Lagrange multiplier solutions

Each step of SMO will optimize two Lagrange multipliers. Without loss of generality, let us denote these two multipliers by α_1 and α_2, respectively. The objective function (3.23) can be rewritten as

$$\min_{\alpha} J_D(\alpha) = \frac{1}{2}\sum_{k,l=1}^{N} y_k y_l K(\mathbf{x}_k, \mathbf{x}_l)\alpha_k\alpha_l - \sum_{k=1}^{N}\alpha_k$$

$$= \frac{1}{2}K_{11}\alpha_1^2 + \frac{1}{2}K_{22}\alpha_2^2 + y_1y_2K_{12}\alpha_1\alpha_2 + y_1\alpha_1 v_1 + y_2\alpha_2 v_2 - \alpha_1 - \alpha_2 + \text{residual terms} \quad (3.1)$$

where

$$\begin{aligned} K_{ij} &= K(\mathbf{x}_i, \mathbf{x}_j) \\ v_i &= \sum_{j=3}^{N} y_j\alpha_j^* K_{ij} = u_i - b^* - y_1\alpha_1^* K_{1i} - y_2\alpha_2^* K_{2i} \\ u_i &= \sum_{j=1}^{N} y_j\alpha_j^* K_{j,i} + b^* \end{aligned}$$

and the starred variables indicate the values at the end of the previous iteration. For the initial state, one can start with $\alpha_i^* = 0$ for $i = 1, \ldots, N$. Residual terms do not depend on either α_1 or α_2. From (3.21), we obtain

$$\alpha_1 + s\alpha_2 = \alpha_1^* + s\alpha_2^* = \kappa \quad (3.2)$$

where $s = y_1 y_2$. Then (3.1) can be represented in terms of α_2 only.

$$\begin{aligned} J_D &= \frac{1}{2}K_{11}(\kappa - s\alpha_2)^2 + \frac{1}{2}K_{22}\alpha_2^2 + sK_{12}(\kappa - s\alpha_2)\alpha_2 + y_1(\kappa - s\alpha_2)v_1 \\ &\quad + y_2\alpha_2 v_2 - (\kappa - s\alpha_2) - \alpha_2 + \text{residual terms} \end{aligned} \tag{3.3}$$

The first and second derivatives are given by

$$\begin{aligned} \frac{dJ_D}{d\alpha_2} &= -sK_{11}(\kappa - s\alpha_2) + K_{22}\alpha_2 - K_{12}\alpha_2 \\ &\quad + sK_{12}(\kappa - s\alpha_2) - y_2 v_1 + s - 1 + y_2 v_2 \end{aligned} \tag{3.4}$$

$$\frac{d^2 J_D}{d\alpha_2^2} = K_{11} + K_{22} - 2K_{12} = \eta \tag{3.5}$$

Notice that $\eta \geq 0$ if $K(\cdot, \cdot)$ satisfies the Mercer's condition. (3.4) can be rearrange as:

$$\alpha_2 \eta = \alpha_2^* \eta + y_2(E_1 - E_2) \Longrightarrow \alpha_2^{\text{new}} = \alpha_2^* + \frac{y_2(E_1 - E_2)}{\eta} \tag{3.6}$$

where $E_i = u_i - y_i$. Because $0 \leq \alpha_2 \leq C$, we need to check the admissible range of α_2, which is determined by the following rules:

- If $s = 1$, then $\alpha_1 + \alpha_2 = \kappa$ (Fig. (4.1))
  - If $\kappa > C$, the max of $\alpha_2 = C$ and the min of $\alpha_2 = \kappa - C$
  - If $\kappa < C$, the max of $\alpha_2 = \kappa$ and the min of $\alpha_2 = 0$

  $\Longrightarrow$ $L = \max(0, \kappa - C)$ and $H = \min(C, \kappa)$.
- If $s = -1$, then $\alpha_1 - \alpha_2 = \kappa$ (Fig. (4.2))
  - If $\kappa > 0$, the max of $\alpha_2 = C - \kappa$ and the min of $\alpha_2 = 0$
  - If $\kappa < 0$, the max of $\alpha_2 = C$ and the min of $\alpha_2 = -\kappa$

  $\Longrightarrow$ $L = \max(0, -\kappa)$ and $H = \min(C, C - \kappa)$.

Hence, we have

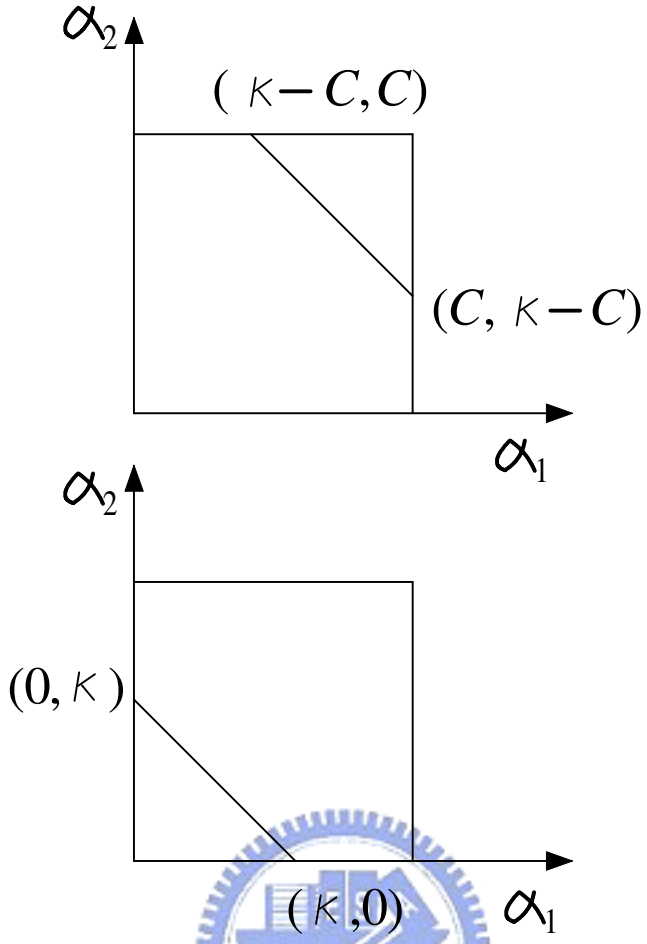


Figure 4.1: Admissible ranges of α_2 for the cases $\alpha_1 + \alpha_2 = \kappa$ and $\kappa > C$ (top) or $\kappa < C$ (Bottom).

$$\alpha_2^{\text{new,clipped}} = \begin{cases} H, & \text{if } H < \alpha_2^{\text{new}} \\ \alpha_2^{\text{new}}, & \text{if } L \leq \alpha_2^{\text{new}} \leq H \\ L, & \text{if } \alpha_2^{\text{new}} < L \end{cases}$$

To summarize, given α_1, α_2 (and the corresponding $y_1, y_2, K_{11}, K_{12}, K_{22}, E_1 - E_2$), we optimize the two α's by the following procedure:

O1. $Let \eta = K_{11} + K_{22} - 2K_{12}$.

O2. If $\eta > 0$, set

$$\triangle\alpha_2 = \alpha_2^{\text{new}} - \alpha_2^* = \frac{y_2(E_1 - E_2)}{\eta}, \tag{3.7}$$

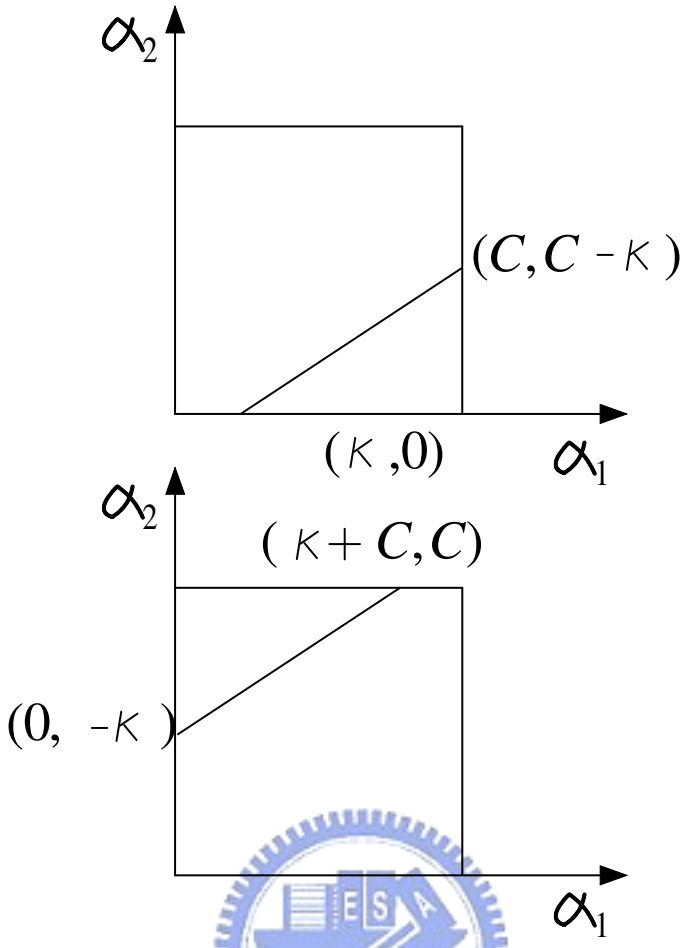


Figure 4.2: Admissible ranges of α_2 for the cases $\alpha_1 - \alpha_2 = \kappa$ and $\kappa > 0$ (top) or $\kappa < 0$ (Bottom).

and clip the solution within the admissible region ($\alpha_2^{\text{new,clipped}}$) so that

$$\triangle\alpha_1 = -s\triangle\alpha_2 \tag{3.8}$$

O3. If $\eta = 0$, we need to evaluate the objective function at the two endpoints and set α_2^{new} to be the one with smaller objective function value. The objective function is given by (See Appendix A)

$$J_D = \frac{1}{2}\eta\alpha_2^2 + (y_2(E_2 - E_1) - \alpha_2^*\eta)\alpha_2 + \text{Const.} \tag{3.9}$$

.

4.1.2 Choices of the Lagrange Multipliers

Consider the KKT (Karush-Kuhn-Tuker) conditions of the primal problem which will be used in picking two Lagrange multipliers stage. The conditions are

$$\frac{\partial \mathcal{L}}{\partial \mathbf{w}} = 0 \quad \longrightarrow \quad \mathbf{w} = \sum_{k=1}^{N} \alpha_k y_k \varphi(\mathbf{x}_k) \tag{3.10}$$

$$\frac{\partial \mathcal{L}}{\partial \mathbf{b}} = 0 \quad \longrightarrow \quad \sum_{k=1}^{N} \alpha_k y_k = 0 \tag{3.11}$$

$$\frac{\partial \mathcal{L}}{\partial \xi_k} = 0 \quad \longrightarrow \quad 0 \leq \alpha_k \leq C, \quad k = 1, \ldots, N \tag{3.12}$$

$$\alpha_k(y_k(\mathbf{w}^T \varphi(\mathbf{x}_k) - b) + \xi_k - 1) \quad = \quad 0 \quad k = 1, \ldots, N \tag{3.13}$$

$$\nu_k \xi_k \quad = \quad 0 \quad k = 1, \ldots, N \tag{3.14}$$

which imply

$$\alpha_i = 0 \implies y_i(\mathbf{w}^T \mathbf{x}_i - b) \geq 1$$

$$0 < \alpha_i < C \implies y_i(\mathbf{w}^T \mathbf{x}_i - b) = 1$$

$$\alpha_i = C \implies y_i(\mathbf{w}^T \mathbf{x}_i - b) \leq 1$$

The heuristics for picking two α_i's for optimization are as follows [13]:

- The outer loop selects the first α_i, the inner loop selects the second α_i that maximizes $|E_2 - E_1|$.
- The outer loop alternates between one sweep through all examples and as many sweeps as possible through the non-boundary examples, selecting the example that violates the KKT condition. Once a violated example is found, a second multiplier is chosen using the second choice heuristic, and the two multipliers are jointly optimized. Then, the SVM is updated according to these two new multipliers, and the outer loop continue to find other violated examples.
- Given the first α_i, the inner loop looks for a non-boundary that maximizes $|E_2 - E_1|$. If this does not make progress, it starts a sequential scan through the non-boundary

examples, starting at a random position; if the fails too, it starts a sequential scan through all the examples, also starting at a random position. If all of above fail, the first α_i is skipped and the inner loop will continue with another chosen first α_i from outer loop.

4.1.3 Update step

Given the incremental changes $\triangle\alpha_1, \triangle\alpha_2$, we update the corresponding E_i's, and b accordingly. Let $E(\mathbf{x}, y)$ be the prediction error of $(\mathbf{x}, y)$

$$E(\mathbf{x}, y) = \sum_{i=1}^{N} \alpha_i y_i K(\mathbf{x}_i, \mathbf{x}) + b - y \tag{3.15}$$

The corresponding variation of E is

$$\triangle E(\mathbf{x}, y) = \triangle\alpha_1 y_1 K(\mathbf{x}_1, \mathbf{x}) + \triangle\alpha_2 y_2 K(\mathbf{x}_2, \mathbf{x}) + \triangle b \tag{3.16}$$

The change in the threshold can be computed by forcing $E_1^{\text{new}} = 0$, if $0 < \alpha_1^{\text{new}} < C$, or $E_2^{\text{new}} = 0$, if $0 < \alpha_2^{\text{new}} < C$). The reason for forcing $E_1 = 0$ is that if $0 < \alpha_1 < C$, the output of SVM is y_1 when the input is x_1 (from the KKT conditions) and by the definition of E_1, $E_1 = 0$. From

$$\begin{aligned} 0 &= E(\mathbf{x}, y)^{\text{new}} \\ &= E(\mathbf{x}, y)^{\text{old}} + \triangle E(\mathbf{x}, y) \\ &= E(\mathbf{x}, y)^{\text{old}} + \triangle\alpha_1 y_1 K(\mathbf{x}_1, \mathbf{x}) + \triangle\alpha_2 y_2 K(\mathbf{x}_2, \mathbf{x}) + \triangle b \end{aligned} \tag{3.17}$$

where $\mathbf{x} = \mathbf{x}_1$ if $0 < \alpha_1^{\text{new}} < C$ and $\mathbf{x} = \mathbf{x}_2$ if $0 < \alpha_2^{\text{new}} < C$. Hence, we obtain

$$-\triangle b = E(\mathbf{x}, y)^{\text{old}} + \triangle\alpha_1 y_1 K(\mathbf{x}_1, \mathbf{x}) + \triangle\alpha_2 y_2 K(\mathbf{x}_2, \mathbf{x}) \tag{3.18}$$

If both α_1, α_2 are equal to 0 or C, we compute two values of the new b for α_1 and α_2 using (3.18), and take the average. Notice that if $0 < \alpha_i^{\text{new}} < C$ for $i = 1, 2$, the $\triangle b$ will be identical.

4.2 Directed Acyclic Graph (DAG) Algorithm

In general, there are two types of approaches for multi-class SVM:

D1. Construct a multi-class classifier by combining several binary classifiers.

D2. Construct a multi-class classifier by solving one optimization problem with all classes [19].

In this section, we focus on Approach D1 and present the DAG algorithm for the multi-class problems. We first describe two multi-class algorithms (*1-v-r* and Max-Wins) for comparison, and then give the details of the DAG algorithm and discuss some results about DAG algorithm.

1-v-r (abbreviation for one-versus-the rest) is a standard method for N-class SVMs. It constructs N SVMs with the i-th SVM trained by all of the data and the labels of data changed according to

$$y_i = \begin{cases} 1 & \text{data is in the } i\text{-th class} \\ -1 & \text{otherwise.} \end{cases}$$

The output is determined by the index (i) of $y_i = 1$.

Max-Wins constructs $\frac{N(N-1)}{2}$ SVMs with each SVM trained on only two out of N classes. This class of SVMs are often referred to as one-verse-one (*1-v-1*) SVMs in multi-class problems. For combining these binary SVMs, each SVM casts one vote for its preferred class, and the final result is the class with the majority votes.

Before discussing the DAG algorithm, we need to introduce the notion of DAG and Decision DAG (DDAG). A DAG, as the name implies, is a graph whose edges have an orientation but form no cycles. A *rooted* DAG has a unique node (rooted node) which there is no edge pointing into it. A *binary* DAG is a graph in which every node has either 0 or 2 edges leaving it. A DDAG is a rooted binary DAG with N leaves labelled by the classes. At each of the $\frac{N(N-1)}{2}$ internal nodes in a DDAG, a Boolean decision is made. The nodes are arranged in a triangle-like shape in which the i-th node in layer

$j < N$ is connected to the ith and $(i+1)$th node in the $j+1$th layer. A typical DDAG with $N = 4$ is illustrated in Fig. 4.3. It is straight froward to use the DAG algorithm for

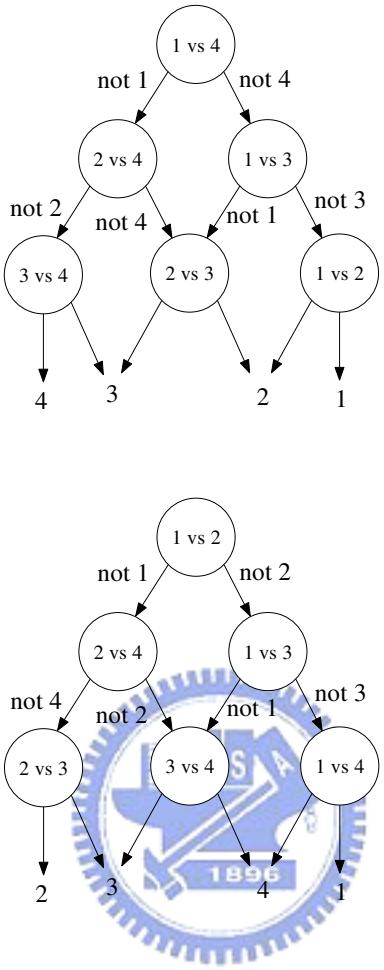


Figure 4.3: A DDAG with $N = 4$ in natural numerical order (top). (Bottom) Due to the frequency of occurrence ($1 \leq 2 < 3 \leq 4$), one may interchange the positions of 2 and 4.

SVM: using two-class maximal margin hyperplanes at each internal node of the DDAG. In general, we refer this implementation as DAGSVM algorithm for emphasizing SVM. The choices of the class order in the DDAG is arbitrary. One possible heuristic is to place the most frequent classes in the center, as illustrated by Fig.4.3. Yet, reordering the order does not yield significant changes in accuracy performance. Also, It has been experimentally shown that the DAGSVM algorithm is faster to train or evaluate than

either the *1-v-r* or Max-Wins, while maintaining comparable accuracy to both of these algorithms [12].

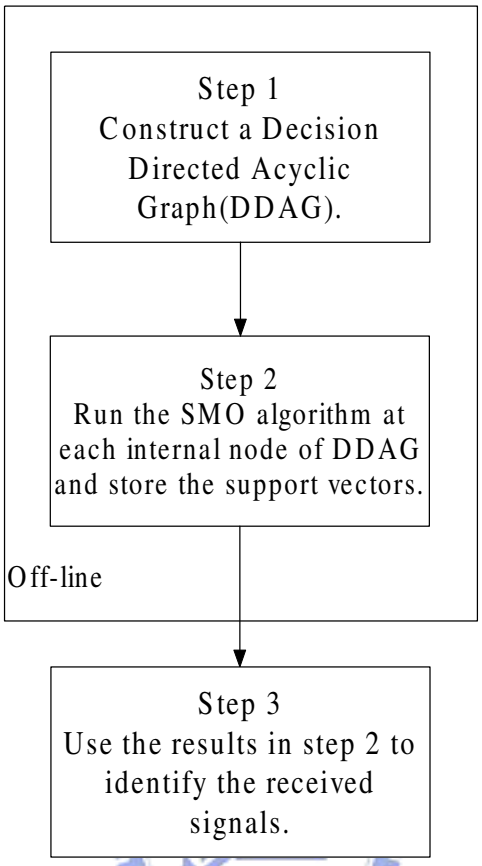


Figure 4.4: The flow chart to implement the proposed SVM-based receivers.

To implement the proposed SVM-based receivers, there are three steps, as illustrated by Fig. 4.4.

S1. Construct a decision directed acyclic graph(DDAG).

S2. At each internal node of the DDAG, one uses the SMO algorithm to find the two-class maximal margin hyperplane and store the support vectors(SVs).

S3. Use the stored SVs, the SVM-based receiver can identify the received signals.

Note that the step 1. and step 2. can be carried out off-line. It can, therefore, save the test period.

4.3 Numerical Examples and Discussion

Unless otherwise specified, all SVM-based receivers use the RBF kernel with $\sigma = 0.5$ and the following system parameters are assumed in all the examples given in this section: $M = 4$, $L = 1$, $U = 1$, and $\eta = 1$. The reason to choice the RBF kernel is due to the fact that it has been proved that this kernel is universal, which means that linear combinations of this kernel computed at the data points can approximate any function[11].

Example 1 *4FSK in AWGN channels.*

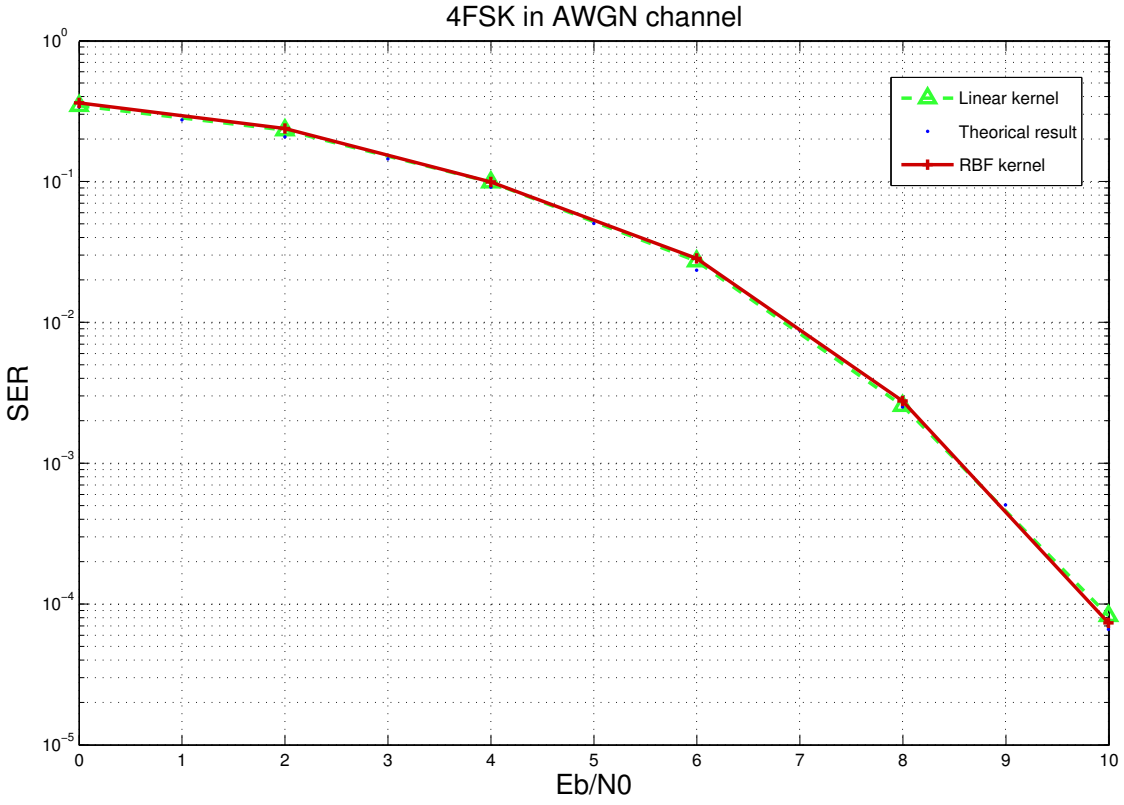


Figure 4.5: Symbol error rate performance of optimum and SVM-based 4FSK receivers in AWGN channel.

Fig. 4.5 reveals that a receiver based on the SVM technique can provide performance very close to that achieved by a conventional optimal MFSK receiver. More specifically,

the probability of symbol error for the optimal noncoherent M-ary orthogonal signal detector is given by [16]

$$P_M(e) = \sum_{n=1}^{M-1} (-1)^{n+1} \binom{M-1}{n} \frac{1}{n+1} exp\left[-\frac{n log_2 M}{n+1} \frac{E_b}{N_0} \right] \tag{3.19}$$

The receiver whose performance is given by the dashed curve uses RBF kernel, 600 *training samples and* $C = 1$. *The solid one with linear kernel,* 600 *training samples and* $C = 6$ *is plotted as well. The circle points are calculated according to (3.19). Both receivers yield almost the same performance as that of the optimum receiver.*

Example 2 *4FSK in a single-user AWGN channel with diversity order* $L = 4$.

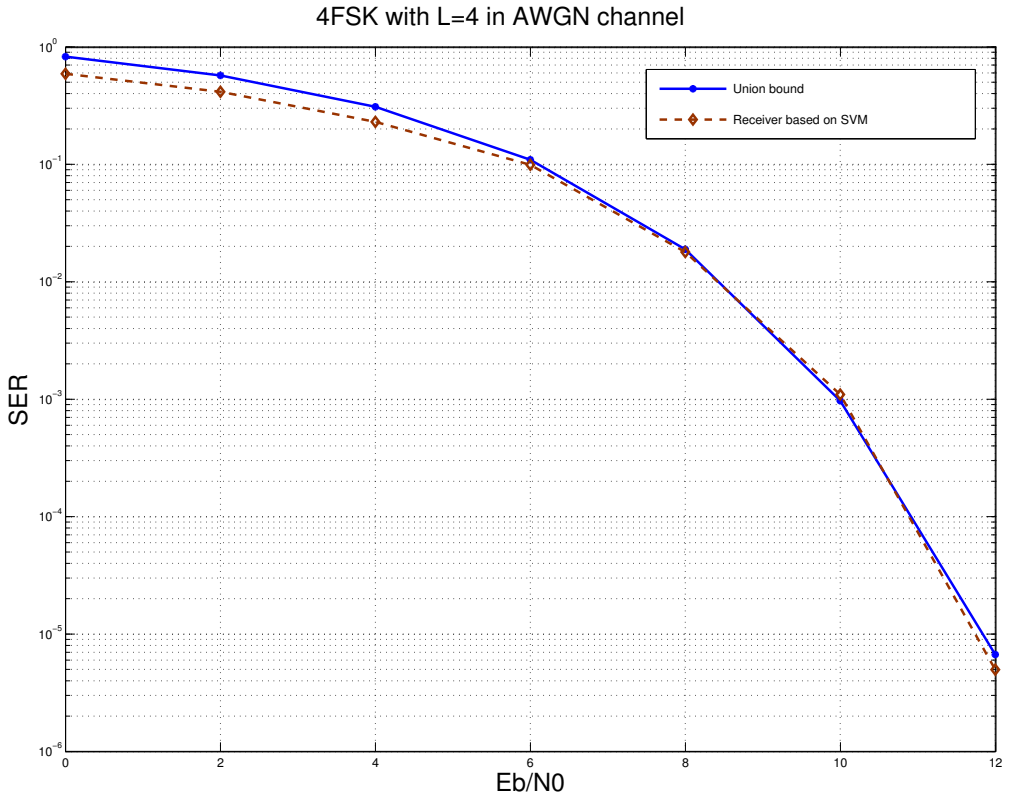


Figure 4.6: Symbol error rate performance of the optimum and SVM-based 4FSK receivers in AWGN channel and diversity order $L = 4$.

In Fig. 4.6, we plot the performance curves for the SVM-based receiver and the optimal noncoherent 4FSK receiver. The performance of the latter is computed by using

the union bound [16] for general M-ary orthogonal signal with diversity L.

$$P_M(e) < (M-1)P_2(e;L) \tag{3.20}$$

where

$$P_2(e;L) = \frac{1}{2^{2L-1}} e^{-\frac{kr_b}{2}} \sum_{n=0}^{L-1} c_n (\frac{1}{2} kr_b)^n \tag{3.21}$$

and

$$c_n = \frac{1}{n!} \sum_{k=0}^{L-1-n} \binom{2L-1}{k} \tag{3.22}$$

r_b being the SNR per bit and $k = log_2 M$.

The solid line corresponds to the performance obtained with 600 training samples and $C = 0.9$. It is obvious that the performance is almost identical to that of the ideal noncoherent 4FSK receiver except in low SNRs. It is known that [16], for relatively small values of M, the union bound is sufficiently tight for most practical applications. We conclude that the MFSK receiver based on SVM can offer performance very close or identical to the optimal receiver when diversity reception is involved.

Example 3 *4FSK in Rayleigh fading channels.*

As shown in Fig. 4.7, the performance of the proposed receiver is similar to that of the conventional optimal receiver (matched filter). The solid curve is the SER performance of the SVM-based receiver with 1800 training samples and $C = 0.9$. Again, we can see that the proposed SVM-based receiver provides performance identical to that of the conventional optimal receiver.

Example 4 *4FSK in frequency nonselective Rayleigh fading channels with diversity order $L = 4$.*

The SVM-based receiver is obtained with 1800 training samples and $C = 0.1$. The threshold b is chosen to be $\sqrt{2 * (1 + \frac{1}{SNR}) ln(1 + SNR)} * N_0$ according to [14].

The solid curve in Fig.4.8 represents the performance of the receiver proposed by Goodman et al. [9] and the dashed curve is the performance of our SVM-based receiver. In this case our receiver outperforms Goodman's, which is only a suboptimal receiver.

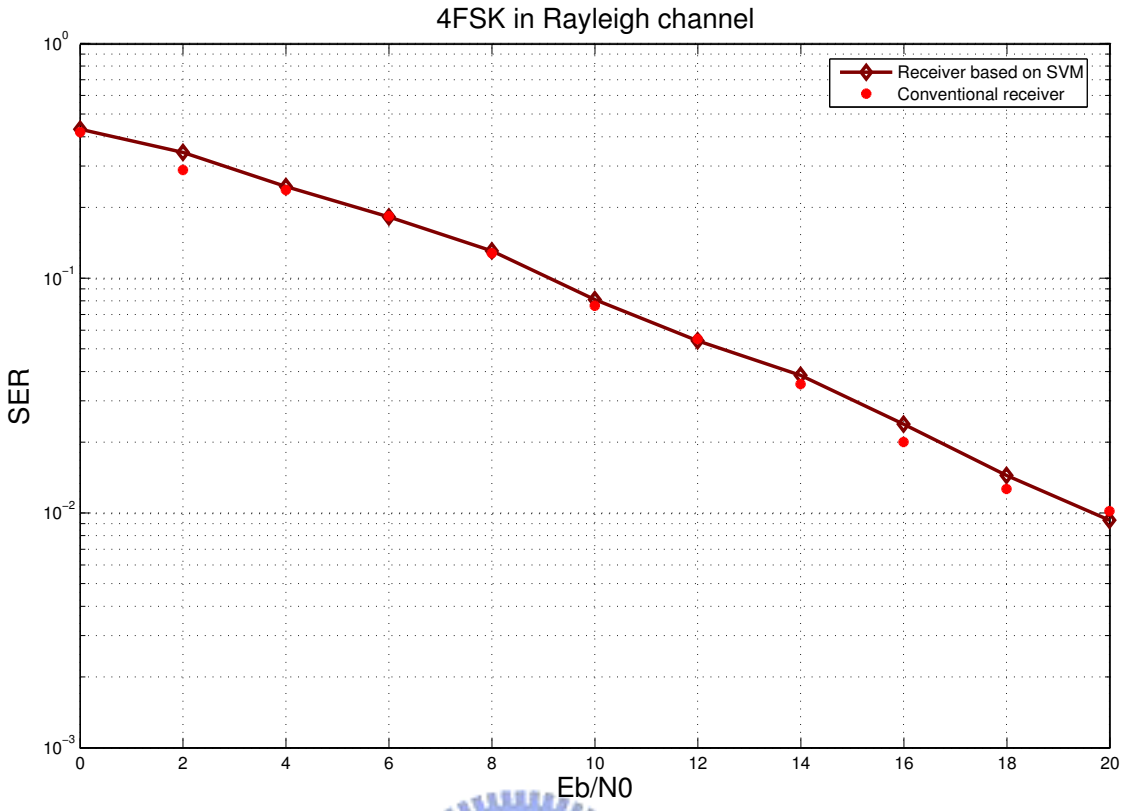


Figure 4.7: Performance comparison of optimum and SVM-based 4FSK receivers in Rayleigh fading channels.

Example 5 *4FSK in a Rayleigh fading channel with diversity order* $L = 4$, $B = 2$ *and two system users.*

The dashed curve in Fig.4.9 is the performance of our SVM-based receiver. The proposed receiver is achieved with 1800 training samples and $C = 0.1$. *The solid line is the performance of the receiver proposed by [17] with* $q = 8$. *It is clear from Fig. 4.9 that our proposed receiver outperform that in [17] too.*

Example 6 *4FSK in Rayleigh fading channels with diversity order* $L = 4$, $B = 2$, $\eta = 1.2$ *and two system users (one interferer).*

The dashed line in Fig.4.10 is the performance of our SVM-based receiver. The proposed receiver is obtained with 1800 training samples and $C = 0.1$. *The solid line is*

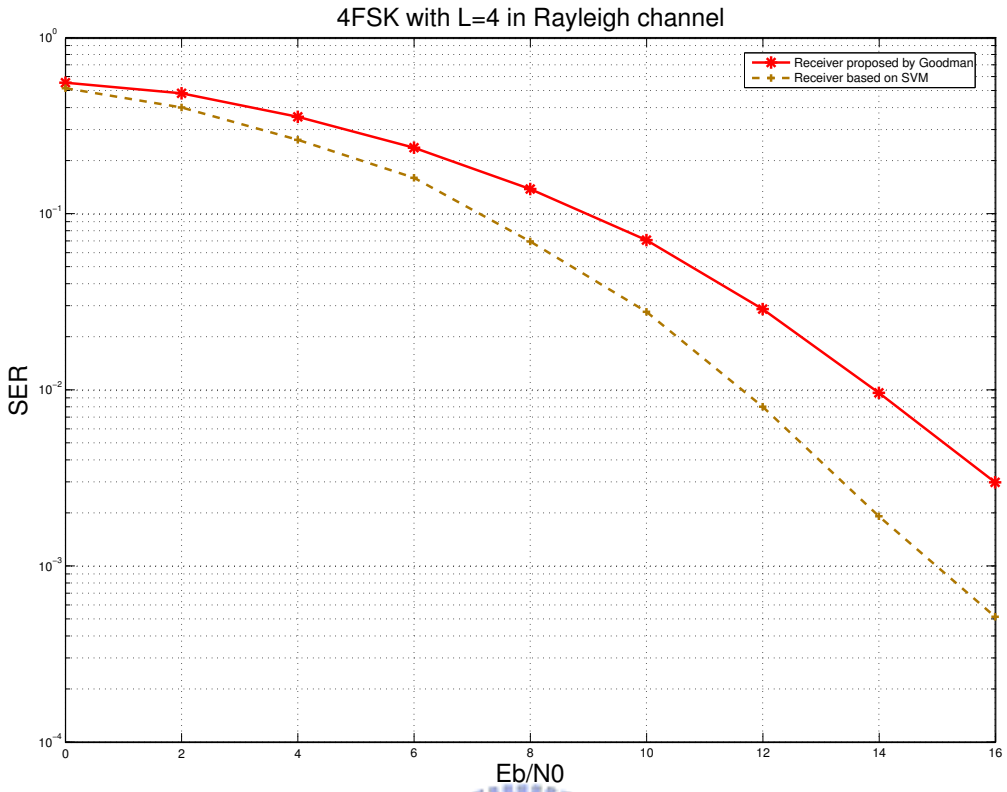


Figure 4.8: Performance comparison of Goodman's and SVM-based 4FSK receivers in Rayleigh fading channels.

the performance of receiver proposed in [17] with $q = 8$. *Again, Fig. 4.10 shows that our proposed receiver outperforms that in [17].*

Example 7 Robustness of SVM-based detectors

In this example, it is illustrated by Fig. 4.11 that SVM may be insensitive to the parameter: E_b/N_0. *The solid line is the performance under the SVM receiver whose support vectors are obtained with the assumption* $E_b/N_0 = 20$ *dB but are used for different* E_b/N_0*s. The dashed line, on the other hand, represents the performance of the SVM receiver using* E_b/N_0*-dependent (optimized) support vectors.*

The fact that both receivers yield almost identical performance indicates that the proposed SVM-based receiver is relatively robust against changing E_b/N_0. *Such robust*

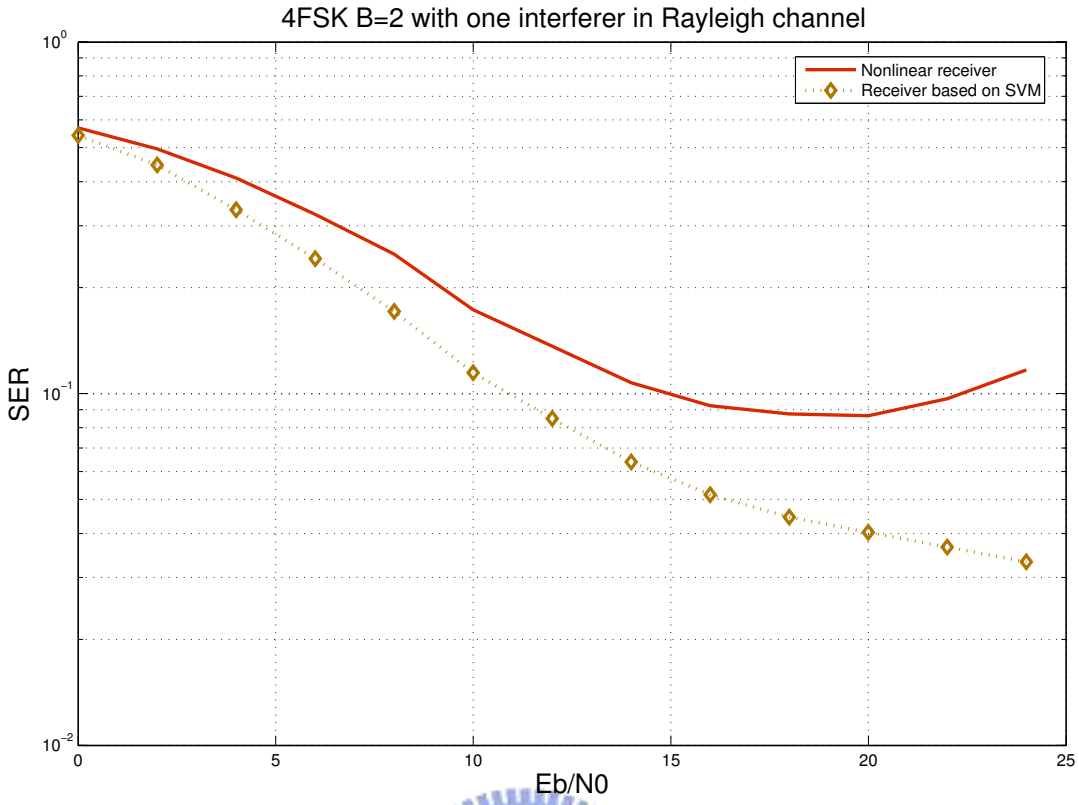


Figure 4.9: Performance comparison of the 4FSK receiver of [17] and SVM-based receiver in a Rayleigh fading channel with $B = 2$ and one interferer.

behavior also simplifies our receiver design for there is no need to store different sets of SVs for different $E_b/N_0 s$.

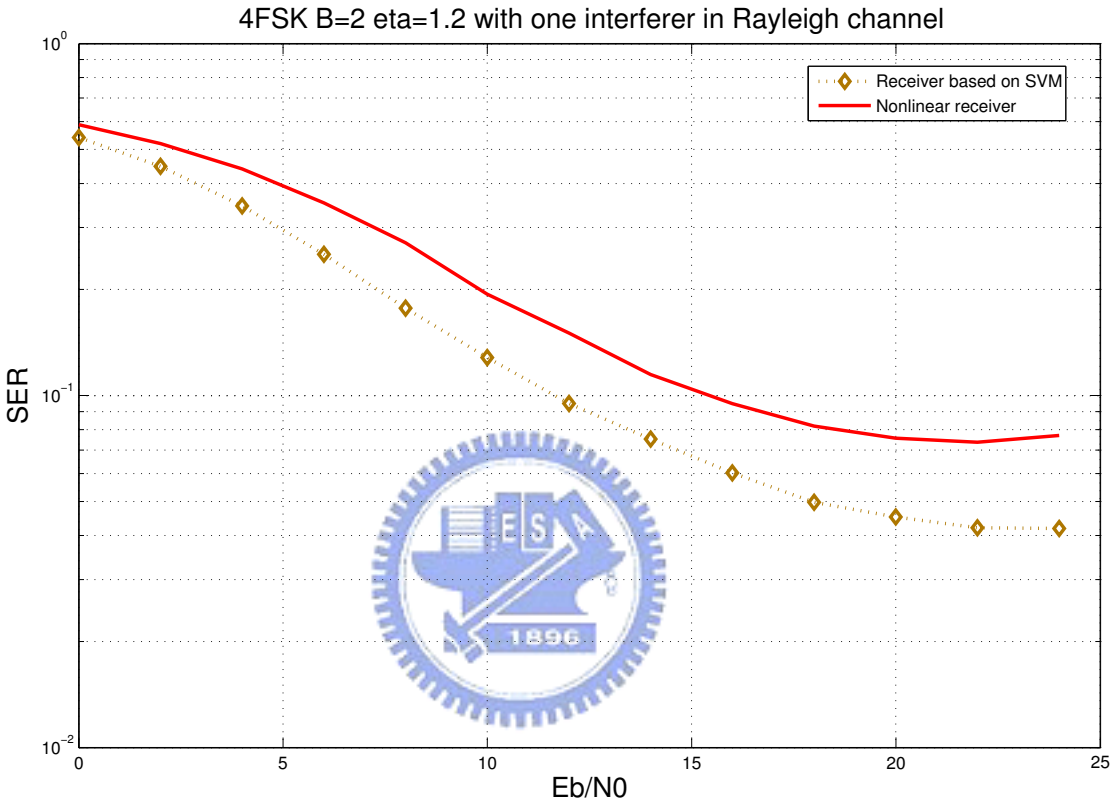


Figure 4.10: Performance comparison of the 4FSK receiver of [17] and SVM-based receiver in Rayleigh fading channels with $B = 2$, $\eta = 1.2$ and one interferer.

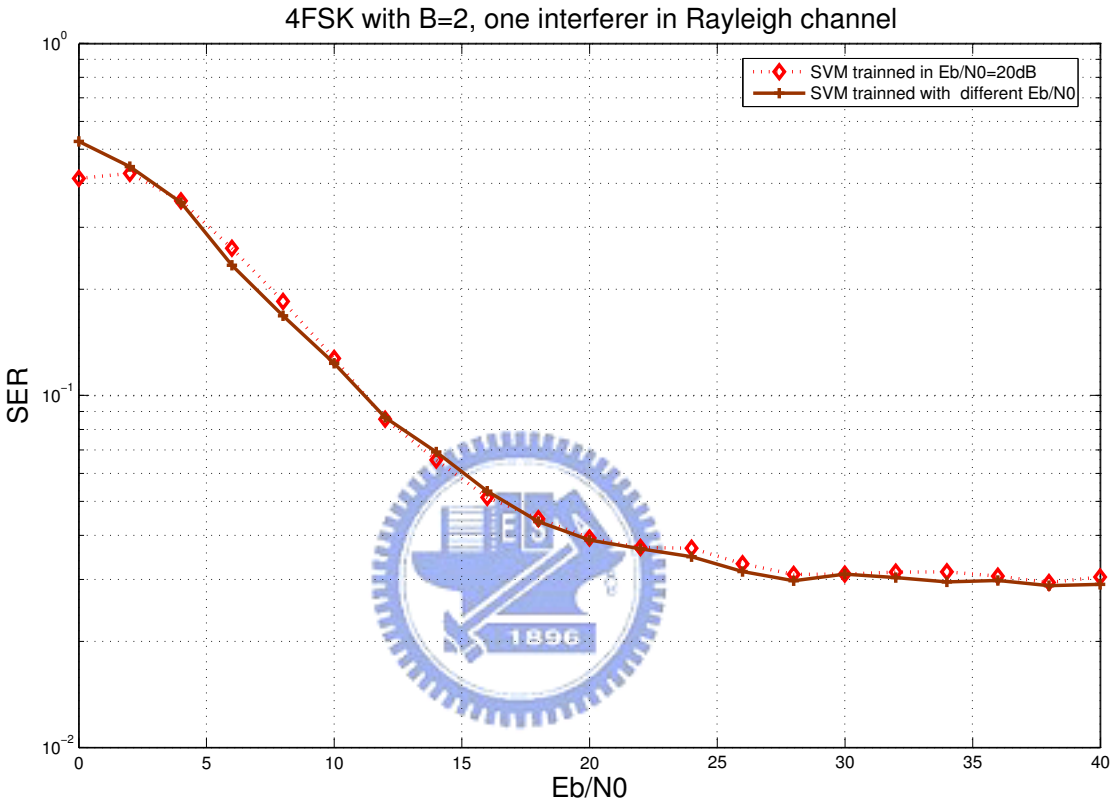


Figure 4.11: Performance comparison of SVM-based receivers with different training ways.

Chapter 5

Conclusion and Future Works

5.1 Conclusion

This thesis documents our investigation results on the design of receiver based on SVM for MFSK/FHMA system. From the simulation results, we may infer that the receiver based on SVM can approximate to the optimal receiver or outperform proposed receivers in chapter 2. The main reason is due to the fact that the conditional probability density is intractable. The SVM-based receiver, however, does not require the knowledge and usually the performances can approximate to the ML results. Also, the SVM-based receiver is insensitive to the parameter: E_b/N_0. With this property, we can save the memory and implement the receiver without sorting other results based on different E_b/N_0.

5.2 Future Works

There are several problems need to be addressed in this technique and we hope we can solve them in the future.

(1.) The channel state may not be static, i.e. the channel model may be time-varying. In this case, the SVM techniques proposed here may not be used directly due to the SMO algorithm and the QP problems.

(2.) When the size of M is large, the simulation will take large memories and time.

This is due to the SMO and DAG algorithm and it may be better to solve the QP problem once at all or to handle the training set in advance.

In practice, the size of M will be large. For instance, in [9], it set $M = 256$. Also, when we apply this technique to Flash-OFDM, this problem will appear again.

(3.) In this thesis, we do not indicate the optimal kernel and the choice of parameter given kernel. We believe that the problem is ongoing research for the moment.

(4.) To combine the SVM-based receiver with error correction code will appear some problems. One of them is the size of M, which is discussed in (2). For instant, the size of M may be 204 for RS$(204, 188)$. Another is the **soft input** for the error correcting decoder. In our thesis, the output of the receiver is one of $\{1, 2, 3, \ldots M\}$. It could not be used to be the soft input of the decoder directly. In this case, it may rely on the notion of regression, which is out of scope here.

Appendix A

The Objective Function for SMO Algorithm

In the section, we derive the (3.9). By SMO algorithm, one can break a large QP problem into a series of smallest possible QP problems, and the smallest possible QP problem can be solved analytically, which avoids using a time-consuming numerical QP optimization as an inner loop. Also, the matrix computation is avoided, so the amount of memory required for SMO is reduced. Hence, one can solve a large QP problem efficiently.

In this algorithm, since the $\eta = 0$ is possible, one need some other steps to make the algorithm work. In those steps, one need to know the equation (3.9). Hence we derive the equation for completing the SMO algorithm here.

We begin these deductions with equation (3.3).

$$\begin{aligned} J_D &= \frac{1}{2}K_{11}(\kappa - s\alpha_2)^2 + \frac{1}{2}K_{22}\alpha_2^2 + sK_{12}(\kappa - s\alpha_2)\alpha_2 + y_1(\kappa - s\alpha_2)v_1 \\ &\quad + y_2\alpha_2 v_2 - (\kappa - s\alpha_2) - \alpha_2 + \text{residual terms} \\ &= \frac{1}{2}(K_{11} + K_{22} - 2K_{12})\alpha_2^2 + (-K_{11}\kappa s + sK_{12}\kappa - y_1 s v_1 + y_2 v_2 + s - 1)\alpha_2 \\ &\quad + \frac{1}{2}K_{11}\kappa^2 + y_1\kappa v_1 - \kappa + \text{residual terms} \end{aligned} \tag{3.1}$$

where $v_i = \sum_{j=3}^{N} y_j \alpha_j^* K_{ij} = u_i - b^* - y_1\alpha_1^* K_{1i} - y_2\alpha_2^* K_{2i}$

Consider $y_1 s v_1$ and $y_2 v_2$ separably,

$$\begin{aligned} y_1 s v_1 &= y_1 s(u_1 - b^* - y_1\alpha_1^* K_{11} - y_2\alpha_2^* K_{21}) \\ &= y_1 s u_1 - y_1^2 s\alpha_1^* K_{11} - s^2\alpha_2^* K_{21} - y_1 s b^* \\ &= y_2 u_1 - (s\kappa - \alpha_2^*)K_{11} - \alpha_2^* K_{21} - y_2 b^* \\ y_2 v_2 &= y_2(u_2 - b^* - y_1\alpha_1^* K_{12} - y_2\alpha_2^* K_{22}) \\ &= y_2 u_2 - s\alpha_1^* K_{12} - \alpha_2^* K_{22} - y_2 b^* \\ &= y_2 u_2 - (s\kappa - \alpha_2^*)K_{12} - \alpha_2^* K_{22} - y_2 b^* \end{aligned}$$

where

$$\alpha_1^* + s\alpha_2^* = \kappa \Longrightarrow s\alpha_1^* + \alpha_2^* = s\kappa \tag{3.2}$$

and

$$y_1 s = y_1^2 y_2 = y_2 \tag{3.3}$$

Then consider

$$\begin{aligned} -y_1 s v_1 + y_2 v_2 &= y_2(u_2 - u_1) - \alpha_2^*(K_{11} + K_{22} - 2K_{12}) + s\kappa K_{11} - s\kappa K_{12} \\ &= y_2(u_2 - u_1) - \alpha_2^*\eta + s\kappa K_{11} - s\kappa K_{12} \end{aligned}$$

Therefore,

$$\begin{aligned} -K_{11}\kappa s + sK_{12}\kappa - y_1 s v_1 + y_2 v_2 + s - 1 &= y_2(u_2 - u_1) - \alpha_2^*\eta + s - 1 && (3.4) \\ &= y_2(u_2 - u_1 - y_2 + y_1) - \alpha_2^*\eta && (3.5) \\ &= y_2(E_2 - E_1) - \alpha_2^*\eta && (3.6) \end{aligned}$$

Finally, substitute (3.4) into (3.1)

$$J_D = \frac{1}{2}\eta\alpha_2^2 + (y_2(E_2 - E_1) - \alpha_2^*\eta)\alpha_2 + \text{Const.}$$

where Cons.=residual terms$+\frac{1}{2}K_{11}\kappa^2 + y_1\kappa v_1 - \kappa$.

Bibliography

[1] C. J. C. Burges, "A tutorial on support vector machines for pattern recognition," *Knowledge Discovery and Data Mining*, vol. 2, pp. 121 - 167, 1998.

[2] S. Chen, A. K. Samingan, L. Hanzo, "Support vector machine multiuser receiver for DS-CDMA signals in multipath channels," *IEEE Trans. Neural Networks,* vol. 12, no. 3, pp. 604 - 611, May. 2001.

[3] W. Chu and C. J. Colbourn, "Optimal frequency hopping sequences via cyclotomy," *IEEE Trans. Inform. Theory*, pp. 1139-1141, Mar. 2005.

[4] G. R. Cooper, R. W. Nettleton, "A spread spectrum technique for high capacity mobile communication," *IEEE Trans. Veh. Technol.*, vol. VT-27, pp. 264 - 275, NOV. 1978.

[5] N. Cristianini, J. Shawe-Taylor *An Introduction to Support Vector Machines*, Cambridge University Press.

[6] G. Einarsson, "Address assignment for a time-frequency-coded, spread-spectrum system," *Bell Syst. Tech. J.*, vol. 59, no.7, pp. 1241 - 1255, Sep. 1980.

[7] R. Fuji-Hara, Y. Miao and M. Mishima, "Optimal frequency hopping sequences: a combinatorial approach," *IEEE Trans. Inform. Theory*, pp. 2408-2420, Oct. 2004.

[8] X. Gong, A. Kuh, "Support vector machine for multiuser detection in CDMA communications," *33rd Asilomar Conference on Signals, Systems and Computers*, Monetary CA, vol. 1, pp.685 - 689, Nov. 1999.

[9] D. J. Goodman, P. S. Henry, and V. K. Prabhu, "Frequency-hopped multilevel FSK for mobile radio," *Bell Syst. Tech. J.*, vol. 59, no.7, pp. 1257 - 1275, Sep. 1980

[10] S. S. Keerthi, S. K. Shevade, C. Bhattacharyya, and K. R. K. Murthy, "Improvements to Platt's SMO algorithm for SVM classifier design," *Neural Computation*, vol.13 , issue.3, pp. 637 - 649, Mar. 2001

[11] G.S. Kimeldorf and G. Wahba, "Some results in tchebycheffian spline functions," *J. Math. Anal. Applicat.*, vol. 33, no. 1, pp. 82V95, 1971.

[12] J. C. Platt, N. Cristianini, J. Shawe-Taylor, "Large Margin DAGs for Multiclass Classification," in Proceedings of Neural Information Processing Systems, NIPS'99, pp. 547-553. MIT Press, 2000

[13] J. C. Platt, "Sequential Minimal Optimization: A Fast Algorithm for Training Support Vector Machines," in Microsoft Research, Technical Report MSR-TR-98-14, Apr. 1998

[14] Schwartz, Bennett, and Stein *Communication Systems and Techniques*, McGraw-Hill, 1966.

[15] G. Solomon, "Optimal frequency hopping sequences for multiple access," in *Proc. 1973 Symp. Spread Spectrum Commun.*, San Diego, CA, Mar13 - 16, 1973, pp. 33-35.

[16] John G. Proakis, *Digital Communications*, 4rd Edition, McGraw-Hill, 2001.

[17] Y. Su, Y. Shen, and C. Hsiao, "On the detection of a class of fast frequency-hopped multiple access signals," *IEEE J. Select. Areas Commun.*, vol. 19, no. 11, pp. 2151 - 2164, Nov.2001.

[18] B. Schölkopf, A. Smola *Learning with Kernel*, MIT Press, Cambridge, MA, 2002.

[19] J.A.K. Suykens, T. Van Gestel, J. De Brabanter, B. De Moor, J. Vandewalle, *Least Squares Support Vector Machines*, World Scientific, Singapore, 2002

[20] U. Timor, "Improved decoding scheme for frequency-hopped multilevel FSK system," *Bell Syst. Tech. J.*, vol. 59, no. 10, pp. 1839 - 1855, Dec. 1980.

[21] U. Timor, "Multistage decoding of frequency-hopped FSK system," *Bell Syst. Tech. J.*, vol. 60, no. 4, pp. 471 - 483, Apr. 1981.

[22] A. J. Viterbi, "A processing satellite transponder for multiple access by low rate mobile users," *Proc. Digital Satellite Commun. Conf.* (Montreal, P.Q., Canada), pp. 166 - 174, Oct. 1978.

簡　　歷：

姓　　名：劉人仰

居　住　地：台北市

研 究 方 向：Machine Learning

學　經　歷：

中山大學電機工程學系　　(1999～2003)

交通大學電信工程碩士班　　(2003～2005)

Graduate Course：

Information Theorem
Adaptive Signal processing
Detection and Estimation Theory
Digital Signal Processing
Computer Network
Algorithm
Matrix Computation
Statistic
Real Analysis (I)