Discrete-Time Crossing-Point Estimation for Switching Power Converters

Graeme Smecher

Department of Electrical & Computer Engineering
McGill University
Montreal, Canada

October 2008

A thesis submitted to McGill University in partial fulfillment of the requirements for the degree of Master of Engineering.

Your file Votre référence
ISBN: 978-0-494-66928-0
Our file Notre référence
ISBN: 978-0-494-66928-0

Abstract

In a number of electrical engineering problems, so-called "crossing points" – the instants at which two continuous-time signals cross each other – are of interest. Often, particularly in applications using a Digital Signal Processor (DSP), only periodic samples along with a partial statistical characterization of the signals are available. In this situation, we are faced with the following problem: Given limited information about these signals, how can we efficiently and accurately estimate their crossing points?

For example, an audio amplifier typically receives its input from a digital source decoded into regular samples (e.g. from MP3, DVD, or CD audio), or obtained from a continuous-time signal using an analog-to-digital converter (ADC). In a switching amplifier based on Pulse-Width Modulation (PWM) or Click Modulation (CM), a signal derived from the sampled audio is compared against a deterministic reference waveform; the crossing points of these signals control a switching power stage. Crossing-point estimates must be accurate in order to preserve audio quality. They must also be simple to calculate, in order to minimize processing requirements and delays.

We consider estimating the crossing points of a known function and a Gaussian random process, given uniformly-spaced, noisy samples of the random process for which the second-order statistics are assumed to be known. We derive the Maximum A-Posteriori (MAP) estimator, along with a Minimum Mean-Squared Error (MMSE) estimator which we show to be a computationally efficient approximation to the MAP estimator.

We also derive the Cramér-Rao bound (CRB) on estimator variance for the problem, which allows practical estimators to be evaluated against a best-case performance limit. We investigate several comparison estimators chosen from the literature. The structure of the MMSE estimator and comparison estimators is shown to be very similar, making the difference in computational expense between each technique largely dependent on the cost of evaluating various (generally non-linear) functions.

Simulations for both Pulse-Width and Click Modulation scenarios show the MMSE estimator performs very near to the Cramér-Rao bound and outperforms the alternative estimators selected from the literature.

Sommaire

Dans bon nombre de problèmes d'ingénierie électrique, il est intéressant de noter les points dits points de croisement, soit l'instant où deux signaux continus se croisent. Souvent, surtout dans les applications utilisant un processeur de signal numérique (PSN), seuls des échantillons périodiques avec une caractérisation statistique partielle des signaux sont disponibles. Dans une telle situation, nous sommes face au problème suivant: vu l'information limitée sur ces signaux, comment peut-on estimer de façon efficace et exacte leurs points de croisement?

Par exemple, dans un amplificateur audio avec modulation d'impulsions en durée (MID) ou en position (Click Modulation, ou CM), un signal inconnu est comparé à une onde de référence déterministe; les points de croisement de ces signaux contrôlent un étage de puissance commutatif. L'entrée d'un tel amplificateur est habituellement décodée en échantillons réguliers (par exemple, l'audio d'un MP3, d'un DVD ou d'un CD) ou encore obtenue à partir d'un signal continu utilisant un convertisseur analogique-numérique (CAN). Les estimations de points de croisement doivent être précises afin de conserver la qualité audio. Elles doivent aussi être simples à calculer afin de minimiser les exigences et les délais du traitement.

Nous voulons estimer les points de croisement d'une fonction connue et d'un processier aléatoire gaussien, grâce à des échantillons, également espacés et brouillés par le bruit, du processier aléatoire pour lesquels les statistiques du second ordre sont présumées être connues. Nous dérivons l'estimateur du maximum a posteriori (MAP), ainsi qu'un estimateur de type Erreur Quadratique Moyenne Minimale (EQMM); nous montrons que ce dernier est une approximation numérique efficace de l'estimateur MAP.

Nous dérivons également la Borne de Cramér-Rao (BC) de la variance de l'estimateur pour le problème, ce qui permet l'évaluation d'estimateurs pratiques par rapport à une limite de performance dans la meilleure situation. Nous étudions plusieurs estimateurs de comparaison choisis parmi la littérature. Nous prouvons que la structure de l'estimateur EQMM et celle des estimateurs de comparaison sont très semblables, rendant la différence d'utilisation de ressources informatiques entre chacune des techniques hautement dépendante du coût relié à l'évaluation de diverses fonctions (habituellement non linéaires).

Les simulations à la fois des scénarios avec modulation d'impulsions en durée et avec modulation d'impulsions en position (CM) montrent que l'estimateur EQMM s'approche

très près de la CRB et est plus efficace que les autres estimateurs choisis dans la littérature.

Acknowledgments

I am thankful for the assistance and support provided by my supervisor, Dr. Benoît Champagne. I also grateful for insights provided by Dr. Harry Leib (McGill University), and correspondance from Dr. James Cavers (Simon Fraser University.)

Contents

List of Figures

List of Tables

List of Algorithms

List of Acronyms

CD	Compact Disc
ADC	Analog-to-Digital Converter
CM	Click Modulation
DEPWM	Double-Edge Pulse-width Modulation (i.e. using a triangular carrier)
DSP	Digital Signal Processing (or Processor)
DVD	Digital Video (or Versatile) Disc
EMI	Electromagnetic Interference
FET	Field Effect Transistor
LEPWM	Leading-Edge Pulse-width Modulation (i.e. using a sawtooth carrier)
MP3	MPEG-1 Audio Layer 3
MPEG	Moving Picture Experts Group
NSPWM	Naturally-Sampled Pulse-width Modulation
PWM	Pulse-width Modulation
SNR	Signal-to-Noise Ratio
TEPWM	Trailing-Edge Pulse-width Modulation (i.e. using a sawtooth carrier)
USPWM	Uniformly-Sampled Pulse-width Modulation
WSS	Wide-Sense Stationary
ZOH	Zero-Order Hold, or sample-and-hold

Chapter 1

Introduction

1.1 The Crossing-Point Estimation Problem

In many signal processing applications, the samples of an unknown, continuous signal are used to estimate the times at which this signal crosses a known function. The known function depends on the application, and is typically zero, a fixed level, or a periodic carrier. Often, only noisy samples of the unknown signal are available, along with some information about their statistical properties. This thesis concerns itself with three questions:

- Is there an optimal method to estimate the crossing points of the unknown signal and the known function?
- If so, is there a computationally efficient approximation to the optimal method?
- How accurate can the resulting estimates be?

This crossing-point estimation problem arises in a number of applications. For example, zero-crossings are of interest in pitch detection for speech signals [1], demodulation of FM and FSK signals [2], and frequency estimation for power-quality monitoring [3]. Level-crossings have been extensively studied [4], and are of practical interest in non-linear sampling applications [5]. Carrier crossings have also received substantial theoretical treatment [6], and are of practical interest in Pulse-Width Modulation (PWM) [7, 8], Click Modulation (CM) [9], and non-linear sampling applications [10].

The examples of PWM and CM are of particular interest due to the growing demand for high-fidelity, inexpensive switching amplifiers. In these applications, crossing points of a

bandlimited signal and a known carrier (which may be sawtooth, triangular, or sinusoidal) must be accurately determined in order to preserve the fidelity of the input signal (e.g. 16 bits at a sampling rate of $f_s = 44.1$ kHz, in typical audio applications.) In consumer-grade equipment, this process must also be computationally efficient to avoid costs associated with an expensive Digital Signal Processor (DSP) or a high gate count in custom silicon.

In this thesis, we investigate the crossing-point estimation problem from the perspective of statistical signal processing. Although we approach the problem in a general way, we focus on the particular applications of PWM and CM due to the rapid progress and continuing interest in switching amplifier research and design.

1.2 Previous Work

Several investigations of the discrete-time crossing-point estimation problem have been conducted in order to design digitally controlled switching amplifiers using PWM. In all such cases, estimating crossing points was cast as a two-step process: crossing points were first rapidly and coarsely located by observing points at which samples of the random signal and carrier cross each other. Then, nearby samples of the random signal and carrier were used to generate a more accurate estimate of the crossing point.

The most common approach to refining crossing-point estimates is to approximate the random signal between samples using a continuous interpolating function. It then remains to estimate the crossing points of the interpolating and carrier functions, for which many root-finding techniques (e.g. Brent's algorithm or Newton's method) are applicable.

The simplest choice for the interpolator is also the most frequently used: the unique polynomial of order $M-1$ passing through M sample points of the random signal. For example, when $M = 2$, the polynomial interpolator corresponds to straight-line interpolation between samples on either side of the crossing-point. In [11], straight-line interpolation is abandoned in favour of higher-order polynomial interpolation. However, straight-line interpolation is explored for a different parameter set and found to perform adequately in [12]. Cubic and higher-order polynomial interpolators are explored in [8, 13, 14, 15, 16, 7]. These references, all of which focused on PWM audio amplifiers, suggest the estimation accuracy of low-order polynomial interpolators varies strongly with design conditions.

An alternative scheme [14, 7] begins with an infinite-sum expression for the crossing points assuming the unknown input is an analytic, nonrandom signal and given knowledge

of all of its derivatives. This expression is truncated to an appropriate number of terms, and the derivatives of the random signal are estimated using numerical techniques.

Several surveys [15, 13, 17, 18, 19, 7] describe, evaluate and compare the above techniques. The impact of crossing-point estimation errors in PWM are considered in [20].

None of the approaches mentioned above take advantage of statistical knowledge of the underlying signals. It is unclear if and when these techniques are optimal in any sense, as they are designed on an ad-hoc basis. While they can perform adequately in an oversampled regime when the random signal is low-pass in nature, these techniques do not generalize to arbitrary signal models, and may not be as computationally efficient as alternatives.

1.3 Thesis Contributions

This thesis approaches the discrete-time crossing-point estimation problem from a statistical perspective under the Maximum A-Posteriori (MAP) framework. By adopting a statistical framework, it is possible to inject and exploit information about the random signal, such as its autocorrelation function, which would not be feasible with a nonstatistical approach. It is also possible to model and compensate for measurement noise.

We derive the MAP estimator for the problem, a methodological process yielding an estimate that meets certain optimality criteria. We also derive the Cramér-Rao bound (CRB) on estimator variance for the problem, which provides an absolute metric against which estimators may be compared. We introduce an alternative crossing-point estimator using the Minimum Mean-Squared Error (MMSE) estimator for the random signal, and show it to be an approximation to the MAP estimator under oversampled, high-SNR conditions. We consider the computational complexity of these and several estimators chosen from the literature.

Finally, we present simulated results using both sinusoidal (for CM) and sawtooth (for PWM) carrier crossings as a scenario. We compare the performance of the MAP and MMSE estimators with the Cramér-Rao bound and a number of estimators chosen from the literature. The MAP and MMSE estimators approach the CRB and outperform the alternatives simulated.

1.4 Organization

The remainder of this work is organized into five chapters.

In Chapter 2, we present an introduction to switching amplifiers, focusing on the techniques of Pulse-Width Modulation and Click Modulation. This material motivates exploration of the crossing-point estimation problem and provides some context for the simulations presented later.

In Chapter 3, we formally define the discrete-time crossing-point estimation problem. We also introduce several constraints on the general problem introduced by Pulse-Width and Click Modulation. These constraints allow a number of simplifications to be used in the derivations that follow.

In Chapter 4, we derive both MAP and MMSE estimators, as well as the Cramér-Rao bound on estimator variance. We also formally develop some of the previously published solutions to the crossing-point estimation problem in order to compare them with the estimators proposed in this thesis. We consider the computational cost of these techniques.

In Chapter 5, we present some experimental results. These results are gathered from simulations focusing on PWM and CM as representative test scenarios.

Finally, we conclude with Chapter 6. Mathematical proofs are provided in the appendices for longer derivations.

Chapter 2

Switching Amplification

2.1 A Basic Switching Amplifier

Consider an amplifier with gain K, which accepts the signal $a(t)$ and produces $K\hat{a}(t)$. Generally speaking, it is desirable that $\hat{a}(t)$ approximates $a(t)$ as closely as possible; any differences imply distortion.

A lack of distortion is not the only consideration facing amplifier designers. Particularly in consumer electronics, two additional factors – efficiency and cost – are of growing importance. A more efficient amplifier can substantially improve a device's battery life, size, weight, and reliability. These improvements, in turn, allow reductions in the cost of a device, particularly as consumer trends demand smaller, more integrated, highly mobile products.

The central difficulty in building an efficient amplifier surrounds the drivers, the transistors that deliver bulk power to the load. While these can be coaxed into doing an excellent job of reproducing signals with high fidelity, they generally do so with very poor efficiency — in a traditional, class-AB audio amplifier, typical performance estimates are between 50% and 72.5% [21]. Switching amplifiers confront this efficiency deficit by transforming the audio signal into a form that can be efficiently amplified, but from which the undistorted audio can easily be recovered.

Switching amplifier designs may be represented using the basic structure of Figure 2.1. This block accepts an input signal $a(t)$, and outputs the signal $\hat{a}(t)$ amplified by the gain K. Ideally, this amplifier reproduces $\hat{a}(t) = a(t)$. However, within the amplifier, the input $a(t)$ is transformed into a highly distorted signal. As we will see, switching amplifiers

take advantage of the fact that some types of distortion can be permissible and, in fact, advantageous.

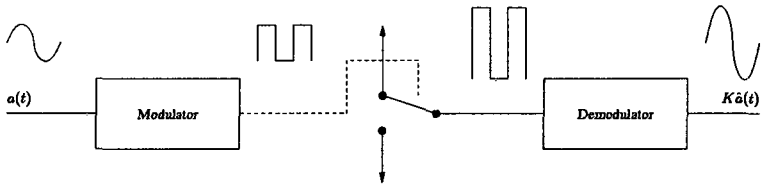


Fig. 2.1 A Basic Switching Amplifier

The input signal $a(t)$ is converted into a two-level control signal by the modulator. In general, this modulator encodes the input signal into the pulse edges (or, equivalently, pulse centers and widths) of the control signal.

Next, the output from the modulator is used to control a switch. This switch, typically built out of Field Effect Transistors (FETs), converts the low-power control input into a high-power signal capable of driving a load (e.g. a loudspeaker.) The control signal occupies a small number of states; although multi-level switching converters exist, it suffices here to consider only two-level designs.

The output from the switch passes through the demodulator before reaching the load. Because the switch and demodulator both process the amplifier's full output power, each must be efficiently designed. Inexpensive FET power stages can switch states very rapidly and can be highly efficient. However, to minimize the amplifier's energy dissipation, expense, bulk, and sensitivity to component and operating condition variations, the demodulator is generally restricted to a passive, low-order, low-pass filter.

By restricting the demodulator to a low-order, low-pass filter, the role of the modulator is essentially fixed. This role is discussed in the following section.

2.1.1 The Ideal Switching Modulator

The modulator in Figure 2.1 must convert a continuous-time signal to a switching representation by injecting high-frequency components which may either be neglected or removed by the demodulator. In other words, the modulator's output may be decomposed into disjoint low-pass and high-pass regions. The low-pass region reproduces the input signal $a(t)$ without distortion. Due to the demodulator filter, the high-pass region should not leave

the switching amplifier and may be used arbitrarily by the modulator to shape the input signal into a switching form. This role is illustrated in Figure 2.2, where F_a represents the bandlimit of $a(t)$. The Fourier transform of $a(t)$ (which we presume to exist) is denoted $A(f)$.

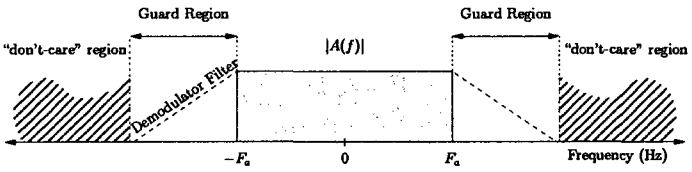


Fig. 2.2 Spectral Characteristics of the Ideal Switching Modulator (Magnitude $|A(f)|$ versus frequency f in Hz)

Although the primary task of the modulator is to generate a switching signal that faithfully reproduces the input signal in a low-pass region, there are a few additional traits a well-designed modulator should exhibit. For instance, the FETs comprising the power switch dissipate substantial amounts of energy during switching, reducing efficiency. In addition, abruptly switching a load with inductive characteristics (such as a loudspeaker) can stress components and introduce electromagnetic interference (EMI). Finally, switching cannot occur instantaneously (as the designer typically assumes), since the switch produces a voltage signal which is neccessarily continuous; this non-ideality also introduces distortion. For these and other reasons, the modulator should be designed with as low an average switching rate as is practical. In a digitally-controlled switching amplifier, the modulator should also be computationally inexpensive, since computational requirements can directly determine the cost of the modulator.

In the following sections we explore Pulse-Width Modulation and Click Modulation, two of the modulation schemes applicable to switching amplifier designs. Although this coverage is not exhaustive (we neglect, for example, delta-sigma modulation), our purpose is to give a sense of the structure of these modulators and the role of a crossing-point estimator within each of them.

2.2 Pulse-Width Modulation

The dominant modulation scheme used in switching amplifier designs is Pulse-Width Modulation (PWM).[1] PWM is a well-known method, used for many decades [22] in (for example) motor control and switching power supplies.

In this section, we present a brief description of several variants of PWM, in order to provide a sense of how PWM signals are generated in continuous-time systems. We also explore the spectral properties of a PWM signal, and discuss how these properties effect the ability of a PWM-based switching amplifier to attain the ideal modulator spectrum of Figure 2.2. For clarity, we present PWM in a continuous-time environment; additional effort is required to build a PWM system in discrete time.

2.2.1 Variants of PWM

Pulse-Width Modulation schemes are generally classified in two ways. The first distinction is the type of sampling: natural sampling (NS) or uniform sampling (US). The second distinction is the form of the carrier signal, which yields Trailing Edge (TE), Leading Edge (LE), or Double-Edge (DE) PWM. These classifications are illustrated with the schematic of a simple pulse-width modulator in Figure 2.3. The block labelled "ZOH" corresponds to a Zero-Order Hold or sample-and-hold device with sample period $T_s = 1/f_s$ [23].

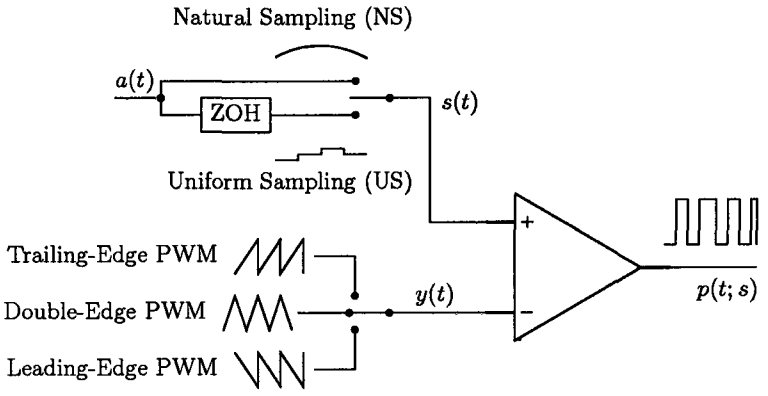


Fig. 2.3 Flavours of PWM

[1]The PWM scheme described here might more properly be called Intersective PWM or IPWM, to distinguish it from other schemes (such as Click Modulation) that also generate a pulse train with varying pulse widths. At the risk of confusion, we use the more conventional acronym PWM.

The input to this pulse-width modulator is the arbitrary (but generally bounded in amplitude) signal $a(t)$. Depending on the sampling method (NS or US) and carrier $y(t)$, the output $p(t;s)$ corresponds to a particular PWM scheme. The carrier, with period $T_s = 1/f_s$, adds a time-varying bias to the input signal. The comparator removes all information from the biased input except for its sign.

PWM Sampling Schemes

The type of sampling – NSPWM or USPWM – has important repercussions in a digital pulse-width modulator. USPWM generates crossing points which are trivial to calculate, but which exhibit non-linear distortion of the input signal in the baseband region (see also Figure 2.5.) Non-linear distortion in the baseband is a sufficiently negative feature that most PWM schemes disregard USPWM except as a tool for the development of improved schemes more closely mimicking Naturally-Sampled PWM [11, 24].

In the following section, we explore the variants of NSPWM illustrated in Figure 2.3.

PWM Carriers

Variants of NSPWM have the general structure shown in Figure 2.4. (We use the notation introduced in [24].)

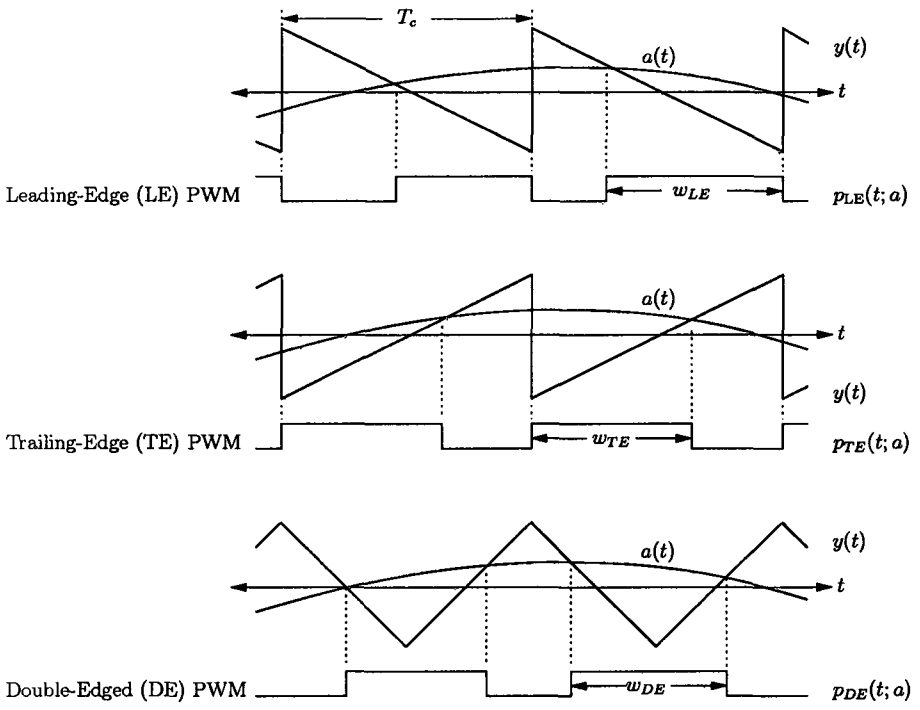


Fig. 2.4 Generating Naturally-Sampled TE, LE, and DE Pulse-Width Modulation

In all three PWM variants, the output signal $p(t;a)$ is defined by the crossing points of a bounded input signal $a(t)$ and a periodic carrier $y(t)$. The output waveform $p(t;a)$ exhibits pulses of width w that encode the desired signal $a(t)$. For TE and LE PWM, the carrier is a sawtooth, and only half of the edges in $p(t;a)$ are determined by the input $a(t)$. For DE PWM, the carrier signal is a triangle wave, and both edges of $p(t;a)$ are determined by $a(t)$. In all three cases, the carriers shown have the same fundamental frequency $f_c = 1/T_c$, and the input $a(t)$ is always bounded by extrema of $y(t)$.

2.2.2 Spectral Characteristics of NSPWM Signals

The spectral characteristics of PWM have been studied in many papers [22, 25, 26, 27, 24]. Closed-form expressions for the PWM output spectra generated by an arbitrary (deterministic) signal have only recently been published, first informally [27] and then more rigorously [24]. We review these results in order to explore the spectral characteristics of a PWM signal. Black's analysis [22] may be used to demonstrate the same characteris-

tics when $a(t)$ is constrained to be sinusoidal; we focus on the more recent approaches to promote generality.

Let the signals $p_{\mathrm{TE}}(t;a)$, $p_{\mathrm{LE}}(t;a)$, and $p_{\mathrm{DE}}(t;a)$ represent (respectively) the output of a trailing-edge, leading-edge, and double-edge Pulse-Width Modulator with input $a(t)$, given $|a(t)| \leq 1$. The Fourier transform of $a(t)$ (which we assume exists) is denoted $A(f)$, and that of $[a(t)]^n$ (for positive integer values of n) is denoted $A_n(f)$. The following results are taken from [24].

Trailing-Edge (TE) NSPWM

For trailing-edge PWM, we have:

$$p_{\mathrm{TE}}(t;a) = a(t) + \sum_{k=1}^{\infty} \frac{2}{k\pi} \left[\sin\left(2\pi k f_c t\right) - (-1)^k \sin\left(2\pi k f_c t - k\pi a(t)\right)\right] \tag{2.1}$$

The corresponding spectrum $P_{TE}(f;a)$ is given by the following:

$$\begin{aligned} P_{TE}(f;a) = A(f) &+ \sum_{k=1}^{\infty} \frac{1}{jk\pi} \left[\delta(f + 2\pi k f_c) - \delta(f - 2\pi k f_c)\right] \\ &+ \sum_{k=1}^{\infty} (-1)^k \sum_{n=1}^{\infty} \frac{(jk\pi)^{n-1}}{n!} \left[A_n(f + k f_c) + (-1)^{n-1} A_n(f - k f_c)\right] \end{aligned} \tag{2.2}$$

Leading-Edge (LE) NSPWM

For leading-edge PWM, we have:

$$p_{\mathrm{LE}}(t;a) = a(t) + \sum_{k=1}^{\infty} \frac{2}{k\pi} \left[(-1)^k \sin\left(2\pi k f_c t + k\pi a(t)\right) - \sin\left(2\pi k f_c t\right)\right] \tag{2.3}$$

The corresponding spectrum $P_{LE}(f;a)$ is given by the following:

$$\begin{aligned} P_{LE}(f;a) = A(f) &- \sum_{k=1}^{\infty} \frac{1}{jk\pi} \left[\delta(f + 2\pi k f_c) - \delta(f - 2\pi k f_c)\right] \\ &+ \sum_{k=1}^{\infty} (-1)^k \sum_{n=1}^{\infty} \frac{(jk\pi)^{n-1}}{n!} \left[(-1)^{n-1} A_n(f + k f_c) + A_n(f - k f_c)\right] \end{aligned} \tag{2.4}$$

Double-Edge (DE) NSPWM

For double-edge PWM, we have:

$$p_{\mathrm{DE}}(t;a) = a(t) + \sum_{k=1}^{\infty} \frac{2(-1)^k}{k\pi}\left[\sin\left(2\pi k f_c t + k\pi\frac{a(t)+1}{2}\right) - \sin\left(2\pi k f_c t - k\pi\frac{a(t)+1}{2}\right)\right] \tag{2.5}$$

The corresponding spectrum $P_{DE}(f;a)$ is given by the following:

$$P_{DE}(f;a) = A(f) + \sum_{k=1}^{\infty}(-1)^k \sum_{n=1}^{\infty}\left[\frac{(j2k\pi)^{2n-2}}{2^{2n-2}(2n-1)!}(S_{2n-1}(f+2kf_c) + S_{2n-1}(f-2kf_c)) - \frac{(j(2k-1)\pi)^{2n-1}}{j2^{2n-1}(2n)!}(S_{2n}(f+(2k-1)f_c) + S_{2n}(f-(2k-1)f_c))\right] \tag{2.6}$$

Note that (2.3) may be derived from (2.1) using the identity $p_{\mathrm{LE}}(t;a) = -p_{\mathrm{TE}}(t;-a)$. In addition, (2.5) may be derived from (2.1) and (2.3) directly by noting that

$$p_{\mathrm{DE}}(t;a) = p_{\mathrm{TE}}\left(t;\frac{1+a}{2}\right) + p_{\mathrm{LE}}\left(t;\frac{1+a}{2}\right) - 1. \tag{2.7}$$

The magnitude spectra of $P_{TE}(f;a)$, $P_{LE}(f;a)$, and $P_{DE}(f;a)$ are similar (except in the case of DEPWM, which does not have impulse components at multiples of f_c.) An illustration for a bandlimited audio signal $a(t)$ with a flat-top spectrum is shown in Figure 2.5.

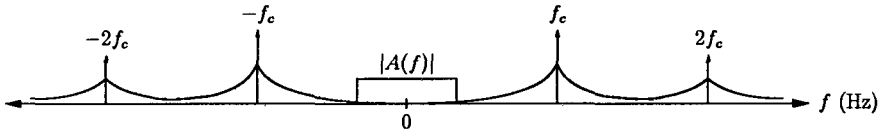


Fig. 2.5 Magnitude Spectrum of a TEPWM or LEPWM Signal

The spectrum shown in Figure 2.5 has three salient features:

- The flat component at the centre corresponds to the input signal,
- There is an infinite sequence of delta terms at multiples of the carrier frequency, and
- Each of these carrier terms is accompanied by a phase-modulated term that produces a "skirt" around it in the spectrum.

In the following section, we explore the consequences that the structure of all three PWM signals has on a PWM-based switching amplifier.

2.2.3 Disadvantages of NSPWM for Switching Amplification

The "skirts" around each carrier term in Figure 2.5 are generated by phase-modulation terms $A_n(f)$ in the corresponding PWM spectrum. In general, phase modulation of an arbitrary signal $a(t)$ generates a signal that is not bandlimited; thus, each phase-modulation term in (2.1), (2.3) and (2.5) has infinite spectral support in general, and some energy from each term in the infinite summation contributes to the baseband region in Figure 2.2.

In practice, the presence of unbounded phase-modulation terms in NSPWM motivates designers to increase the switching frequency f_c in order to decrease the amount of baseband distortion. Although this strategy cannot completely remove the baseband distortion, a suitably high switching frequency can render this type of distortion negligible compared to other sources.

An increase in switching frequency comes at a cost. A higher switching frequency generate more edges in the modulator output, which produces more distortion due to imperfect switching, dissipates more energy within the switching FETs, and radiates more electromagnetic interference (EMI) due to fast-slewing control signals and inductive kick-back when the load has inductive characteristics. Digitally-controlled switching amplifiers typically generate switching waveforms using a dedicated peripheral that is only able to place edges with a clock-dependent accuracy; a higher switching frequency produces more edges per unit time and correspondingly more instances in which the switching signal is distorted due to the finite clock rate and other limitations of this peripheral.

The spectral characteristics of PWM therefore introduce a set of design trade-offs. Switching frequencies cannot be too low, otherwise significant energy from phase-modulation terms will leak into the baseband region of Figure 2.2. Once inside the baseband region, energy from phase-modulation terms passes the demodulator filter in Figure 2.1. Once past the demodulator filter, this unwanted energy is dissipated by the load, where it reduces the efficiency and fidelity of the amplifier. On the other hand, overly high switching frequencies exacerbate other imperfections and inefficiencies in the switch and modulator.

In the following section, we introduce Click Modulation. This alternative to PWM schemes comes at a higher signal-processing cost, but avoids the fundamental trade-off afflicting PWM in switching amplifiers.

2.3 Click Modulation

In the previous section, we described the spectral characteristics of a PWM signal and showed that it only approximated the ideal modulator spectrum of Figure 2.2. In a PWM system operating on the input signal $a(t)$, baseband distortion resulted from the spectrally unbounded, phase-modulation terms accompanying each carrier harmonic.

Click Modulation (CM) [9] is an alternative to PWM that provides a distortion-free baseband, allowing a lower carrier frequency. In a CM system, we seek to bandlimit each of PWM's phase-modulation terms. We do so directly by forming the phase modulation $\phi(t)$ of $a(t)$, and low-pass filtering it, producing $\hat{\phi}(t)$. Under the right conditions, we can reverse the phase-modulation process on the filtered signal $\hat{\phi}(t)$ and obtain a signal $\hat{a}(t)$ which may be pulse-width modulated without generating baseband distortion.

Figure 2.6 shows the block diagram of a click modulation system [9]. This system first creates the pre-envelope $a^+(t) = a(t) + j\hat{a}(t)$ of its input $a(t)$. The signal $a^+(t)$ has no spectral content at frequencies $f < 0$, and encodes $a(t)$ in $\mathbb{R}\{a^+(t)\}$.[2] [28] The pre-envelope $a^+(t)$ is supplied to an analytic exponential modulator (AEM), which generates $\phi(t) = e^{-ia^+(t)}$. The signal $\phi(t)$, which is complex in general, encodes $a(t)$ uniquely in its phase $\arg\phi(t)$ if $a(t)$ is appropriately bounded. Then, $\phi(t)$ is low-pass filtered to form $\hat{\phi}(t)$. In a properly designed CM system, the phase $\arg\hat{\phi}(t)$ no longer reproduces $a(t)$ exactly, but they are spectrally identical over a low-pass region determined by the designer.

The remainder of the CM system is equivalent to explicitly forming $\arg\hat{\phi}(t)$ and passing it to a TEPWM system [9, 27]. Two ground-referenced comparators combine to form the switching signal $q(t)$. These comparators are replaced by crossing-point estimators in a discrete-time click modulator; however, there are two important considerations concerning crossing-point estimation in this system:

- Two comparators in Figure 2.6 suggest two crossing-point estimators in a discrete-time implementation. In practice, only one crossing-point estimator is required in this system because the crossing-points of $-\sin(ct + \theta)$ are known a-priori.

- The remaining comparator is ground-referenced, suggesting a zero-crossing problem. However, the signal $s(t)$ applied to the other terminal is the sum of an unknown portion and a deterministic (sinusoidal) portion; as simulations demonstrate (see

[2]The $\mathbb{R}\{\cdot\}$ operator produces the real part of the argument. Similarly, the $\mathbb{I}\{\cdot\}$ operator denotes the imaginary part of the argument.

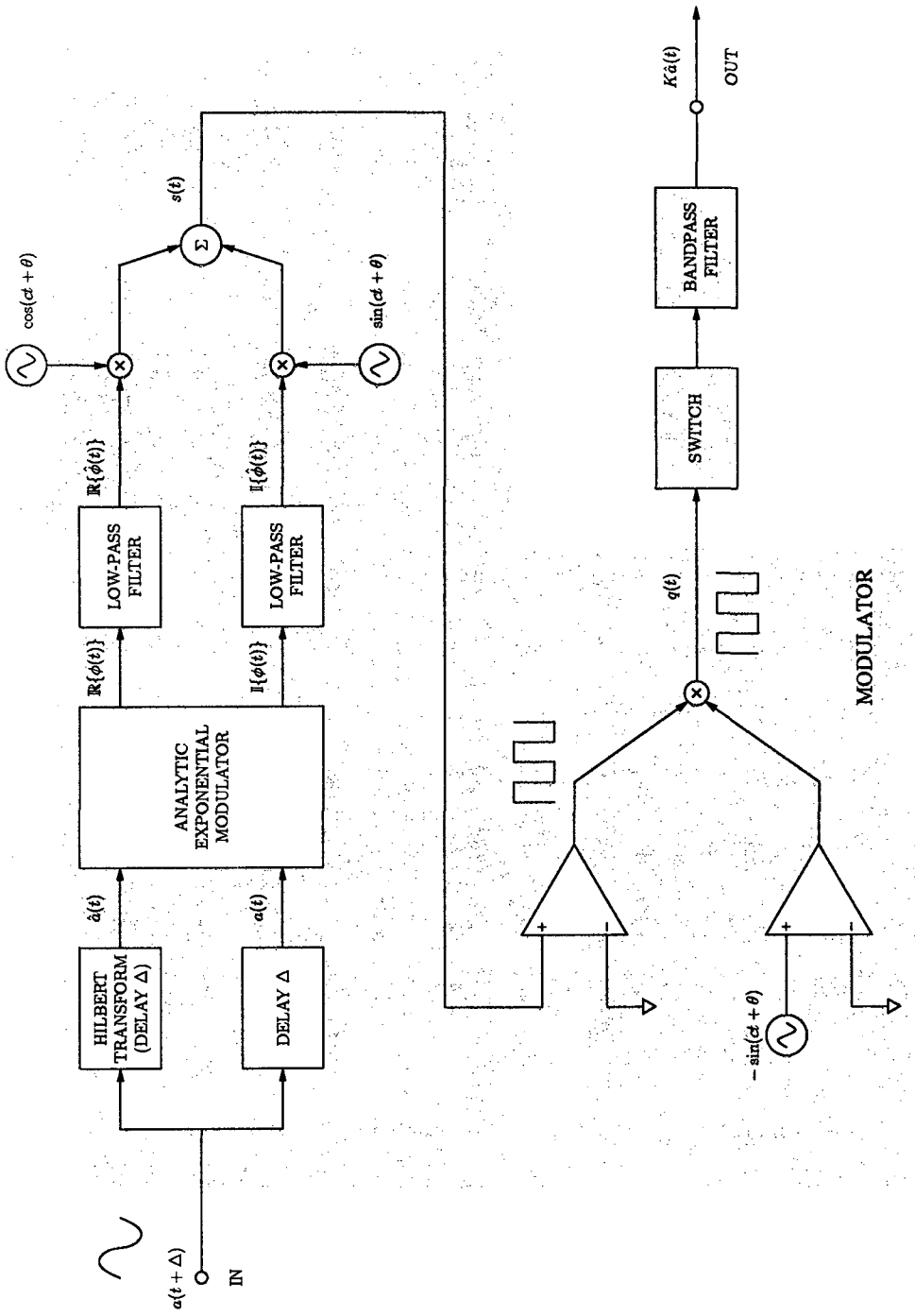


Fig. 2.6 Block Diagram of a Click Modulation System

Chapter 5), better performance may be obtained by decomposing the zero-crossing problem into a sinusoidal carrier-crossing problem.

The interpretation of CM as TEPWM with an additional predistortion stage is illustrated in Figure 2.7.

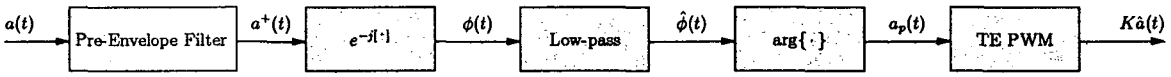


Fig. 2.7 Block Diagram of CM as a Pre-Distorted PWM Scheme

Once again, the pre-envelope $a^+(t)$ is formed, phase-modulated and filtered. In this formulation, the predistorted signal $\arg\hat{\phi}(t)$ is explicitly formed and pulse-width modulated. We note that this formulation is primarily useful for explanatory purposes; there are practical reasons not to construct a CM system in this manner. For example, although $\hat{\phi}(t)$ is low-pass in nature, $\arg\{\hat{\phi}(t)\}$ may be an extremely wideband signal, requiring a prohibitively high oversampling rate in a digital system.

For the complete design details of a CM system, interested readers are referred to [9].

2.3.1 Design Considerations for CM in Switching Amplifiers

As described in [9], CM requires filters which are extremely difficult to realize. In Figure 2.6, the input audio signal is processed by a Hilbert transformer. In the alternate formulation illustrated in Figure 2.7, an analytic filter removes negative-frequency components of the input audio. An audio signal may have meaningful content as low as 20 Hz; in either formulation, the resulting filter has a transition region as narrow as 20 or 40 Hz – an extremely impractical requirement, particularly when ripple and stopband attenuation requirements are stringent. This limitation has been noted [29, 30], although an alternative scheme is neither completely nor rigorously described.

The remaining design challenges in a CM system are surmountable: other filters within the system may be designed with reasonable tolerances using ordinary DSP techniques. The nonlinearities involved are characterized by rapidly decaying harmonics, suggesting that a low oversampling rate may not introduce too much distortion due to aliasing. The comparators in Figure 2.6 are precisely the crossing-point estimation problem considered in this thesis. However, the Hilbert filter at the beginning of a CM system cannot be

economically realized, and it is not obvious from published research how to sidestep this issue.

2.3.2 Required Developments for a Practical CM System

The "missing link" in a CM system is clearly a method to avoid extremely narrow transition regions in the Hilbert filter of Figure 2.6. Another interesting avenue of research into a CM system would be the introduction of a double-sided CM scheme; as the published CM system is comparable to TE PWM, a double-edge equivalent could exhibit reduced carrier power (relative to the power of the desirable audio signal) and improved rejection of some distortion due to imperfections in the switching waveform.

Chapter 3

Problem Formulation

The preceding chapters described the discrete-time crossing-point estimation problem informally and provided some motivation and background for further investigation. In this chapter, we present a formal definition of the problem.

First, we describe the problem in its general form. We then describe a simplified approach that is commonly adopted in practice, and which we will investigate in the sequel.

Following the formulation of the general crossing-point estimation problem, we focus on the particular applications of PWM and CM. Properly-functioning PWM and CM systems place additional constraints on the signals involved; these constraints simplify the developments in the following chapter.

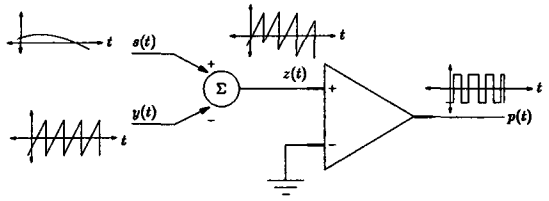
3.1 Discrete-Time Crossing-Point Estimation

Let $s(t)$ be a continuous, real-valued, wide-sense stationary (WSS) Gaussian random process with zero mean, autocorrelation function $r_s(t-u) = E\{s(t)s(u)\}$, and variance $\sigma_s^2 = r_s(0)$. Let $y(t)$ be a known, deterministic signal. We wish to determine the sequence of crossing-points $\tau_1 < \tau_2 < \ldots$ of $s(t)$ and $y(t)$, or equivalently, the zero-crossings of $z(t)$ defined as follows:

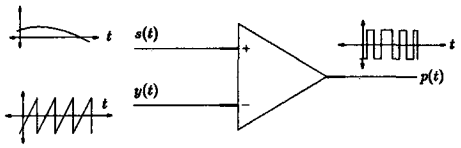
$$z(t) \triangleq s(t) - y(t). \tag{3.1}$$

In a continuous-time system, these crossing points may be detected using the comparator circuits shown in Figure 3.1. In these circuits, the crossing points are encoded in the edge positions of the output signal. The floating-reference circuit in Figure 3.1(a) is clearly

identical to the ground-referenced circuit in Figure 3.1(b); however, the equivalent forms lead naturally into distinct estimators in the sequel.



(a) Ground-referenced comparator



(b) Floating-reference comparator

Fig. 3.1 Comparator Circuits for Continuous-Time Detection of Crossing Points

In the discrete-time problem, we are provided with a set of K consecutive uniformly-spaced noisy samples from $s(t)$, where $k \in \mathbb{Z}$ is an integer index. We define these noisy samples as $x[k]$, i.e.

$$x[k] \triangleq s(kT_s + T_d) + n[k] \tag{3.2}$$

where T_s denotes the sampling period, T_d is a sampling offset, and $n[k]$ is an additive measurement noise. This noise signal may be used to model, for example, quantization effects. We model $n[k]$ as a WSS discrete-time Gaussian random process with zero mean and autocorrelation function $r_n[k-l] = \sigma_n^2 \delta[k-l]$, where σ_n^2 denotes the variance and δ is the Kronecker delta function.[1] The relationship between $s(t)$ and $x[k]$ is depicted in Figure 3.2.

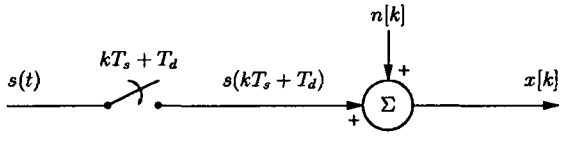


Fig. 3.2 Sampling and Noise Model

[1] That is, $\delta[k] = 1$ for $k = 0$; $\delta[k] = 0$ otherwise.

The discrete-time crossing-point estimation problem may be stated as follows. Given the known carrier $y(t)$ and K samples of $x[k]$, estimate the sequence of points $\tau_0 < \tau_1 < \cdots$ satisfying $s(\tau_i) = y(\tau_i)$, or equivalently, $z(\tau_i) = 0$.

Because K is arbitrary and the number of crossing points τ_i is not limited, the complexity of this problem is unbounded. In the following section, we describe a two-step approach to the discrete-time crossing-point estimation problem with a much simpler form.

3.2 Simplified Discrete-Time Crossing-Point Estimation

Following published approaches to discrete-time crossing-point estimation, [14, 7, 12] we impose a two-step structure on our solution to the general problem as shown in Figure 3.3. Each step may be viewed as a separate estimation process:

- first, coarsely locate crossing-points using the estimator E_1;
- then, apply the more refined estimator E_2 in the neighbourhood of each crossing point to generate a more exact estimate.

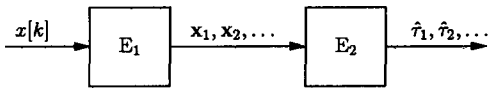


Fig. 3.3 Two-Step Crossing-Point Estimation

In the coarse estimator E_1, we define $\xi[k]$ as follows:

$$\xi[k] = x[k] - y(kT_s + T_d) \tag{3.3}$$

The coarse estimator monitors $\xi[k]$ for changes in sign. When the noise term is sufficiently small, $\xi[k] \approx z(kT_s + T_d)$. Thus, neglecting the possibility that multiple zero crossings of $z(t)$ occur within each sampling interval, sign changes in $\xi[k]$ (i.e. when $\xi[k-1]\xi[k] < 0$) bound each zero crossing τ_i to a single sample interval $\mathcal{T}_i$ defined as follows:

$$\tau_i \in \mathcal{T}_i \triangleq ([k-1]T_s + T_d, kT_s + T_d) \tag{3.4}$$

In (3.4) and throughout the remaining development, we neglect the vanishing probability that the zero crossing occurs precisely on a sampling instant. The coarse estimator may

miss zero crossings when the sampling rate is not sufficiently large when compared to the Nyquist rate, because multiple zero crossings may occur in a single sample interval. A number of improvements to E_1 are possible; however, this estimator is not the focus of our investigation.

Once the coarse bound on τ_i has been generated, we restrict our investigation to samples immediately surrounding $\mathcal{T}_i$. We form the M-dimensional vectors $\mathbf{x}_i$ and $\boldsymbol{\xi}_i$ using M_1 consecutive samples of $x[k]$ and $\xi[k]$ preceeding τ_i and $M_2 = M - M_1$ consecutive samples immediately following τ_i:

$$\mathbf{x}_i = \begin{bmatrix} x[k-M_1] \\ \cdots \\ x[k-1] \\ x[k] \\ \cdots \\ x[k+M_2-1] \end{bmatrix} \qquad \boldsymbol{\xi}_i = \begin{bmatrix} \xi[k-M_1] \\ \cdots \\ \xi[k-1] \\ \xi[k] \\ \cdots \\ \xi[k+M_2-1] \end{bmatrix} \tag{3.5}$$

The second estimator E_2 must solve the simplified discrete-time crossing-point estimation problem, which is defined as follows: Given $y(t)$, the sample vector $\mathbf{x}_i$ (equivalently, $\boldsymbol{\xi}_i$), and the bracketing interval $\mathcal{T}_i$ for the ith zero crossing of $z(t)$, find an estimate $\hat{\tau}_i$ of the true crossing-point τ_i.

Because the carrier signal $y(t)$ is completely specified, we are able to consider E_2 in two equivalent forms. These forms are the discrete-time equivalents of the continuous-time cases shown in Figure 3.1. In one form, we use the vector $\mathbf{x}_i$ to determine crossing points of $s(t)$ and $y(t)$. In the second form, we use the vector $\boldsymbol{\xi}_i$ to estimate zero crossings of $z(t)$. These two forms are shown in Figures 3.4 and 3.5.

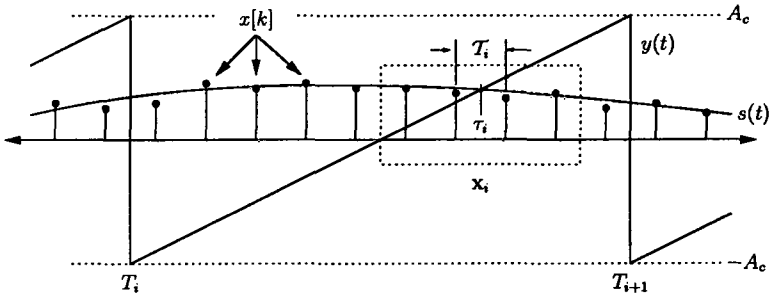


Fig. 3.4 Forming $\mathbf{x}_i$; $M = 4$

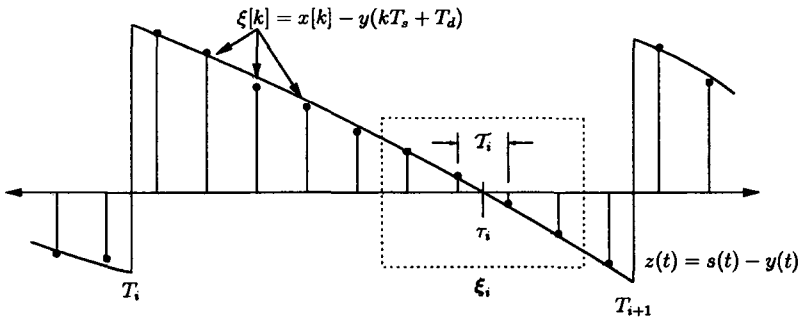


Fig. 3.5 Forming $\boldsymbol{\xi}_i$; $M = 4$

In the following chapter, we focus on E_2 and attempt to develop good estimators for τ_i by exploiting statistical knowledge about the input vector $\mathbf{x}_i$. Because henceforth we consider only a single crossing point at a time, we may drop the subscript i on τ_i, $\mathcal{T}_i$, $\mathbf{x}_i$, and $\boldsymbol{\xi}_i$ without ambiguity.

We have now defined the discrete-time crossing-point estimation problem in its general and simplified forms. When considering these problems as they apply to PWM and CM systems, several additional constraints are imposed that greatly simplify the derivations that follow. We consider these constraints in the following section.

3.3 Carrier Function Constraints

In the previous sections, we formally defined the crossing-point estimation problem under the assumption the carrier function $y(t)$ was completely specified. The actual form of the carrier signal $y(t)$ depends on the application. In this section, we explore the effects the carrier signal has on the problem definition.

In both CM and PWM, the carrier function $y(t)$ is a periodic function: a sinusoid in the case of CM, and a triangle-wave or a sawtooth for different varieties of PWM. Figure 3.6 shows the permissible carrier waveform for each scheme. For these carriers, the phase is arbitrary but known, the time average (DC) value is zero, and the peak amplitude is $\pm A_c$.

For the discontinuous (upper) waveforms in Figure 3.6, the discontinuities in $y(t)$ produce additional crossing-points whose locations are exactly known; we therefore neglect these edges and focus on the unknown (variable) crossing points.

In order for our model to accurately match the application, we wish to guarantee that

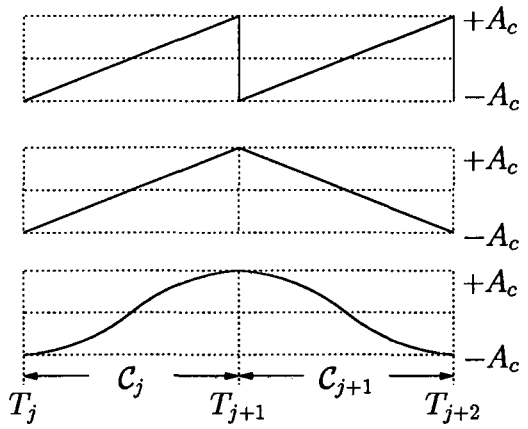


Fig. 3.6 Typical Click and Pulse-Width Modulation Carrier Functions

exactly one crossing-point occurs in each interval $\mathcal{C}_j$ defined as follows:

$$\mathcal{C}_j \triangleq (T_j, T_{j+1}) \tag{3.6}$$

To do so, we begin by bounding $s(t)$ such that $-A_c < s(t) < A_c$; this ensures at least one crossing point occurs in $\mathcal{C}_j$.

Avoiding multiple zero crossings within each interval $\mathcal{C}_j$ involves a more complicated relationship between the carrier period and signal bandwidth [25]. The occurance of multiple, or no, crossing-points within each interval indicates an undesirable condition known as modulator overload. In the sequel, we assume modulator overload does not occur.[2]

3.4 Motivating a Closer Look

In this chapter, we formally defined the crossing-point problem. We introduced statistical characterizations of the signals involved, and introduced a model for the effects of measurement noise.

To motivate the estimators developed in the following chapter, we note that none of the published crossing-point estimators we reviewed in Section 1.2 allowed this statistical information (such as the sample autocorrelation) to be exploited. In addition, these estimators

[2]We cannot, of course, guarantee that the Gaussian signal $s(t)$ is absolutely bounded below a particular amplitude with probability 1. Instead, we require the probability of an excursion beyond A_c to be negligible.

were introduced on an ad hoc basis, prompting questions about their generality.

For example, consider a crossing-point estimator that approximates $s(t)$ using $\hat{s}(t)$, defined as the unique polynomial of order $N-1$ passing through the N points of $\mathbf{x}$. This estimator, which we will denote POLS in the sequel, generates crossing-point estimates τ_i by determining the points at which $y(t) = \hat{s}(t)$ using a root-finding method such as Newton's method.

How general is this estimator? Polynomial interpolation has been shown [31] to approximate sinc-kernel interpolation when the sampling rate f_s is high compared to the Nyquist rate for the signal being interpolated. This suggests the POLS estimator may perform well in the absence of noise, when $s(t)$ is a lowpass signal and f_s is high compared to the Nyquist rate. However, when noise is substantial, or when $s(t)$ is not suitably oversampled, POLS may not perform adequately. As we will show, an estimator that intelligently makes use of autocorrelation information is superior to polynomial curve fitting.

Various improvements to basic polynomial interpolation (e.g. PLS, or Penalized Least-Squares curve fitting [32]) have been examined in the literature; few of them have been evaluated in the specific contexts of PWM amplification or crossing-point problems. PLS, for example, balances fidelity (how close the chosen approximating function comes to each of the datapoints) with the smoothness of the fitted curve. In the case of polynomial approximations, such a scheme might vary the order of the polynomial adaptively, balancing mean-squared error with a penalty that promotes smooth (low-order) solutions. This approach raises several questions: How can known or predictable parameters (e.g. bandwidth, noise, or variance) be used to adapt order? Is the additional complexity tolerable? Finally, if we neglect noise (or assume it can be perfectly removed under such a scheme), there is still no guarantee that the adapted curve suitably matches the underlying function. However, the estimators we examine also have an order associated with them; dynamically modifying this parameter is an interesting avenue for future research.

Finally, a statistical investigation of the crossing-point estimation problem can yield results beyond the estimators themselves. For example, the Cramér-Rao bound for the problem (which we derive in the following chapter) allows the performance of nonstatistical and statistical estimators alike to be evaluated against an absolute limit. Without knowledge of this limit, it is difficult to establish how good a crossing-point estimator can get.

Chapter 4

Algorithm Development

Having defined the crossing-point estimation problem in the preceding chapter, we now investigate several solutions. We first derive the Maximum A-Posteriori (MAP) estimator for the problem, a methodical process that furnishes an estimator which is known to approach optimality under certain criteria. The MAP estimator is, however, not always convenient or efficient to calculate. Following the derivation of the MAP estimator, we introduce the Minimum Mean-Squared Error (MMSE) estimator, a computationally straightforward estimator. We link the newly developed MAP and MMSE estimators by showing that the MMSE estimator is an approximation to the MAP estimator under certain conditions.

We then derive the Cramér-Rao bound (CRB) for the problem. The CRB reflects the best-case performance for any unbiased crossing-point estimator (which may not necessarily be attainable); it allows practical estimators to be judged against an absolute metric.

We also introduce a number of reference estimators to the discrete-time crossing-point estimation problem. These estimators have been selected from the literature and represent the most common solutions currently adopted in practice.

4.1 Distribution of Sample Vectors

Consider the discrete-time random process $x[k]$ given by (3.2), where $s(t)$ and $n[k]$ are statistically independent Gaussian, wide-sense stationary (WSS) processes with zero mean and respective autocorrelation functions $r_s(t-u)$ (continuous-time) and $\sigma_n^2\delta[n-l]$ (discrete-time). It follows that $x[k]$ is Gaussian with zero mean, autocorrelation function $r[k] = r_s(kT_s) + \sigma_n^2\delta[k]$, and variance $\sigma^2 = r[0] = \sigma_s^2 + \sigma_n^2$, where $\sigma_s^2 = r_s(0)$.

We now consider the joint distribution of the M-dimensional observation vector $\mathbf{x}$ defined as follows, where $M = M_1 + M_2$:

$$\mathbf{x} = \begin{bmatrix} x[k-M_1] \\ \cdots \\ x[k-1] \\ x[k] \\ \cdots \\ x[k+M_2-1] \end{bmatrix} = \begin{bmatrix} s([k-M_1]T_s + T_d) + n[k-M_1] \\ \cdots \\ s([k-1]T_s + T_d) + n[k-1] \\ s(kT_s + T_d) + n[k] \\ \cdots \\ s([k+M_2-1]T_s + T_d) + n[k+M_2-1] \end{bmatrix} \tag{4.1}$$

We may also express the observation vector $\mathbf{x}$ as $\mathbf{x} = \mathbf{s} + \mathbf{n}$, where $\mathbf{s}$ and $\mathbf{n}$ are, respectively, the (unobservable) contributions from $s(kT_s + T_d)$ and $n[k]$:

$$\mathbf{s} = \begin{bmatrix} s([k-M_1]T_s + T_d) \\ \cdots \\ s([k-1]T_s + T_d) \\ s(kT_s + T_d) \\ \cdots \\ s([k+M_2-1]T_s + T_d) \end{bmatrix} \qquad \mathbf{n} = \begin{bmatrix} n[k-M_1] \\ \cdots \\ n[k-1] \\ n[k] \\ \cdots \\ n[k+M_2-1] \end{bmatrix} \tag{4.2}$$

The probability density function (pdf) of $\mathbf{x}$, $f(\mathbf{x})$, takes the standard multivariate Gaussian form:

$$f(\mathbf{x}) = \frac{1}{(2\pi)^{M/2}|\Sigma_0|^{1/2}} \exp\left(-\frac{1}{2}\mathbf{x}^T\Sigma_0^{-1}\mathbf{x}\right) \tag{4.3}$$

where $\Sigma_0 = E\{\mathbf{x}\mathbf{x}^T\}$ is an $M \times M$ symmetric Toeplitz sample covariance matrix with determinant $|\Sigma_0|$. We have:

$$\begin{aligned} \Sigma_0 &= E\{(\mathbf{s}+\mathbf{n})(\mathbf{s}+\mathbf{n})^T\} \\ &= E\{\mathbf{s}\mathbf{s}^T\} + E\{\mathbf{n}\mathbf{n}^T\} \\ &= \Sigma_s + \sigma_n^2 I \end{aligned} \tag{4.4}$$

where we have defined $\Sigma_s \triangleq E\{\mathbf{s}\mathbf{s}^T\}$, and where I is the $M \times M$ identity matrix. For subsequent analysis, it is convenient to introduce the correlation vector function $\boldsymbol{\rho}(\tau)$ defined

as follows:

$$\boldsymbol{\rho}(\tau) \triangleq E\{s(\tau)\mathbf{s}\} = \begin{bmatrix} r_s(\tau - [k - M_1]T_s - T_d) \\ \vdots \\ r_s(\tau - [k + M_2 - 1]T_s - T_d) \end{bmatrix}.$$

Now, Σ_s may be expressed in terms of $\boldsymbol{\rho}(\tau)$ as

$$\Sigma_s = \begin{bmatrix} \boldsymbol{\rho}([k - M_1]T_s + T_d)^T \\ \vdots \\ \boldsymbol{\rho}([k + M_2 - 1]T_s + T_d)^T \end{bmatrix}. \tag{4.5}$$

We assume Σ_s is positive definite. In this case, Σ_0 is also positive definite (even in the absence of noise, i.e. when $\sigma_n^2 = 0$.)

4.2 Maximum A-Posteriori (MAP) Estimation

Let $f(\tau|\mathbf{x})$ be the probability density function (pdf) of a zero crossing of $z(t)$ at time $t = \tau$ conditioned on the sample vector $\mathbf{x}$ and given $\tau \in \mathcal{T}$. (Because of the coarse estimator E_1, the search is limited to $\tau \in \mathcal{T}$; please refer to Section 3.2 for details, noting that the subscript i is implicit here. In the sequel, τ is implicitly limited to this range.) The MAP estimate of τ maximizes this function, i.e.:

$$\hat{\tau}_{\mathrm{MAP}} = \underset{\tau}{\operatorname{argmax}}\, f(\tau|\mathbf{x}) \tag{4.6}$$

Let $f(\tau)$ and $f(\mathbf{x})$ represent, respectively, the a-priori pdfs of a zero crossing at time τ and the sample vector $\mathbf{x}$. Further, let $f(\mathbf{x}|\tau)$ be the pdf of the sample vector $\mathbf{x}$ conditioned on a crossing point at τ. We expand the conditional distribution $f(\tau|\mathbf{x})$ using Bayes' rule:

$$f(\tau|\mathbf{x}) = \frac{f(\mathbf{x}|\tau)f(\tau)}{f(\mathbf{x})} \tag{4.7}$$

The MAP estimate is the maximum (with respect to τ) of (4.7). We differentiate the logarithm of the right-hand expression with respect to τ and set the result to 0. This

process yields the canonical MAP equation [33]:

$$\frac{d}{d\tau}\log f(\mathbf{x}|\tau)\bigg|_{\tau=\hat{\tau}_{\text{MAP}}} = -\frac{d}{d\tau}\log f(\tau)\bigg|_{\tau=\hat{\tau}_{\text{MAP}}}$$

We define $S(\mathbf{x}|\tau)$ and $S(\tau)$ (which are sometimes called the conditional and a-priori score functions, respectively) as follows:

$$S(\mathbf{x}|\tau) \triangleq \frac{d}{d\tau}\log f(\mathbf{x}|\tau) \qquad\qquad S(\tau) \triangleq \frac{d}{d\tau}\log f(\tau) \tag{4.8}$$

The MAP estimate now satisfies

$$S(\mathbf{x}|\tau)|_{\tau=\hat{\tau}_{\text{MAP}}} = -\,S(\tau)|_{\tau=\hat{\tau}_{\text{MAP}}} \tag{4.9}$$

In the sequel, we express componentwise derivatives of scalars, vectors, and matrices (which are always with respect to τ) using a dot notation. For example, $\dot{y}(\tau) \triangleq dy(\tau)/d\tau$.

In the following subsections, we derive expressions for $S(\tau)$ and $S(\mathbf{x}|\tau)$.

4.2.1 A-Priori Crossing-Point Density

In this section, we consider the probability density $f_\tau(t)$ of a crossing point at time t, i.e. the probability that $s(t) = y(t)$. To simplify the derivation of this distribution, we make three crucial assumptions:

1. The probability that $|s(t)| \geq A_c$ (i.e. overload) is negligible,
2. The carrier waveform $y(t)$ is invertible (i.e. one-to-one) in every interval $\mathcal{C}_j$, and
3. One (and *only* one) crossing point occurs in every interval $\mathcal{C}_j$.

As described in Chapter 2, each of these assumptions is satisfied in the applications of CM and PWM. Consider a segment $\mathcal{C}_j$ of $y(t)$. (It suffices to consider a rising segment of $y(t)$, as falling segments follow by symmetry.) We use the second and third assumptions to write the following equivalence:

$$F_\tau(t) \triangleq \Pr\{\tau \leq t\} = \Pr\{s(t) \leq y(t)\} \tag{4.10}$$

That is, for rising carrier segments, a crossing point occurs in the interval $(\infty, t) \cap \mathcal{C}_j$ if and only if $s(t) < y(t)$. Thus, each segment of the carrier $y(t)$ allows us to map the known density $f_S(s)$ into the desired density $f_\tau(t)$. This mapping process is illustrated in Figure 4.1.

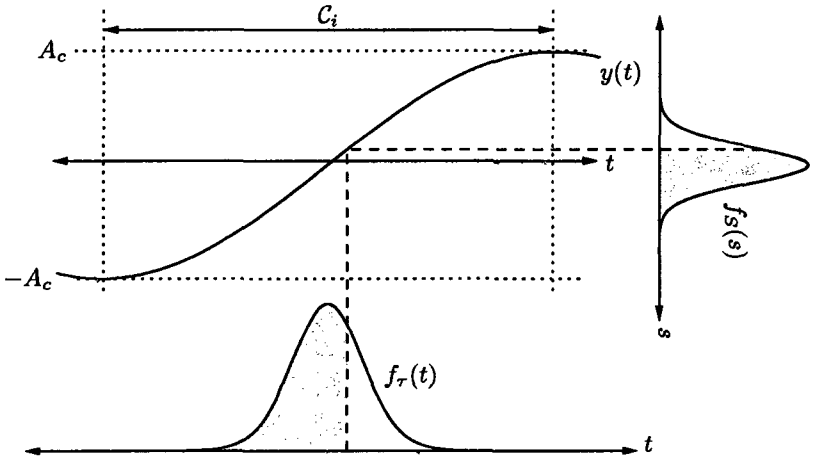


Fig. 4.1 Deriving the A-Priori Crossing Point Probability Density

We have, in the interval $t \in \mathcal{C}_j$:

$$F_\tau(t) \approx \int_{-A_c}^{y(t)} f_S(s)ds = \int_{-A_c}^{y(t)} \frac{1}{\sqrt{2\pi\sigma_s^2}} e^{-\frac{1}{2\sigma_s^2}s^2} ds$$

We differentiate under the integral sign, giving

$$f(\tau) = \frac{\exp\left(-\frac{1}{2\sigma_s^2}y(\tau)^2\right)}{\sqrt{2\pi\sigma_s^2}} \dot{y}(\tau)$$

for $\tau \in \mathcal{C}_j$. This result may be extended to sections of $y(t)$ with negative slope by symmetry:

$$f(\tau) = \frac{\exp\left(-\frac{1}{2\sigma_s^2}y(\tau)^2\right)}{\sqrt{2\pi\sigma_s^2}} |\dot{y}(\tau)| \tag{4.11}$$

for $\tau \in \mathcal{C}_j$. This expression will be used both for MAP estimation and in evaluating the CRB.

4.2.2 Deriving the A-Priori Score Function

Solving for $S(\tau) = \frac{d}{d\tau} \log f(\tau)$ is straightforward when $f(\tau)$ takes the form (4.11). For $\tau \in \mathcal{C}_j$, $y(\tau)$ is one-to-one; therefore, $\dot{y}(\tau) \neq 0$ for all points within it. We have:

$$S(\tau) = \frac{\ddot{y}(\tau)}{\dot{y}(\tau)} - \frac{\dot{y}(\tau)y(\tau)}{\sigma_s^2}. \tag{4.12}$$

Alternately, in the absence of a-priori information about crossing points, we may model τ as uniformly distributed on $\mathcal{C}_j$. This approach is mathematically equivalent to Maximum-Likelihood (ML) estimation. In simulation, the two approaches are essentially indistinguishable, indicating that the a-priori distribution $f(\tau)$ is uninformative under the operating conditions we evaluate. (This point will be discussed further in Chapter 5.)

4.2.3 Conditional Distribution of Sample Vectors

We now consider the distribution of the vector $\mathbf{x}$ conditioned on a crossing point at time τ, or equivalently from (3.1), given $s(\tau) = y(\tau)$. We begin by forming the augmented sample vector

$$\mathbf{x}_\tau = [\ \mathbf{x}^T\ ,\ s(\tau)\ ]^T = [\ (\mathbf{n}^T + \mathbf{s}^T)\ ,\ s(\tau)\ ]^T. \tag{4.13}$$

As $\mathbf{n}$ and $s(\tau)$ are statistically independent and Gaussian, $\mathbf{x}_\tau$ is a Gaussian random vector with zero mean and covariance matrix Σ_τ which may be expressed in partitioned form as follows:

$$\Sigma_\tau = \left[\begin{array}{c|c} \Sigma_0 & \boldsymbol{\rho}(\tau) \\ \hline \boldsymbol{\rho}(\tau)^T & \sigma_s^2 \end{array}\right] \tag{4.14}$$

The conditional distribution of $\mathbf{x}$ given τ is equivalent to the addition of a new random variable with a known value, and can be expressed as

$$\begin{aligned} f(\mathbf{x}|\tau) &= f\left(\mathbf{x}_\tau | s(\tau) = y(\tau)\right) \\ &= \frac{f(\mathbf{x}_\tau)|_{s(\tau)=y(\tau)}}{f_s(y(\tau))} \end{aligned} \tag{4.15}$$

The pdf $f(\mathbf{x}|\tau)$ can be shown to be Gaussian with covariance matrix $\Sigma(\tau)$ and mean $\boldsymbol{\mu}(\tau)$ defined as follows: [34]

$$\Sigma(\tau) = \Sigma_0 - \frac{\boldsymbol{\rho}(\tau)\boldsymbol{\rho}^T(\tau)}{\sigma_s^2} \tag{4.16}$$

$$\boldsymbol{\mu}(\tau) = \frac{\boldsymbol{\rho}(\tau)y(\tau)}{\sigma_s^2} \tag{4.17}$$

We will frequently require the derivatives of $\Sigma(\tau)$ and $\boldsymbol{\mu}(\tau)$. For convenience, we differentiate these quantities here:

$$\dot{\Sigma}(\tau) = -\frac{\dot{\boldsymbol{\rho}}(\tau)\boldsymbol{\rho}^T(\tau) + \boldsymbol{\rho}(\tau)\dot{\boldsymbol{\rho}}^T(\tau)}{\sigma_s^2} \tag{4.18}$$

$$\dot{\boldsymbol{\mu}}(\tau) = \frac{\dot{\boldsymbol{\rho}}(\tau)y(\tau) + \boldsymbol{\rho}(\tau)\dot{y}(\tau)}{\sigma_s^2} \tag{4.19}$$

In the following development, we suppress dependence of y, $\boldsymbol{\rho}$, Σ, and $\boldsymbol{\mu}$ on τ to simplify notation. We have:

$$f(\mathbf{x}|\tau) = \frac{\exp\left(-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^T\Sigma^{-1}(\mathbf{x}-\boldsymbol{\mu})\right)}{(2\pi)^{M/2}|\Sigma|^{1/2}} \tag{4.20}$$

These expressions completely characterize the conditional distribution. We now continue deriving the MAP estimator.

4.2.4 Deriving the Conditional Score Function

The MAP estimate of τ satisfies (4.9). In order to derive the MAP estimator, we require an expression for $S(\mathbf{x}|\tau)$. Combining (4.20) and (4.8), we have:[1]

$$S(\mathbf{x}|\tau) = -\frac{1}{2}\frac{d}{d\tau}\left[\log|\Sigma| + (\mathbf{x}-\boldsymbol{\mu})^T\Sigma^{-1}(\mathbf{x}-\boldsymbol{\mu})\right] \tag{4.21}$$

To simplify this expression, we introduce the following definitions:

$$\begin{aligned} a &= \boldsymbol{\rho}^T\Sigma^{-1}\boldsymbol{\rho}/\sigma_s^2 & c &= \mathbf{x}^T\Sigma^{-1}\boldsymbol{\rho} \\ b &= \boldsymbol{\rho}^T\Sigma^{-1}\dot{\boldsymbol{\rho}}/\sigma_s^2 & d &= \mathbf{x}^T\Sigma^{-1}\dot{\boldsymbol{\rho}} \\ e &= \dot{\boldsymbol{\rho}}^T\Sigma^{-1}\dot{\boldsymbol{\rho}}/\sigma_s^2 \end{aligned} \tag{4.22}$$

[1]Full derivations of (4.21), (4.23) and (4.26) are provided in Appendix A

Using some manipulations and elementary identities for differentiating matrix expressions, (4.21) may be expressed as follows:

$$S(\mathbf{x}|\tau) = b - \frac{(ay-c)(by-d) + \dot{y}(ay-c) + y(by-d)}{\sigma_s^2} \tag{4.23}$$

Due to the terms a, b, c, and d, this form for $S(\mathbf{x}|\tau)$ still depends on the conditional covariance matrix Σ, which in turn depends on τ. We may use the Sherman-Morrison-Woodbury formula [35] to express Σ^{-1} in terms of Σ_0^{-1}:

$$\Sigma^{-1} = \Sigma_0^{-1} + \frac{\Sigma_0^{-1}\boldsymbol{\rho}\boldsymbol{\rho}^T\Sigma_0^{-1}}{\sigma_s^2 - \boldsymbol{\rho}^T\Sigma_0^{-1}\boldsymbol{\rho}} \tag{4.24}$$

This formula is valid provided that Σ_0 is nonsingular (which is assumed a-priori) and provided $\sigma_s^2 \neq \boldsymbol{\rho}^T\Sigma_0^{-1}\boldsymbol{\rho}$, which is satisfied in the presence of noise (See Appendix C for proof.) We apply (4.24) to each of the definitions (4.22) in order to express $S(\mathbf{x}|\tau)$ without using the conditional covariance matrix. Let:

$$\begin{aligned} a_0 &= \boldsymbol{\rho}^T\Sigma_0^{-1}\boldsymbol{\rho}/\sigma_s^2 & c_0 &= \mathbf{x}^T\Sigma_0^{-1}\boldsymbol{\rho} \\ b_0 &= \boldsymbol{\rho}^T\Sigma_0^{-1}\dot{\boldsymbol{\rho}}/\sigma_s^2 & d_0 &= \mathbf{x}^T\Sigma_0^{-1}\dot{\boldsymbol{\rho}} \\ e_0 &= \dot{\boldsymbol{\rho}}^T\Sigma_0^{-1}\dot{\boldsymbol{\rho}}/\sigma_s^2 \end{aligned} \tag{4.25}$$

After substitution, (4.23) may be expressed as:

$$S(\mathbf{x}|\tau) = \frac{d_0 y}{\sigma_s^2} - \frac{b_0(a_0y - c_0)(y - c_0)}{\sigma_s^2(1-a_0)^2} + \frac{b_0\sigma_s^2 + (d_0 - \dot{y})(a_0y - c_0) - yb_0(y - c_0)}{\sigma_s^2(1-a_0)} \tag{4.26}$$

Combining the above results, we may now state the fully simplified MAP estimator.

4.2.5 The MAP Estimator

Combining (4.26) and (4.9), and once again noting that the dependence on τ has been omitted, the MAP estimate satisfies

$$-S(\tau)|_{\tau=\hat{\tau}_{\text{MAP}}} = \frac{d_0 y}{\sigma_s^2} - \frac{b_0(a_0y - c_0)(y - c_0)}{\sigma_s^2(1-a_0)^2} + \frac{b_0\sigma_s^2 + (d_0 - \dot{y})(a_0y - c_0) - yb_0(y - c_0)}{\sigma_s^2(1-a_0)} \tag{4.27}$$

where either (4.12) or 0 (when $f(\tau)$ is uninformative) may be used in place of $S(\tau)$.

In order to generate estimates using this expression, a zero-finding method such as Brent's algorithm [36] or Newton's method may be applied. There is no a-priori guarantee that an unique zero of (4.27) exists in $\mathcal{T}$, and that it corresponds to the maximum of (4.6) within $\mathcal{T}$. We do not investigate uniqueness of the solution in this work. Although (4.27) may exhibit multiple candidates in $\mathcal{T}$, judicious selection of a starting point for the zero-finding algorithm produces excellent results. (In simulations, we provide the MMSE estimate described below as a starting point to the MAP estimator.) The experimental results presented in the following chapter do not suggest any investigation of uniqueness is necessary in practice.

Although (4.27) cannot be simplified further, it is both numerically sensitive and computationally expensive. (One such numerical problem is briefly described in the following section.) To design a more practical estimator, we will make a number of approximations in order to simplify (4.27). In doing so, we will arrive at a MMSE formulation for the problem.

4.3 Minimum Mean-Squared Error (MMSE) Estimation

Consider the minimum mean-squared error (MMSE) estimate $\hat{s}(t)$ of $s(t)$ at arbitrary t given the vector $\mathbf{x}$ of nearby samples. This estimate is given by the Wiener-Hopf equation: [37]

$$\hat{s}(t) = \boldsymbol{\rho}(t)^T \Sigma_0^{-1} \mathbf{x} = c_0$$

The expected mean-squared error for this estimate is given by: [37]

$$\epsilon = \sigma_s^2 - \boldsymbol{\rho}(t)^T \Sigma_0^{-1} \boldsymbol{\rho}(t) = \sigma_s^2 (1 - a_0)$$

As ϵ is a variance and Σ_0 is nonnegative definite, $\epsilon \in [0, \sigma_s^2]$. We may consider ϵ to be a measure of confidence in $\hat{s}(t)$. When $\mathbf{x}$ consists of samples near t, we should expect this error to be very small compared to σ_s^2.

Due to the 2-step structure of our estimator, τ is bounded by (3.4). This bound allows us to choose a vector $\mathbf{x}$ of nearby samples such that $1 - a_0$ is very small over the region of possible zero crossings. When $1 - a_0$ is very small, the second term on the right-hand side of (4.27) dominates. At the MAP estimate $\hat{\tau}$ (where, again, dependence on the time

variable is suppressed), we have:

$$\begin{aligned} 0 &\approx \frac{b_0(a_0 y - c_0)(y - c_0)}{\sigma_s^2(1-a_0)^2} \\ &\approx \frac{b_0(y-c_0)^2}{\sigma_s^2(1-a_0)^2} \end{aligned} \tag{4.28}$$

Thus, when $1-a_0$ is very small, and neglecting for a moment that a_0 and c_0 are themselves functions of τ, the upper expression in (4.28) is approximately parabolic in y with two real roots very near to each other. Our approximation replaces these two roots with a single root of multiplicity 2. In both cases, these two nearby (or coalesced) roots can cause great difficulty in locating MAP estimates using (4.27), since the numerically calculated curve only gradually approaches (and, given computational errors, may not necessarily cross) the origin. Some zero-finding algorithms tend to reject solutions with even multiplicities and converge to nearby (but incorrect) solutions [36]. In practice, it is sometimes necessary to minimize the square of (4.27) instead of using a zero-finding algorithm in order to avoid this problem.

The MMSE (or approximated MAP) estimate corresponds to solutions of (4.28). The two candidates are roots of b_0 and $y - c_0$. However, as b_0 is a function of τ and Σ_0^{-1} only, it does not involve the sample vector in any way. This term may contribute only static solutions to the estimator equation, which we may disregard. Moreover, as we will see, choosing the roots of the remaining term results in an estimator which performs very well and is intuitively satisfying.

Defining the M-dimensional vector $\mathbf{w}_{\text{MMSE}} = \Sigma_0^{-1}\mathbf{x}$, the MMSE estimate must satisfy the simple relation:

$$\mathbf{w}^T \boldsymbol{\rho}(\tau) = y(\tau) \tag{4.29}$$

We denote this estimate as $\hat{\tau}_{\text{MMSE}}$:

$$\hat{\tau}_{\text{MMSE}} = \arg_\tau \left[\mathbf{w}^T \boldsymbol{\rho}(\tau) = y(\tau)\right] \tag{4.30}$$

This result has an intuitive form, since it relates the MMSE estimate of $s(t)$ to the known carrier signal $y(t)$ and solves for the points at which they are equal. In contrast to ordinary MMSE estimation, the relationship between time and amplitude $y(\tau)$ is known; here, we seek the unknown crossing-point time τ using this relationship.

Since both the carrier signal and $\boldsymbol{\rho}(\tau)$ are in general nonlinear functions, a root-finding method must once again be adopted. The structure of the MMSE estimator is illustrated in Algorithm 1.

Algorithm 1 Crossing-Point Estimation via the MMSE Algorithm

```
loop
  if ξ[k − 1]ξ[k] < 0 then
    x = [x[k − M1], ..., x[k + M2 − 1]]^T
    w = Σ^{-1}x (using, e.g. Cholesky factorization of Σ^{-1})
    repeat
      refine guess τ̂ via Newton's or Brent's algorithms
      ε = |w^T ρ(τ̂) − y(τ̂)|
    until ε < tol
  end if
end loop
```

Similar to the MAP estimator above, we have assumed the existence and uniqueness of solutions to (4.30) in $\mathcal{T}$ without providing any assurances of either property. In practice, no such problems arise. In addition, the computational difficulties in the MAP estimator do not occur in the MMSE formulation, and a simple zero-finding algorithm performs very well. (Indeed, we use the MMSE estimate as an initial guess to the MAP estimator during simulation.)

4.4 MMSEZ Estimator

We briefly digress and introduce the MMSEZ estimator, which is a further simplification of the MMSE estimator. However, unlike the MMSE interpolator, the MMSEZ interpolator performs measurably worse than the MAP estimator in simulations. We include the MMSEZ estimator chiefly because the relationship between MMSE and MMSEZ estimators is analogous to the relationship between POLS and POLZ estimators (which are described below in Sections 4.6.2 and 4.6.3), both in structure and performance.

Consider the two-step estimation process illustrated in Figure 3.3. During the estimation step E_1, periodic samples $y(kT_s + T_d)$ of the carrier signal are required to coarsely bound crossing points. From the computational perspective of E_2, these samples of $y(\tau)$ are "free." Otherwise, particularly when $y(t)$ is nonlinear, evaluations at arbitrary t may be expensive enough to avoid.

To take advantage of these known samples, we generate a MMSE estimator for $\boldsymbol{\xi} = [\xi[k - M_1], \ldots, \xi[k + M_2 - 1]]^T$ and solve for its zero crossings. To do so, we define a new vector correlation function $\boldsymbol{\rho}_z(\tau) = E\{z(\tau)\boldsymbol{\xi}\}$ and covariance matrix $\Sigma_z = E\{\boldsymbol{\xi}\boldsymbol{\xi}^T\}$ which are analogous to $\boldsymbol{\rho}(\tau)$ and Σ_0, but which include statistical information about $y(t)$ between samples. (It is not always obvious what the best choice for $\boldsymbol{\rho}_z(\tau)$ is, particularly when $y(t)$ is discontinuous. A detailed investigation of the MMSEZ estimator is not pursued here.)

Algorithm 2 Crossing-Point Estimation via the MMSEZ Algorithm

```
loop
  if ξ[k − 1]ξ[k] < 0 then
    ξ = [ξ[k − M1], ..., ξ[k + M2 − 1]]^T
    w = Σz^{-1} ξ (using, e.g. Cholesky factorization of Σz^{-1})
    repeat
      refine guess τ̂ via Newton's or Brent's algorithms
      ε = |w^T ρz(τ̂)|
    until ε < tol
  end if
end loop
```

Compared to MMSE (see Algorithm 1), MMSEZ avoids computations of $y(\hat{\tau})$ in the root-finding step. As we will see in the following chapter, this reduction in complexity can be accompanied by reduced performance. Moreover, the evaluation of *each element* of $\boldsymbol{\rho}_z(\hat{\tau})$ is likely to incur a similar computational cost to each evaluation of $y(\hat{\tau})$; therefore, the savings may not be meaningful.

4.5 Fundamental Performance Limits

In this section, we consider best-case and worst-case performance metrics for crossing-point estimators. For the best-case limit, we derive the Cramér-Rao bound for the variance of an unbiased crossing-point estimator. The worst-case metric is provided by considering the variance of random crossing-point estimates.

4.5.1 Cramér-Rao Bound (CRB)

The Cramér-Rao bound for the random parameter τ bounds the variance $\sigma^2_{\hat{\tau}|\tau\in T}$ of any unbiased estimator as follows: [33]

$$\sigma^2_{\hat{\tau}} \geq \mathcal{I}_\tau^{-1} \tag{4.31}$$

where $\mathcal{I}_\tau$ is the Fisher information for the random parameter τ, which for the problem under consideration takes the form:[2] [38]

$$\mathcal{I}_\tau = \dot{\boldsymbol{\mu}}^T \Sigma^{-1} \dot{\boldsymbol{\mu}} + \frac{1}{2} \operatorname{tr}\left[\Sigma^{-1}\dot{\Sigma}\Sigma^{-1}\dot{\Sigma}\right] \tag{4.32}$$

This expression may readily be simplified using the notation introduced in (4.22):

$$\mathcal{I}_\tau = \frac{ey^2 + 2by\dot{y} + a\dot{y}^2}{\sigma_s^2} + b^2 + ae \tag{4.33}$$

As before, it is desirable to express $\mathcal{I}_\tau$ without using the conditional covariance matrix Σ. To this end, we use the expressions (4.24) and (4.25) to obtain:

$$\mathcal{I}_\tau = \frac{b_0^2 y^2 + 2b_0 y\dot{y} + a_0\dot{y}^2}{\sigma_s^2(1-a_0)} + \frac{a_0 e_0}{1-a_0} + \frac{b_0^2(1+a_0)}{(1-a_0)^2} + \frac{e_0 y^2}{\sigma_s^2} \tag{4.34}$$

To remove the effect of the specific choice of τ on the bound (4.31), it is convenient to further average this expression over permissible τ in (4.31). How this is done depends on the application; for PWM and CM, although it is possible to consider only the case of E_2 and to average over all possible intervals $\mathcal{T}$ provided by E_1, it is much simpler and more useful to average over the entire interval C_j and provide a Cramér-Rao bound for both E_1 and E_2. To do this, we use the expression (4.11) for $f(\tau)$; we also average over the sampling delay T_d, assuming an uniform distribution over $[0, T_s]$. We then have:

$$\sigma_{\hat{\tau}}^2 \geq T_s \left[\int_0^{T_s} \int_{C_j} \mathcal{I}_\tau(\tau, t_d, y) f(\tau) d\tau dt_d\right]^{-1}. \tag{4.35}$$

4.5.2 UUB (Uniform Upper Bound)

The UUB reflects a lower bound on estimator performance which is attained by simply guessing a random location (in the permitted region $\mathcal{T}$) for the crossing. We generate $\hat{\tau}_{\text{UUB}}$ from an uniform distribution in $\mathcal{T}$. The variance of this estimate is $T_s^2/12$.

[2]The derivation for this expression and the simplified version below may be found in the appendices.

4.6 Comparison Estimators

We have now derived MAP and MMSE estimators for the discrete-time crossing-point estimation problem, as well as upper and lower bounds on performance that provide a convenient metric against which to measure estimators' performances in simulation. Because a number of solutions to the crossing-point estimation problem have been developed in the literature, we introduce several comparison estimators in the following sections.

We begin with two estimators based on Lagrange polynomial estimation. We assume familiarity with polynomial interpolation and Lagrange polynomials; background information can be found in [31, 39, 40]. Polynomial interpolation is widely used in computing the crossing-points of a signal and a sawtooth or triangle wave in PWM applications [15, 12, 7, 14, 8, 41, 16].

Following introduction of the polynomial crossing-point estimators, we present a degenerate case (designated ILIN), which is identical to one of the polynomial estimators (POLZ) when the number of samples is restricted to $M = 2$.

Each estimator introduced in this section is identified with a short, capitalized designation. These designators will be used in the sequel to compactly identify each estimator.

4.6.1 Lagrange Formulation

We adopt a vector formulation of Lagrange interpolation, which promotes the interpretation as a time-varying FIR filter. We define the M-dimensional column vector function $\mathbf{l}(\tau)$ componentwise as follows:

$$\mathbf{l}_m(\tau) = \prod_{\substack{n=k-M_1 \\ n \neq m}}^{k+M_2-1} \frac{\tau - nT_s - T_d}{(m-n)T_s} \qquad m = 0, \ldots, M-1 \tag{4.36}$$

Given a vector of samples $\mathbf{x}$ taken near the arbitrary time τ, an estimate of $s(\tau)$ can be generated via the inner product $\mathbf{x}^T\mathbf{l}(\tau)$. Likewise, $\boldsymbol{\xi}^T\mathbf{l}(\tau)$ provides an estimate of $z(\tau)$. In both cases, noise is neglected, and no information about the structure of $s(t)$ or $y(t)$ is exploited.

Many authors have noted that Lagrange polynomial interpolation is optimal in the sense that it minimizes approximation error at the frequency 0 (i.e. at DC) [39]. This property suggests that Lagrange interpolation of a sampled signal with substantial high-frequency

content will produce good results provided the sampling rate is much greater than the Nyquist rate for that signal. We will investigate the performance of Lagrange schemes later in Chapter 5.

We next introduce two polynomial estimators, denoted POLZ and POLS. POLZ interpolates $\boldsymbol{\xi}$ to estimate $z(\tau)$, and POLS interpolates $\mathbf{x}$ to estimate $s(\tau)$.

4.6.2 POLZ (Polynomial-z) Estimator

The POLZ estimate $\hat{\tau}_{\text{POLZ}}$ is defined as a solution (in $\mathcal{T}$) to

$$\boldsymbol{\xi}^T \mathbf{l}(\tau)\big|_{\tau=\hat{\tau}_{\text{POLZ}}} = 0 \tag{4.37}$$

The POLZ estimator may be visualized with the help of Figure 3.5. In this figure, the sample vector $\boldsymbol{\xi}$ is shown in a dashed box; this estimator fits a polynomial to these M samples surrounding the crossing point τ.

The structure of the POLZ estimator is illustrated in Algorithm 3.

Algorithm 3 Crossing-Point Estimation via the POLZ Algorithm

```
loop
  if ξ[k − 1]ξ[k] < 0 then
    ξ = [ξ[k − M1], ..., ξ[k + M2 − 1]]^T
    repeat
      refine guess τ̂ via Newton's or Brent's algorithms
      ε = |ξ^T l(τ̂)|
    until ε < tol
  end if
end loop
```

Because polynomial estimation performs poorly when the function being approximated exhibits discontinuities, the performance of POLZ should degrade for PWM schemes when $\boldsymbol{\xi}$ incorporates an edge of $y(t)$. However, in the case of PWM, we will shortly argue that POLZ and POLS may be made computationally identical with a little care.

4.6.3 POLS (Polynomial-s) Estimator

The POLS estimate $\hat{\tau}_{\text{POLS}}$ is defined as a solution (in $\mathcal{T}$) to

$$\mathbf{x}^T \mathbf{l}(\tau)\big|_{\tau=\hat{\tau}_{\text{POLS}}} = y(\tau)\big|_{\tau=\hat{\tau}_{\text{POLS}}} \tag{4.38}$$

The POLS estimator may be visualized with the help of Figure 3.4. In this figure, the sample vector $\mathbf{x}$ is shown in a dashed box; this estimator fits a polynomial to these M samples surrounding the crossing point τ.

The POLZ estimator may be more convenient to compute than the POLS estimator, since it does not require evaluations of $y(\tau)$ at arbitrary τ. However, as the polynomial of order $M-1$ passing through M points is unique, the POLZ and POLS estimators are mathematically identical when $y(\tau)$ is a polynomial. When $y(\tau)$ is not a polynomial, as in CM applications, the POLZ and POLS methods perform differently and it is important to distinguish between the two methods.

In the case of PWM, $y(\tau)$ is piecewise linear. Assuming the interval $\mathcal{T}$ does not straddle a discontinuity in $y(t)$, it is possible to simply replace $y(t)$ with a linear function that matches it precisely within $\mathcal{T}$. A POLZ estimator implemented in this fashion produces results identical to a POLS estimator, which suggests why no distinction between POLZ and POLS is present in the literature.

The structure of the POLS estimator is illustrated in Algorithm 4.

Algorithm 4 Crossing-Point Estimation via the POLS Algorithm

```
loop
  if ξ[k − 1]ξ[k] < 0 then
    x = [x[k − M1], ..., x[k + M2 − 1]]^T
    repeat
      refine guess τ̂ via Newton's or Brent's algorithms
      ε = |x^T l(τ̂) − y(τ̂)|
    until ε < tol
  end if
end loop
```

4.6.4 ILIN Estimator

We introduce a final estimator which we will designate ILIN. This is not actually a "new" estimator, since it is equivalent to POLZ when the number of samples M is equal to 2. We consider this special case because it is generally treated separately from higher-order polynomial methods, and since its solution need not be implicitly defined.

The computational structure of the ILIN estimator is shown in Algorithm 5.

Algorithm 5 Crossing-Point Estimation via the ILIN Algorithm

loop
  if $\xi[k-1]\xi[k] < 0$ **then**
    $\hat{\tau} = (k-1)T_s + T_d + \frac{T_s\xi[k-1]}{\xi[k-1]-\xi[k]}$
  end if
end loop

Note that this algorithm requires a division by a variable quantity $\xi[k-1]-\xi[k]$. In the MMSE, POLZ, and POLS algorithms above, divisions could be pre-computed as reciprocal multiplications; thus, in implementations (e.g. most DSPs) where divisions are expensive, they could be avoided. This ILIN algorithm avoids iterative calculation of the solution, at the expense of this unavoidable division.

4.7 Computational Complexity

We have now defined the MAP and MMSE estimators, the Cramér-Rao bound, and several reference estimators. In practice, the computational expense of each method is often as vital a consideration as its performance. For switching audio amplifiers in particular, crossing-points must be calculated as many as several million times per second; for this reason, before evaluating the performance of each, we consider their computational cost. (We may disqualify the MAP estimator outright on this condition.)

Complete crossing-point estimators using the POLZ, POLS, MMSEZ, and MMSE schemes share a very similar structure (see Algorithms 1, 2, 3, and 4.) First, each zero-crossing τ is coarsely located in the region $\mathcal{T}$ using the estimator E_1. Next, samples in the neighbourhood of $\mathcal{T}$ are passed to the refined estimator E_2 to generate a better estimate. The structure of E_2 is different for each scheme, but in each case, E_2 generates estimates $\hat{\tau}$ using an iterative root-finding process such as Newton's method or Brent's algorithm [36].

For the POLZ and POLS cases when $M = 1$ or $M = 2$, the crossing point may be calculated in closed form and the iterative step is not strictly necessary. However, $M = 1$ is a degenerate case in which E_2 may be entirely neglected and $M = 2$, corresponding to straight-line interpolation, requires division operations which may be costly and involve an iterative computation anyway. We neglect these details in the following discussion.

We define the following quantities:

μ is the cost (in elementary operations) of evaluating a scalar nonlinear function. For example, μ is the cost of evaluating $y(\tau)$, and $M\mu$ is the cost of evaluating $\rho(\tau)$;

λ is the probability (per sample of $x[k]$) that E_1 will detect a crossing point and require an evaluation of E_2;

k is the number of iterations of the root-finding algorithm (e.g. Brent's method). (For simplicity, we assume that each root-finding iteration k evaluates the relevant functions only at a single time instant τ.)

Now, the computational cost for each scheme (in elementary floating-point operations per second) may be evaluated as follows:

$$C = f_s(\mu + 1) + f_s\lambda(\ldots) \tag{4.39}$$

where f_s is the sampling rate. The first term $f_s(\mu + 1)$ is the cost of the coarse estimator E_1 and is shared by all the schemes we investigate. The second cost, which depends on the estimator used, is more interesting and is the subject of the subsections that follow.

4.7.1 Cost of MMSE

Referring to Algorithm 1, we note that the Cholesky factorization L of $\Sigma = LL^T$ may be precomputed. Therefore, we may compute the vector $\mathbf{w} = \Sigma^{-1}\mathbf{x}$ by foreward- and back-substitution at a total cost of $2M^2$.

The cost C_{MMSE} may be expressed as follows:

$$C_{\text{MMSE}} = f_s(\mu + 1) + f_s\lambda(2M^2 + k[M\mu + 2M + \mu]). \tag{4.40}$$

4.7.2 Cost of MMSEZ

The algorithm for MMSEZ (Algorithm 2) is nearly identical to the algorithm for MMSE. The primary difference is the reduced number of evaluations of $y(\tau)$. We may write C_{MMSEZ} directly:

$$C_{\text{MMSEZ}} = f_s(\mu + 1) + f_s\lambda(2M^2 + k[M\mu + 2M - 1]) \tag{4.41}$$

reflecting a minor savings in computations with respect to the MMSE algorithm.

4.7.3 Cost of POLS

Both Lagrange estimators make use of the vector-valued function $\mathbf{l}(\tau)$. Although we know the form of this function, we regard it as an arbitrary non-linear function with evaluation cost $M\mu$ in keeping with the cost analysis of the MMSE and MMSEZ estimators.

The cost of the POLS scheme is therefore

$$C_{\text{POLS}} = f_s(\mu + 1) + f_s\lambda k[M\mu + 2M + \mu]. \tag{4.42}$$

4.7.4 Cost of POLZ

The cost of the POLZ scheme is

$$C_{\text{POLZ}} = f_s(\mu + 1) + f_s\lambda k[M\mu + 2M - 1]. \tag{4.43}$$

The POLZ estimator enjoys a slightly lower computation cost compared with the POLS estimator.

4.7.5 Remarks on Cost

Two factors determine the relative cost of each scheme: the number of samples M, and the cost μ of evaluating a nonlinear function. Because the number of samples is likely to be low (typically 4-8), the expense of functional evaluation dominates. Each of $\mathbf{l}(\tau)$, $\boldsymbol{\rho}(\tau)$, and $y(\tau)$ have varying complexity depending on their form, the technology used to implement them, and the approximations or computational shortcuts that are often used in practice. As we have focused on the general crossing-point estimation problem, we do not consider a particular implementation in sufficient detail to make a definitive statement concerning complexity.

We may, however, draw the following conclusions:

- POLZ is only slightly simpler than POLS, and
- MMSEZ is only slightly simpler than MMSE; finally,
- MMSE and MMSEZ are more expensive than POLS and POLZ.

As we will see in Chapter 5, MMSE and POLS outperform MMSEZ and POLZ, suggesting that MMSE and POLS are the chief methods of interest.

Chapter 5

Performance Simulation

In the previous chapter, we derived the MAP and MMSE estimators and the associated CRB. We have also reviewed the MMSEZ, POLZ, POLS, and ILIN crossing-point estimators for the purpose of comparing their performance. In this chapter, we present simulations in which the performance of these methods are evaluated and characterized as a function of various system parameters.

5.1 Scenario

In the following subsections, we introduce the two scenarios of interest: one for Pulse-Width Modulation, and one for Click Modulation. The chief difference between these scenarios is the form of the carrier function $y(t)$. The results for each scenario are comparable, though they differ in some interesting ways.

5.1.1 Operating Point Parameters

We explore the effects of varying a number of parameters. In order to limit the number of possible parameter variations, we hold each at the operating point in Table 5.1 and alter one parameter at a time.

The parameters $f_s = 1/T_s$, M, A_c, σ_n^2 and σ_s^2 are described in the preceding chapters. We have introduced the parameter Ω_M which corresponds to the bandlimit of $s(t)$, which we assume is strictly low-pass in nature.

We have chosen a very high signal-to-noise ratio (SNR) σ_s/σ_n. In open-loop, switching audio amplifiers using a high-quality digital audio source (such as CD or DVD audio), the

Parameter	Value	Description
$f_s = 1/T_s$	192 kHz	sampling rate
$\Omega_M/2\pi$	24 kHz	bandlimit of $s(t)$
M	4	number of samples ($M_1 = M_2 = 2$)
A_c	1	carrier amplitude
σ_n^2	$(2^{-15})^2/12$	noise variance
σ_s^2	$(1/8)^2$	signal variance

Table 5.1 Operating Point Parameters

sole source of noise is due to finite word length. A value of 16 bits is typical, corresponding to the noise variance $\sigma_n^2 = (2^{-15})^2/12$. In a PWM amplifier, the input samples $x[k]$ correspond directly to audio samples, and this choice for σ_n^2 is obvious. In a CM system, a substantial amount of signal processing must occur to transform the audio signal into the $x[k]$ applied to a crossing-point estimator. We neglect the possibility that this signal-processing injects significant additional noise, and use the same σ_n^2 for the CM scenario as well.

The signal variance is $(1/8)^2$. This figure is chosen somewhat arbitrarily to maximize the SNR without a substantial probability of modulator overload (i.e. $|s(t)| \geq A_c$.)

5.1.2 Pulse-Width Modulation

For Pulse-Width Modulation (PWM), we adopt a triangle-wave carrier $y(t)$ with amplitude A_c and period $T_c = 2\pi/\Omega_M$. The signal $s(t)$ is bandlimited to the same angular frequency Ω_M.

Energy in a typical audio signal ranges from 20Hz to 20kHz, though it is usually concentrated within the low end of the audible spectrum. There is additional structure in an audio signal due to other factors; for example, psychoacoustic properties of human hearing are often exploited to increase perceived quality (viz. noise shaping) or decrease storage requirements (viz. MP3 audio). A flat, strictly bandlimited model neglects all of these details, though it is not obvious how a richer model would be useful.

5.1.3 Click Modulation

For Click Modulation (CM), we adopt a sinusoidal carrier $y(t) = A\cos(\Omega_m t + \theta)$. The signal $s(t)$ is bandlimited to the same (angular) frequency Ω_m.

A spectrally flat, bandlimited signal model is an obvious choice for the CM scenario. The

upper comparator in Figure 2.6 corresponds to a crossing-point estimator in a discrete-time implementation. (The lower comparator processes the deterministic input $-\sin(ct+\theta)$, and therefore needs no estimator.) The input $s(t)$ to the upper estimator may be expressed as the sum of a deterministic, sinusoidal component $\cos ct$ and a bandlimited unknown component $g(t)$ with a maximum bandwidth $b > c$ (see [9, Eq. 5].) The unknown component $g(t)$ has significant spectral content near b, particularly for audio signals with substantial low-frequency content. However, the steps that transform audio into $g(t)$ obscure much of the structure in the original audio signal.

5.2 Methodology

In this section, we describe the synthesis models used for each of the signals required, and explain the methodology used to conduct the simulations.

5.2.1 Signal Model

For both PWM and CM scenarios, $s(t)$ is modeled as a wide-wense stationary Gaussian random process with zero mean and flat, bandlimited spectral characteristics. That is, the power spectral density (psd) of $s(t)$ is given by

$$P_s(\omega) = \begin{cases} \sigma_s^2, & |\omega| < \Omega_M \\ 0, & \text{otherwise.} \end{cases} \tag{5.1}$$

Accordingly, the autocorrelation function of $s(t)$ may be expressed as: [42]

$$r_s(t) = \sigma_s^2 \operatorname{sinc}(\Omega_m t/\pi). \tag{5.2}$$

Note that this expression for $r_s(t)$ is used to evaluate $\rho(\tau)$ within the MAP, MMSE and MMSEZ estimators.

During computation, in addition to the samples of $s(t)$ required to form the vectors $\mathbf{x}$ and $\boldsymbol{\xi}$ in (3.5), we require precise knowledge of each of the true crossing points of $s(t)$ and $y(t)$ in order to determine the error for each estimator. To determine these crossing points with high precision, it is necessary to generate $s(t)$ in such a way that it may be evaluated at arbitrary time instants with reasonable computational requirements.

We generate a continuous function $s(t)$ which is parameterized by a collection of random

variables in two ways: the Karhunen-Loève expansion (KLE) [42], and an ad-hoc approach using a sum of sinusoidal signals. The KLE approach is more attractive from a theoretical perspective for reasons that are explained below, but is slow during simulation. The sum-of-sines approach is much more practical to simulate, and results generated by these methods are indistinguishable.

Signal Generation using the Karhunen-Loève Expansion

We begin with the Karhunen-Loève expansion for the bandlimited signal $s(t)$:

$$s(t) = \underset{N\to\infty}{\text{l.i.m.}} \sum_{n=1}^{N} s_n \psi_n(t) \quad t_0 \le t \le t_{M-1} \tag{5.3}$$

where l.i.m. represents the limit in the mean-squared sense and $[t_0, t_{M-1}]$ is the observation interval. For strictly bandlimited spectra, the functions $\psi_n(t)$ are scaled Prolate Spheroidal Wave Functions (PSWFs) [33]. The corresponding s_n are uncorrelated, zero-mean Gaussian random variables with variance λ_n, where λ_n are parameter-dependent eigenvalues associated with each eigenfunction $\psi_n(t)$. Theoretical background concerning the PSWFs themselves may be found in [33, 43, 44].

In practice, the expected energy λ_n associated with each eigenfunction within the observation interval decreases rapidly as n increases, and the summation in (5.3) may be truncated. We truncate the summation when additional terms contribute less than 10^{-6} of the total energy in $s(t)$ over the observation interval. Depending on the scenario being simulated, this requires between 6 and 10 summation terms.

Signal Generation using a Sum of Sinusoids

The KLE is an optimal representation for bandlimited functions in the sense that it maximizes the energy in the first N terms of (5.3); thus, an expansion with finite N using the PSWFs as basis functions exhibits less distortion than any other choice of basis functions. However, the PSWFs and corresponding eigenvalues are slow to calculate in practice. We introduce an alternative approach which is not as theoretically pleasing, but which produces indistinguishable results in simulation and is much faster to calculate.

We use the following signal model:

$$s(t) = \sum_{n=1}^{N} \sqrt{2\sigma_s^2/N} \sin(\omega_n t + \phi_n) \quad t_0 \le t \le t_{M-1} \tag{5.4}$$

where ω_n and ϕ_n are uniform random variables, all independent, over the (respective) intervals $[0, \Omega_M]$ and $[0, 2\pi]$. We use $N = 10$ terms in the summation. The various RVs $\{\omega_n\}_{n=1}^N$, $\{\phi_n\}_{n=1}^N$ are independently generated.

5.2.2 Monte Carlo Simulations

For each simulation, short segments of $s(t)$ are generated randomly using one of the above approaches. In order to minimize the number of non-negligible components N required for the Karhunen-Loève expansion (5.3), these segments are of minimum length, i.e. the observation interval $(M-1)T_s$. This allows us to evaluate $s(t)$ precisely at any point within the observation interval.

Next, the carrier phase θ of $y(t)$ is chosen randomly. Along with $s(t)$, this completely specifies $y(t)$ and $z(t) = s(t) - y(t)$ for each experiment. We sample $s(t)$ and $z(t)$ and add noise, forming the sample vectors $\mathbf{x}$ and $\boldsymbol{\xi}$.

Before passing these vectors to each estimator, two conditions must be met:

- $z(t)$ must have a crossing point in $\mathcal{T} \cap \mathcal{C}_j$ (the sampling and carrier intervals defined in (3.4) and (3.6) respectively);
- this crossing point must be observable by the coarse estimator E_1 by looking at the M_1th and (M_1+1)th samples of $\boldsymbol{\xi}$.[1]

Candidate scenarios that do not meet these two conditions are discarded.

Finally, each estimator is evaluated in turn. Because each estimator takes an iterative form, some starting estimate must be provided; except for the impractical MAP estimator, we select the center of $\mathcal{T}$.

Solving for the MAP estimate presents practical difficulties for the reasons explained in Section 4.3. The MAP estimate is defined implicitly in (4.27). However, this expression is ambiguous when multiple solutions exist in $\mathcal{T}$, since one estimate must be chosen from amongst several candidates. (This ambiguity also exists for the other estimators with implicitly defined solutions. However, only the MAP estimator exhibits multiple candidates in practice.) As the SNR and oversampling rate are increased, the same process that allowed the MAP estimator to be approximated by the MMSE estimator renders the MAP estimate difficult to locate: as the MMSE and MAP estimators converge, the two nearby solutions

[1] These samples correspond to $\xi[k-1]$ and $\xi[k]$, which are used by E_1 to determine where E_2 is applied.

to (4.27) corresponding to $(y - c0) = 0$ and $(a_0 y - c0) = 0$ approach a single solution $(y - c0)^2 = 0$, for which the simple root-finding algorithm we employ (MATLAB's `fzero` function) encounters difficulty.[2] To avoid the problem, we obtain MAP estimates using the following process:

- we use the MMSE estimate as an initial guess, and
- instead of applying a root-finding method to (4.9), we minimize $|S(\mathbf{x}|\tau) + S(\tau)|^2$.

Although more circuitous, this method of obtaining MAP estimates is more robust in practice.

5.2.3 Removal of Outliers

There are two types of outliers produced in simulations:

- individual experiments which produce results outside $\mathcal{T}$, and
- individual experiments with an estimation error that is disproportionately larger than other experiments using the same parameters.

The first class of outliers is readily identifiable in situ, since the estimate $\hat{\tau}$ and the bounding region $\mathcal{T}$ are known. When such an outlier occurs, the estimator may fall back on an alternative estimator, or may choose to replace the invalid crossing-point estimate with the nearest valid crossing point (i.e. an endpoint of $\mathcal{T}$.) It should be noted that when the SNR is low, crossing points may actually occur outside the region $\mathcal{T}$ identified by the coarse estimator E_1; in these situations, valid experimental results are incorrectly flagged as outliers.

The second class of outliers must be identified by statistical analysis of an ensemble of experimental results, given exact knowledge of where each crossing point actually occurred. These outliers, which are typically due to numerical errors in pathological cases, cannot be identified in situ and are not flagged or removed in simulation.

In most scenarios, no outliers were detected or removed. Where outliers occurred, the number of experiments which produced them are listed along with the performance figures below.

[2]The function whose roots we wish to obtain has two very closely spaced solutions, or in the limit, a single root with multiplicity 2. Many numerical root-finding algorithms are known to perform poorly or even fail with close or multiple roots. [36]

5.2.4 Reference Estimators

In addition to the MAP and MMSE estimators corresponding to (4.27) and (4.30), and the Cramér-Rao bound given in (4.31) and (4.34), we evaluate a number of alternative estimators. These estimators are taken from the literature and are useful for comparison.

UUB Uniform Upper-Bound; see Section 4.5.2.

ILIN Linear interpolation between the nearest samples of $\xi[k]$ using Algorithm 5. This method is a degenerate case of POLZ when $M = 2$. See e.g. [7] and Section 4.6.1 above.

POLZ Solves $\hat{\xi}(t) = 0$ using Algorithm 3.

POLS Solves $\hat{x}(t) = y(t)$ using Algorithm 4.

MMSEZ Approximation to the MMSE estimator using Algorithm 2.

MMSE Approximation to the MAP estimator using Algorithm 1.

MAP MAP estimator using (4.27) where $f(\tau)$ is given by (4.11).

CRB Cramér-Rao Bound; see Section 4.5.1.

5.3 Results

We present results for two scenarios: click modulation (using a sinusoidal carrier) and pulse-width modulation (using a triangular carrier.) Each data point on the following graphs represents a statistical analysis of approximately 20,000 experiments.

5.3.1 Click Modulation

Figure 5.1 shows the estimator performance as the number of samples is varied over the even numbers between 2 and 10. All estimators except ILIN (which uses 2 samples in all cases), along with the CRB, rapidly approach a limit beyond which more samples do not improve performance. This behaviour reflects the fact that additional samples are unable to meaningfully improve performance because they are increasingly distant from the crossing-point, and are not substantially correlated with $s(\tau)$. The percentage of outliers associated

with the data in Figure 5.1 are shown in Table 5.2. Note that whenever the estimated bias approaches 0, small variations (which could be due to genuine estimator bias or numerical errors, for example) cause large visual differences in the log-scale bias graphs; this effect should not be attributed to estimator behaviour.

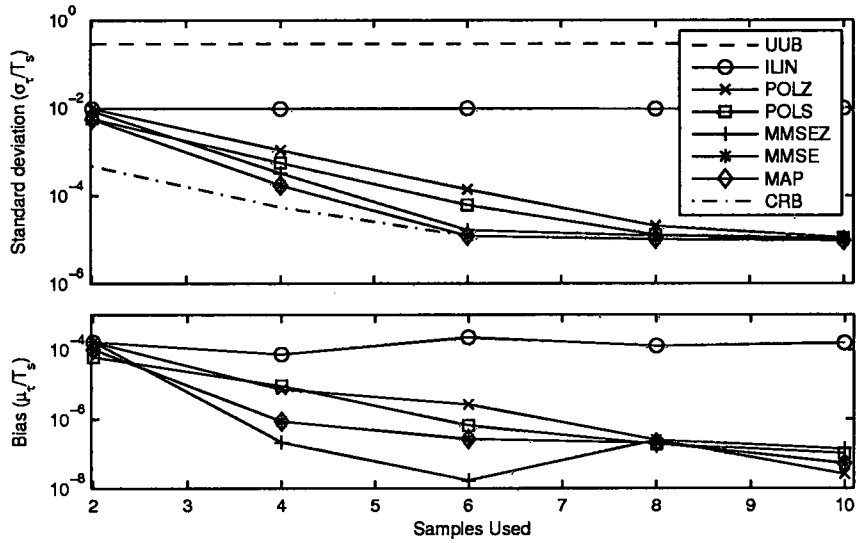


Fig. 5.1 Estimator Performance (CM); Number of Samples M Varies

M	MMSE	MMSEZ	MAP	POLS	POLZ
2	–	–	$\ll 0.1\%$	–	–

Table 5.2 Outliers associated with Fig. 5.1

Figure 5.2 shows the estimator performance as a function of the oversampling ratio. The horizontal axis is normalized to the sampling rate so that 1 corresponds to sampling $s(t)$ at its Nyquist rate. Critical (Nyquist-rate) sampling is excluded; at such a low oversampling ratio, crossing points are not suitably separated and the assumptions of our two-step estimator are not valid. Each method improves as the oversampling ratio is increased, although the ILIN estimator does not improve as rapidly as the other methods.

Where outliers associated with the data in Figure 5.2 occurred, they are tabulated in Table 5.3.

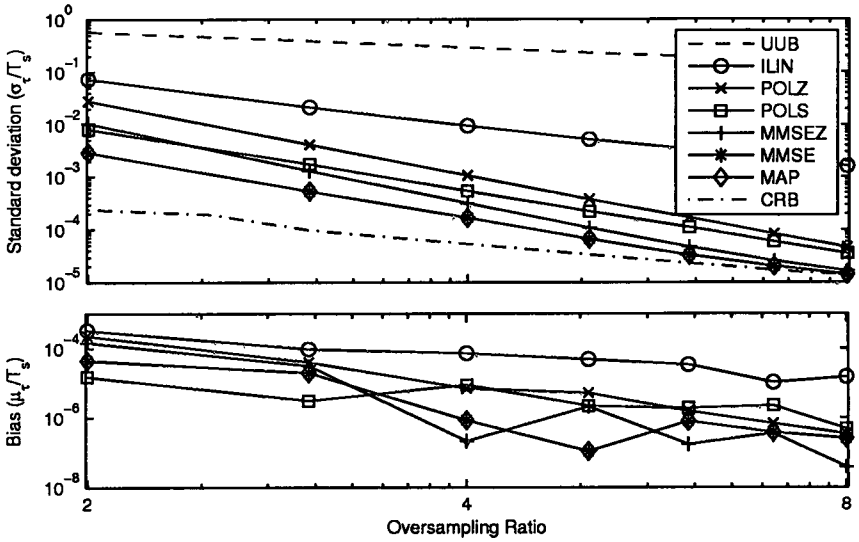


Fig. 5.2 Estimator Performance (CM); Oversampling Ratio Varies

OSR	MMSE	MMSEZ	MAP	POLS	POLZ
4	–	–	$\ll 0.1\%$	–	–

Table 5.3 Outliers associated with Fig. 5.2

Figure 5.3 shows the estimator performance as the signal-to-noise ratio (σ_s/σ_n, shown in dB) is varied. For high-fidelity switching amplifiers, the SNR may be well over 96 dB. In the high-SNR regime, when high accuracy is required, the MAP and MMSE estimators have a distinct advantage over the MMSEZ, POLS, POLZ, and ILIN estimators. At low SNRs, the MMSE and MAP estimators produced results flagged as outliers at a rate of several percent; no other estimators produce outliers (see Table 5.4.) In addition, at low SNRs, scenarios with crossing-points not detected by E_1 are simply discarded. This "scenario filtering" process removes predominantly difficult (noisy) scenarios.

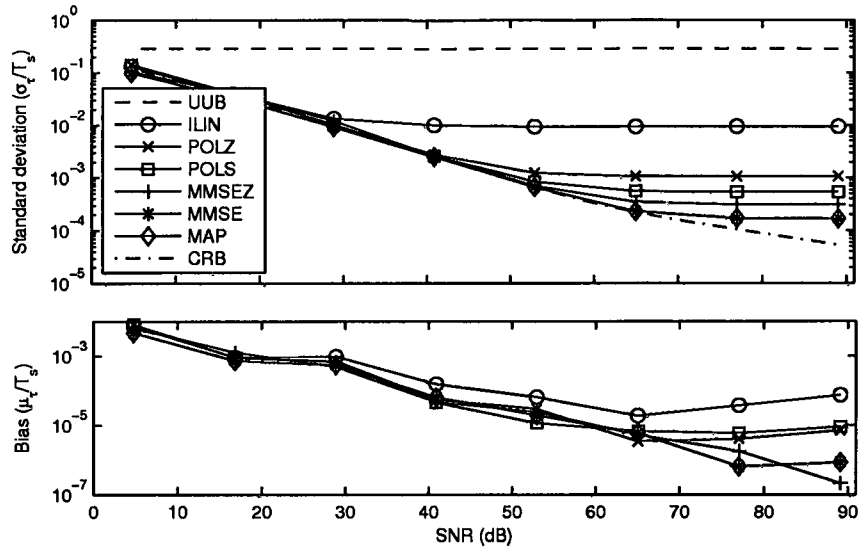


Fig. 5.3 Estimator Performance (CM); SNR Varies

SNR	MMSE	MMSEZ	MAP	POLS	POLZ
5dB	4.0%	3.4%	3.9%	$\ll 0.1\%$	$\ll 0.1\%$
17dB	0.8%	3.1%	0.8%	–	–
29dB	0.1%	1.5%	0.1%	–	–
41dB	$\ll 0.1\%$	0.1%	$\ll 0.1\%$	–	–
53dB	$\ll 0.1\%$	$\ll 0.1\%$	$\ll 0.1\%$	–	–

Table 5.4 Outliers associated with Fig. 5.3

Figure 5.4 shows the estimator performance as the carrier amplitude (A_c) is varied. A large carrier amplitude, combined with a sinusoidal carrier, decreases the ability of the POLZ and MMSEZ estimators to approximate the carrier with a straight line. Correspondingly, the performance of the POLZ and MMSEZ schemes diverge from the POLS and MMSE schemes as A_c is increased. When the amplitude is low with respect to the signal standard deviation, it is not guaranteed that zero crossings are adequately separated and our model may not be accurate.

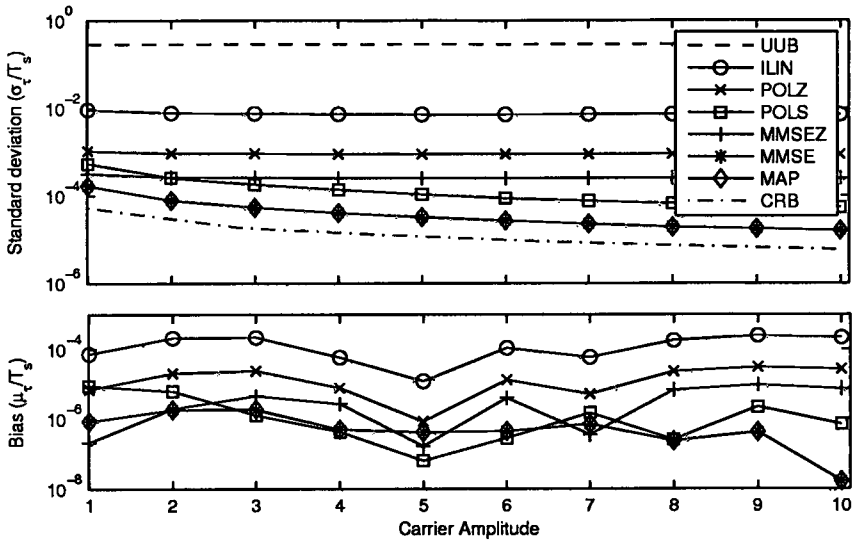


Fig. 5.4 Estimator Performance (CM); Carrier Amplitude Varies

5.3.2 Pulse-Width Modulation

We now review the performance of each scheme in a PWM scenario. As these results are often similar to the CM scenario described above, we focus on the differences between the two scenarios.

Figure 5.5 shows the estimator performance as the number of samples M is varied. This result shows the performance of the POLZ and MMSEZ schemes *decreasing* as additional samples are supplied to the estimator. In both cases, this effect reflects the inability of a continuous interpolator to accurately model a discontinuous $y(t)$. As the length M of the sample vector $\boldsymbol{\xi}$ is increased, the probability that it will contain a discontinuity in $y(t)$ increases. In the case of POLZ, the interpolator is polynomial, and therefore continuous. With MMSEZ, the interpolator is strictly bandlimited, and is unable to model a discontinuous $y(t)$ which has infinite spectral support. (The relationship between polynomial and bandlimited interpolation is well documented in [31, 40].) Outliers associated with the data in Figure 5.5 are tabulated in Table 5.5.

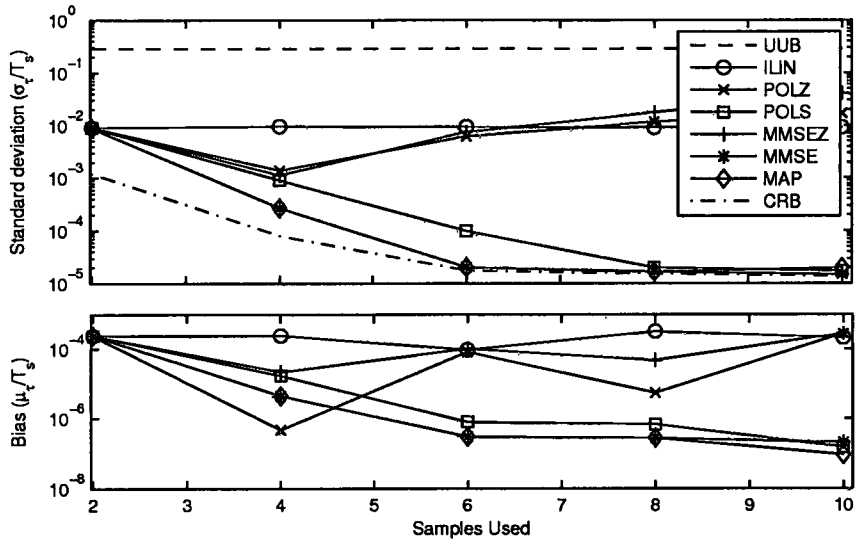


Fig. 5.5 Estimator Performance (PWM); Number of Samples M Varies

M	MMSE	MMSEZ	MAP	POLS	POLZ
10	$\ll 0.1\%$	0.3%	$\ll 0.1\%$	–	–
8	–	$\ll 0.1\%$	–	–	–
2	–	–	$\ll 0.1\%$	–	–

Table 5.5 Outliers associated Fig. 5.5

Figure 5.6 shows the estimator performance as a function of the oversampling rate. Once again, the horizontal axis is normalized to the sampling rate so that 1 corresponds to critical sampling. As in Figure 5.5, the probability that ξ contains a nonlinearity strongly affects the performance of the POLZ and MMSEZ estimators; when the oversampling ratio is large, this probability is negligible and the performance of POLZ and MMSEZ approach that of POLS and MMSE.

Figure 5.7 shows the estimator performance as the signal-to-noise ratio (σ_s/σ_n, shown in dB) is varied. In contrast to the above scenario where M was varied, we can see the performance of the POLZ estimator converges to that of POLS as the oversampling ratio is increased. Likewise, the performance of MMSEZ converges to that of MMSE. As noted above, the ability of any estimators to perform below the CRB is due to an unfair "scenario filtering" that occurs when the coarse estimator E_1 is not able to reliably detect crossing-

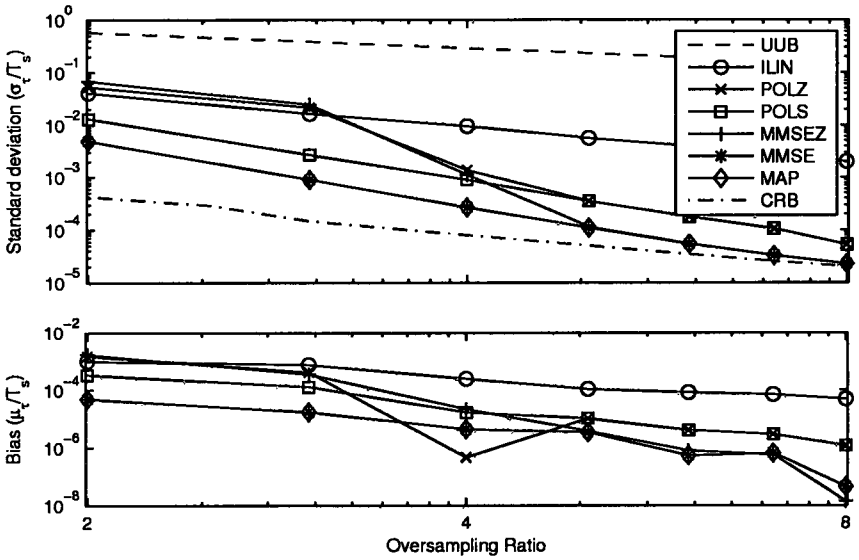


Fig. 5.6 Estimator Performance (PWM); Oversampling Ratio Varies

points during simulations.

Where outliers associated with the data in Figure 5.7 occurred, they are tabulated in Table 5.6.

SNR	MMSE	MMSEZ	MAP	POLS	POLZ
5dB	7.0%	5.3%	7.0%	0.1%	0.1%
17dB	1.3%	0.8%	1.3%	–	–
29dB	$\ll 0.1\%$	$\ll 0.1\%$	$\ll 0.1\%$	–	–
41dB	$\ll 0.1\%$	$\ll 0.1\%$	$\ll 0.1\%$	–	–
53dB	$\ll 0.1\%$	$\ll 0.1\%$	$\ll 0.1\%$	–	–

Table 5.6 Outliers associated Fig. 5.7

Figure 5.8 shows the estimator performance as the carrier amplitude (A_c) is varied. Compared to the results for a sinusoidal carrier (see Figure 5.4), the performance of the ILIN estimator improves more rapidly when the carrier is triangular. This effect reflects the fact that when the carrier amplitude is significantly larger than the signal amplitude, the signal $z(t)$ being linearly interpolated is predominantly linear.

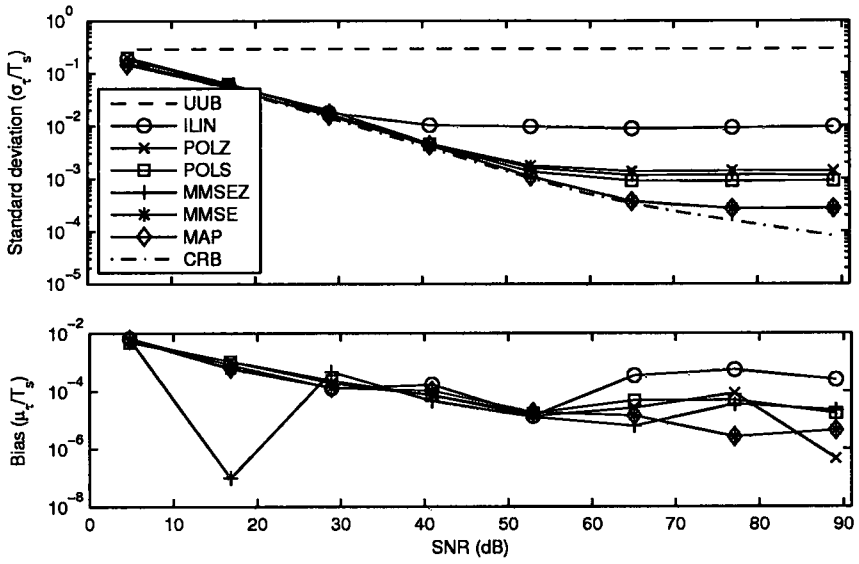


Fig. 5.7 Estimator Performance (PWM); SNR Varies

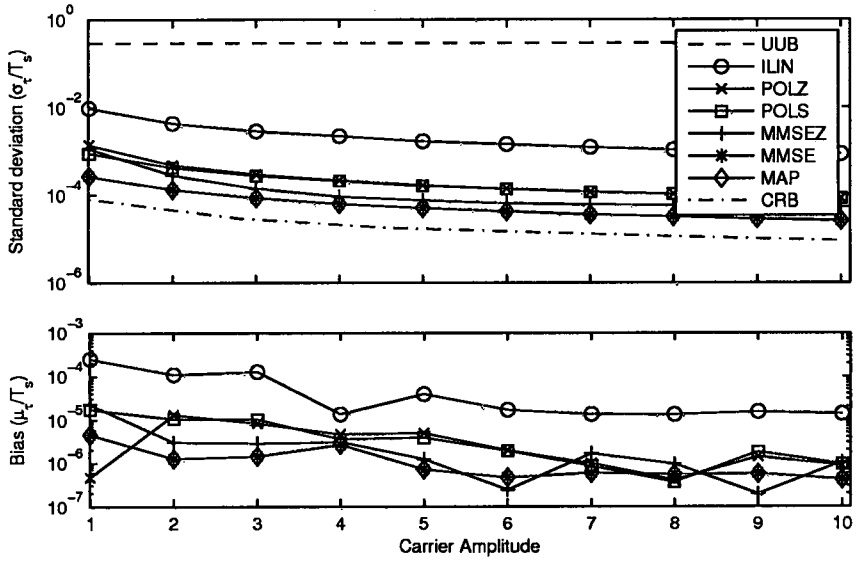


Fig. 5.8 Estimator Performance (PWM); Carrier Amplitude Varies

5.3.3 Discussion

We now draw some conclusions from the simulations presented in the previous subsections.

The performance of the MMSE estimator was, in all simulations conducted, indistinguishable from that of the MAP estimator. This result is critical, since the MAP estimator is computationally expensive enough to render it useless in resource-constrained, real-time applications. The ability of the MMSE estimator to accurately and efficiently approximate the MAP estimate provides us with a practical and effective estimator.

In simulations, the polynomial estimators POLS and POLZ did not always approach the CRB. In these cases, the MMSE estimator performed substantially better than the nearest polynomial scheme. It is worth noting that the operating point described in Table 5.1, chosen to model a practical scenario for switching audio amplification, is one such scenario.

Simulations also exposed the limitations of the POLZ and MMSEZ estimators. Not only do these estimators perform poorly when they encounter discontinuities in $y(t)$, their performance when compared to POLS and MMSE (respectively) is often reduced. The appeal of MMSEZ and POLZ is diminished considering that their estimates are only slightly simpler to compute than POLS and MMSE estimates.

Chapter 6

Conclusions

6.1 Thesis Review

In this thesis, we explored the discrete-time crossing-point estimation problem. This problem arises in a number of applications, including discrete-time modulator design for switching amplifiers. We reviewed the most common approach to the problem, based on a two-step estimation process involving a coarse estimator E_1 followed by a refined estimator E_2.

We then reviewed two choices for the refined estimator E_2 based on polynomial interpolation. These estimators, denoted POLS and POLZ, were selected from the literature. Most publications focus on an application (PWM) in which the distinction between POLS and POLZ is not crucial; in our analysis, the differences between the schemes was emphasized.

We introduced a statistical formulation for crossing-point estimation, which led to a MAP estimator and the Cramér-Rao bound for the problem. Because the MAP estimator was too complex to be useful in resource-constrained, real-time applications, we simplified the MAP estimator, producing a computationally efficient MMSE estimator,

We presented simulations for both Pulse-Width Modulation and Click Modulation scenarios. In all cases, the MMSE estimator closely tracked the MAP estimator and outperformed the MMSEZ, POLS, POLZ and ILIN estimators.

This work is important for three primary reasons: firstly, it casts an outstanding problem in a rigorous statistical framework, and provides the means with which to fairly evaluate competing estimators. Secondly, it introduces the MMSE estimator and shows it to perform very near the Cramér-Rao bound, suggesting there is not much room to improve on the scheme from the perspective of estimation accuracy. Finally, the MMSE estimator

substantially outperforms the alternative estimators in a scenario chosen to model high-performance, switching audio amplification.

6.2 Future Work

The MMSE estimator has an algorithmic structure that is nearly identical to the POLZ and POLS estimators, although it requires potentially expensive evaluation of the vector correlation function. The task of implementing the MMSE algorithm is greatly dependent on the computational resources that silicon can provide. However, there are only a few broad classes of implementation device (e.g. FPGA, DSP, general-purpose CPU, or custom silicon) available to designers, and it would be worthwhile to investigate balancing accuracy with computational efficiency for one or several of these.

References

[1] B. Kedem, "Spectral analysis and discrimination by zero-crossings," *Proc. IEEE*, vol. 74, pp. 1477–1493, Nov. 1986.

[2] T. Scholand, A. Burnic, C. Spiegel, A. Waadt, A. Hessamian-Alinejad, and P. Jung, "Applying zero-crossing demodulation to the Bluetooth enhanced data rate mode," in *Proc. IEEE Personal, Indoor and Mobile Radio Comm.*, Sept. 2006, pp. 1–5.

[3] G. E. Mog and E. P. Ribeiro, "Zero crossing determination by linear interpolation of sampled sinusoidal signals," in *Transmission and Distribution Conf. and Exposition: Latin America, 2004 IEEE/PES*, Nov. 2004, pp. 799–802.

[4] I. Blake and W. Lindsey, "Level-crossing problems for random processes," *IEEE Trans. Inf. Theory*, vol. 19, pp. 295–315, May 1973.

[5] K. M. Guan and A. C. Singer, "Opportunistic sampling by level-crossing," in *IEEE Int. Conf. on Acoustics, Speech and Signal Processing*, Apr. 2007, vol. 3, pp. 1513–1516.

[6] A. J. Rainal, "Origin of Rice's formula," *IEEE Trans. Inf. Theory*, vol. 34, pp. 1383–1387, Nov. 1988.

[7] C. Pascual and P. D. Krein, "High-fidelity PWM inverter for digital audio amplification based on real-time DSP," *IEEE Trans. Power Electron.*, vol. 18, pp. 473–485, Jan. 2003.

[8] V. Shimanskiy and S.-C. Jang, "Iterative methods for natural sampling," in *Proc. Audio Eng. Soc. 121st Conv.*, San Francisco, CA, Oct. 2006.

[9] B. F. Logan, "Click modulation," *AT&T Bell Labs Tech. J.*, vol. 63, pp. 401, Mar. 1984.

[10] I. Bar-David, "An implicit sampling theorem for bounded bandlimited functions," *Information and Control*, vol. 24, pp. 36–44, Jan. 1974.

[11] J. M. Goldberg and M. B. Sandler, "New high accuracy pulse width modulation based digital-to-analogue converter/power amplifier," *IEE Proc. Circuits Devices Syst.*, vol. 141, pp. 315–324, Aug. 1994.

[12] B.-H. Gwee, J. S. Chang, and V. Adrian, "A micropower low-distortion digital class-D amplifier based on an algorithmic pulsewidth modulator," *IEEE Trans. Circuits Syst. I*, vol. 52, pp. 2007–2022, Oct. 2005.

[13] E. Bresch and W. T. Padgett, "TMS320C67-based design of a digital audio power amplifier introducing a novel feedback strategy," Tech. Rep., Rose-Hulman Institute of Technology, 2002.

[14] K. Nguyen and D. Sarwate, "Up-sampling and natural sample value computation for digital pulse width modulators," in *Proc. 40th Ann. Conf. on Info. Sci. and Syst.*, Mar. 2006, pp. 1096–1101.

[15] P. Midya, B. Roeckner, P. Rakers, and P. Wagh, "Prediction correction algorithm for natural pulse width modulation," in *Proc. Audio Eng. Soc. 109th Conv.*, Los Angeles, CA, Sept. 2000.

[16] C. Pascual and B. Roeckner, "Computationally efficient conversion from pulse-code-modulation to naturally-sampled pulse-width-modulation," in *Proc. Audio Eng. Soc. 109th Conv.*, Los Angeles, CA, Sept. 2000.

[17] P. Craven, "Toward the 24-bit DAC: Novel noise-shaping topologies incorporating correction for the nonlinearity in a PWM output stage," *J. Audio Eng. Soc.*, vol. 41, pp. 291–313, May 1993.

[18] A. Floros and J. Mourjopoulos, "Distortion-free 1-bit PWM coding for digital audio signals," *EURASIP Journal on Advances in Signal Processing*, vol. 2007, pp. 1–12, Jan. 2007.

[19] J. Andersen, D. Chieng, S. Harris, J. Klaas, M. Kost, and S. Taylor, "Second generation intelligent class D amplifier controller integrated circuit enables both low cost and high performance amplifier designs," in *Proc. Audio Eng. Soc. 129th Conv.*, Paris, France, May 2006.

[20] S. Santi, R. Rovatti, and G. Setti, "Spectral aliasing effects of PWM signals with time-quantized switching instants," in *Proc. Int. Conf. on Circ. and Syst.*, May 2004, vol. 4, pp. 689–92.

[21] A. S. Sedra and K. C. Smith, *Microelectronic Circuits*, Oxford University Press, Inc., New York, 4th edition, 1998.

[22] H. S. Black, *Modulation Theory*, D. Van Norstrand Company, Inc., 1953.

[23] A. V. Oppenheim, R. W. Schafer, and J. R. Buck, *Discrete-Time Signal Processing*, Prentice-Hall, 2nd edition, 1999.

[24] Z. Song and D. V. Sarwate, "The frequency spectrum of pulse width modulated signals," *Signal Processing*, vol. 83, pp. 2227–2258, Oct. 2003.

[25] D. Maksimovic, L. Marco, A. Poveda, and E. Alarcón, "Bandwidth limits in PWM switching amplifiers," in *Proc. 2006 IEEE Int. Symp. on Circ. and Syst.*, May 2006.

[26] B. Wilson, Z. Ghassemlooy, and A. Lok, "Spectral structure of multitone pulse width modulation," in *Electronics Letters*, Apr. 1991, vol. 27, pp. 702–704.

[27] P. Wagh, "Closed-form spectral analysis of pulse-width modulation," in *2001 IEEE Int. Symp. on Circuits and Systems*, 6-9 May 2001, vol. 3, pp. 799–802.

[28] S. Haykin, *Communications Systems*, John Wiley & Sons, 4th edition, 2001.

[29] M. Streitenberger, F. Felgenhauer, H. Bresch, and W. Mathis, "Zero-position coding with separated baseband in low-power class-D audio amplifiers for mobile communications," in *5th Int. Conf. on Telecom. in Modern Satellite, Cable and Broadcasting Service*, 19–21 Sept. 2001, vol. 2, pp. 567–570.

[30] M. Streitenberger, F. Felgenhauer, H. Bresch, and W. Mathis, "Zero position coding (ZePoC) — a generalized concept of pulse-length modulated signals and its application

to class-D audio power amplifiers," in *Proc. Audio Eng. Soc. 110th Conv.*, Amsterdam, The Netherlands, 2001.

[31] R. W. Schafer and L. R. Rabiner, "A digital signal processing approach to interpolation," *Proceedings of the IEEE*, vol. 61, pp. 692–702, June 1973.

[32] P. Green and B. Silverman, *Nonparametric Regression and Generalized Linear Models: A Roughness Penalty Approach*, CRC Press, 1994.

[33] H. L. Van Trees, *Detection, Estimation, and Modulation Theory*, John Wiley and Sons, Inc., 1968.

[34] A. C. Rencher, *Multivariate Statistical Inference and Applications*, John Wiley & Sons, 1998.

[35] G. Golub and C. Van Loan, *Matrix Computations*, Johns Hopkins University Press, 1996.

[36] W. H. Press, *Numerical Recipes in C*, Cambridge University Press, 1988.

[37] S. Haykin, *Adaptive Filter Theory*, Prentice Hall, 1991.

[38] S. Kay, *Fundamentals of Statistical Signal Processing*, vol. 1, Prentice-Hall, 1993.

[39] T. I. Laakso, V. Valimaki, M. Karjalainen, and U. K. Laine, "Splitting the unit delay," *IEEE Signal Process. Mag.*, vol. 13, pp. 30–60, Jan. 1996.

[40] E. Meijering, "A chronology of interpolation: from ancient astronomy to modern signal and image processing," *Proceedings of the IEEE*, vol. 90, pp. 319–342, Mar. 2002.

[41] P. Midya, B. Roeckner, and S. Bergstedt, "Digital correction of PWM switching amplifiers," *IEEE Trans. Power Electron.*, vol. 2, pp. 68–72, June 2004.

[42] A. Papoulis and S. U. Pillai, *Probability, Random Variables and Stochastic Processes*, McGraw-Hill, 2002.

[43] D. Slepian and H. O. Pollak, "Prolate spheroidal wave functions, Fourier analysis and uncertainty – I," *Bell Sys. Tech. J.*, vol. 40, pp. 43–63, 1961.

[44] H. J. Landau and H. O. Pollak, "Prolate spheroidal wave functions, Fourier analysis and uncertainty – II," *Bell Sys. Tech. J.*, vol. 40, pp. 65–84, 1961.

[45] J. R. Magnus and H. Neudecker, *Matrix Differential Calculus*, John Wiley & Sons, 1999.

[46] Einar Hille, *Lectures on Ordinary Differential Equations*, Addison-Wesley, Reading, MA, 1969.

[47] B. Porat and B. Friedlander, "Computation of the exact information matrix of Gaussian time series with stationary random components," *IEEE Trans. Acoustics, Speech, and Signal Processing*, vol. ASSP-34, pp. 118–130, Feb. 1986.

Appendix A

Derivation of the Conditional Score Function

In this appendix, we derive the conditional score function $S(\mathbf{x}|\tau)$ defined as follows:

$$S(\mathbf{x}|\tau) \triangleq \frac{d}{d\tau} \log f(\mathbf{x}|\tau) \tag{A.1}$$

where

$$f(\mathbf{x}|\tau) = \frac{\exp\left(-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^T\Sigma^{-1}(\mathbf{x}-\boldsymbol{\mu})\right)}{(2\pi)^{M/2}|\Sigma|^{1/2}}. \tag{A.2}$$

and the vector $\boldsymbol{\mu}$ and matrix Σ are functions of the variable τ.

We begin by reviewing several basic properties that are used throughout the sequel. Then, we derive the general expression for $S(\mathbf{x}|\tau)$. Finally, we apply this result to our problem to generate a simpler expression.

A.1 Basic Proofs

These basic results are commonly known. Where complete proofs are omitted, they may be found in [45].

Property 1. *The derivative $\dot{\Sigma}^{-1}$ of the inverse of a nonsingular, square matrix Σ may be expressed as follows:*

$$\boxed{\dot{\Sigma}^{-1} = -\Sigma^{-1}\dot{\Sigma}\Sigma^{-1}} \tag{A.3}$$

Proof. As Σ is invertible, we may write

$$\Sigma^{-1}\Sigma = I$$

where I is the $M \times M$ identity matrix. We differentiate both sides, giving:

$$\dot{\Sigma}^{-1}\Sigma = -\Sigma^{-1}\dot{\Sigma}$$

Multiplying both sides by Σ^{-1} from the right yields the desired result. □

Property 2. *The derivative of a log-determinant is given by the following:*

$$\boxed{\frac{d}{d\tau}\log|\Sigma| = \mathrm{tr}\left\{\Sigma^{-1}\dot{\Sigma}\right\}} \tag{A.4}$$

Proof. This identity is a trivial extension of Jacobi's theorem, for which a proof is given in [46]. □

A.2 General Case

Combining (A.1) and (A.2), we have:

$$S(\mathbf{x}|\tau) = -\frac{1}{2}\frac{d}{d\tau}\log|\Sigma| - \frac{1}{2}\frac{d}{d\tau}(\mathbf{x}-\boldsymbol{\mu})^T\Sigma^{-1}(\mathbf{x}-\boldsymbol{\mu})$$

We may immediately differentiate the second term. Noting that Σ is symmetric, we may simplify the result to the following expression:

$$S(\mathbf{x}|\tau) = -\frac{1}{2}\frac{d}{d\tau}\log|\Sigma| + \dot{\boldsymbol{\mu}}^T\Sigma^{-1}(\mathbf{x}-\boldsymbol{\mu}) - \frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^T\dot{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu})$$

We apply Property 2 to the first term:

$$\boxed{S(\mathbf{x}|\tau) = -\frac{1}{2}\mathrm{tr}\left\{\Sigma^{-1}\dot{\Sigma}\right\} + \dot{\boldsymbol{\mu}}^T\Sigma^{-1}(\mathbf{x}-\boldsymbol{\mu}) - \frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^T\dot{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu})} \tag{A.5}$$

This is the most general expression for the conditional score, and is independent of the problem we consider. In the following section, we use the identities defined in (4.22) to simplify this result further.

A.3 Specialization to Discrete-Time Crossing Point Estimation

Equation (A.5) provides a general expression for $S(\mathbf{x}|\tau)$. In the following section, we apply this result to the crossing-point estimation problem in order to obtain a simpler expression.

We begin with the first term of (A.5). From the definition of Σ in (4.16), we have:

$$\dot{\Sigma} = -\frac{\dot{\boldsymbol{\rho}}\boldsymbol{\rho}^T + \boldsymbol{\rho}\dot{\boldsymbol{\rho}}^T}{\sigma_s^2}$$

and

$$\begin{aligned}
-\frac{1}{2}\operatorname{tr}\left\{\Sigma^{-1}\dot{\Sigma}\right\} &= \frac{1}{2\sigma_s^2}\operatorname{tr}\left\{\Sigma^{-1}\dot{\boldsymbol{\rho}}\boldsymbol{\rho}^T + \Sigma^{-1}\boldsymbol{\rho}\dot{\boldsymbol{\rho}}^T\right\} \\
&= \frac{1}{2\sigma_s^2}\left[\operatorname{tr}\left\{\Sigma^{-1}\dot{\boldsymbol{\rho}}\boldsymbol{\rho}^T\right\} + \operatorname{tr}\left\{\Sigma^{-1}\boldsymbol{\rho}\dot{\boldsymbol{\rho}}^T\right\}\right] \\
&= \frac{1}{2\sigma_s^2}\left[\boldsymbol{\rho}^T\Sigma^{-1}\dot{\boldsymbol{\rho}} + \dot{\boldsymbol{\rho}}^T\Sigma^{-1}\boldsymbol{\rho}\right] \\
&= \frac{\boldsymbol{\rho}^T\Sigma^{-1}\dot{\boldsymbol{\rho}}}{\sigma_s^2} \\
&= b
\end{aligned}$$

For the second term of (A.5), the definitions of $\boldsymbol{\mu}$ and Σ yeild

$$\dot{\boldsymbol{\mu}}^T\Sigma^{-1}(\mathbf{x}-\boldsymbol{\mu}) = \left(\frac{1}{\sigma_s^2}\dot{\boldsymbol{\rho}}y + \boldsymbol{\rho}\dot{y}\right)^T\Sigma^{-1}\left(\boldsymbol{x} - \frac{\boldsymbol{\rho}y}{\sigma_s^2}\right)$$

Thus, using the definitions of a, b, c and d,

$$\dot{\boldsymbol{\mu}}^T\Sigma^{-1}(\mathbf{x}-\boldsymbol{\mu}) = -\frac{\dot{y}(ay-c) + y(by-d)}{\sigma_s^2}$$

To simplify the third and final term of (A.5), we begin by applying Property 1, the definition

of $\boldsymbol{\mu}$, and the above expression for $\dot{\Sigma}$:

$$\begin{aligned}
-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^T\dot{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu}) &= -\frac{1}{2\sigma_s^2}(\mathbf{x}-\boldsymbol{\mu})^T\Sigma^{-1}\left(\dot{\boldsymbol{\rho}}\boldsymbol{\rho}^T+\boldsymbol{\rho}\dot{\boldsymbol{\rho}}^T\right)\Sigma^{-1}(\mathbf{x}-\boldsymbol{\mu}) \\
&= -\frac{1}{\sigma_s^2}\left(\mathbf{x}-\frac{\boldsymbol{\rho}y}{\sigma_s^2}\right)^T\Sigma^{-1}\dot{\boldsymbol{\rho}}\boldsymbol{\rho}^T\Sigma^{-1}\left(\mathbf{x}-\frac{\boldsymbol{\rho}y}{\sigma_s^2}\right) \\
&= -\frac{(ay-c)(by-d)}{\sigma_s^2}
\end{aligned}$$

Combining the above expressions yeilds the desired result:

$$\boxed{S(\mathbf{x}|\tau) = b - \frac{(ay-c)(by-d)+\dot{y}(ay-c)+y(by-d)}{\sigma_s^2}} \tag{A.6}$$

Appendix B

Derivation of Fisher Information for Multivariate Normal Distributions

B.1 General Case

In this section, we derive the Fisher Information Matrix (FIM) for a multivariate, real Gaussian distribution where the covariance matrix Σ and mean $\boldsymbol{\mu}$ are functions of the single parameter τ.

Theorem 1. *The Fisher Information for a multivariate real Gaussian random variable* $\mathbf{x}$ *with mean* $\boldsymbol{\mu}(\tau)$ *and covariance matrix* $\Sigma(\tau)$ *is given by the following:*[1]

$$\boxed{\mathcal{I} = \dot{\boldsymbol{\mu}}^T \Sigma^{-1} \dot{\boldsymbol{\mu}} + \frac{1}{2} \operatorname{tr}\left[\Sigma^{-1}\dot{\Sigma}\Sigma^{-1}\dot{\Sigma}\right]} \tag{B.1}$$

Proof. This derivation is similar to that found in [47], and the results are identical. Our derivation is notationally consistent with the rest of this work and corrects a minor error in their derivation.

We have:

$$\mathcal{I} = E\left\{\left[\frac{d}{d\tau}\log f(\mathbf{x})\right]^2\right\} \tag{B.2}$$

where $f(\mathbf{x})$ is given by (A.2). Although the squared quantity is identical to $S(\mathbf{x}|\tau)$ for our problem, it is straightforward and instructive to complete the derivation for the general

[1] As elsewhere, the overdot represents differentiation with respect to τ, and the dependence on τ has been suppressed for convenience.

case. We have:

$$\frac{d}{d\tau}\log f(\mathbf{x}) = -\frac{1}{2}\operatorname{tr}\left\{\Sigma^{-1}\dot{\Sigma}\right\} + \dot{\boldsymbol{\mu}}^T\Sigma^{-1}(\mathbf{x}-\boldsymbol{\mu}) - \frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^T\dot{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu}) \tag{B.3}$$

Squaring this expression, we get

$$\begin{aligned}\left[\frac{d}{d\tau}\log f(\mathbf{x})\right]^2 = {} & \frac{1}{4}\operatorname{tr}\left\{\Sigma^{-1}\dot{\Sigma}\right\}^2 + \left[\dot{\boldsymbol{\mu}}^T\Sigma^{-1}(\mathbf{x}-\boldsymbol{\mu})\right]^2 + \frac{1}{4}\left[(\mathbf{x}-\boldsymbol{\mu})^T\dot{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu})\right]^2 \\ & + \frac{1}{2}\operatorname{tr}\left\{\Sigma^{-1}\dot{\Sigma}\right\}(\mathbf{x}-\boldsymbol{\mu})^T\dot{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu}) \\ & - \operatorname{tr}\left\{\Sigma^{-1}\dot{\Sigma}\right\}\dot{\boldsymbol{\mu}}^T\Sigma^{-1}(\mathbf{x}-\boldsymbol{\mu}) \\ & - \dot{\boldsymbol{\mu}}^T\Sigma^{-1}(\mathbf{x}-\boldsymbol{\mu})(\mathbf{x}-\boldsymbol{\mu})^T\dot{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu})\end{aligned} \tag{B.4}$$

Thus,

$$\begin{aligned}E\left\{\left[\frac{d}{d\tau}\log f(\mathbf{x})\right]^2\right\} = {} & \frac{1}{4}\operatorname{tr}\left\{\Sigma^{-1}\dot{\Sigma}\right\}^2 \\ & + E\left\{\left[\dot{\boldsymbol{\mu}}^T\Sigma^{-1}(\mathbf{x}-\boldsymbol{\mu})\right]^2\right\} \\ & + \frac{1}{4}E\left\{\left[(\mathbf{x}-\boldsymbol{\mu})^T\dot{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu})\right]^2\right\} \\ & + \frac{1}{2}\operatorname{tr}\left\{\Sigma^{-1}\dot{\Sigma}\right\}E\left\{(\mathbf{x}-\boldsymbol{\mu})^T\dot{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu})\right\} \\ & - \operatorname{tr}\left\{\Sigma^{-1}\dot{\Sigma}\right\}E\left\{\dot{\boldsymbol{\mu}}^T\Sigma^{-1}(\mathbf{x}-\boldsymbol{\mu})\right\} \\ & - E\left\{\dot{\boldsymbol{\mu}}^T\Sigma^{-1}(\mathbf{x}-\boldsymbol{\mu})(\mathbf{x}-\boldsymbol{\mu})^T\dot{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu})\right\}\end{aligned} \tag{B.5}$$

We now evaluate each expectation term in sequence. The second term of (B.5) is the second moment of $\dot{\boldsymbol{\mu}}^T\Sigma^{-1}(\mathbf{x}-\boldsymbol{\mu}) \sim N(0, \dot{\boldsymbol{\mu}}^T\Sigma^{-1}\Sigma\Sigma^{-1}\dot{\boldsymbol{\mu}})$, and thus equals $\dot{\boldsymbol{\mu}}^T\Sigma^{-1}\dot{\boldsymbol{\mu}}$. For the third

term, we have [47]:

$$\begin{aligned} E\left\{\left[(\mathbf{x}-\boldsymbol{\mu})^T\dot{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu})\right]^2\right\} &= E\left\{\sum_{i,j,m,n} x_i\dot{\Sigma}^{-1}_{ij}x_jx_m\dot{\Sigma}^{-1}_{mn}x_n\right\} \\ &= \sum_{i,j,m,n}\dot{\Sigma}^{-1}_{ij}\dot{\Sigma}^{-1}_{mn}E\left\{x_ix_jx_mx_n\right\} \\ &= \sum_{i,j,m,n}\dot{\Sigma}^{-1}_{ij}\dot{\Sigma}^{-1}_{mn}\left[\Sigma_{ij}\Sigma_{mn}+\Sigma_{im}\Sigma_{jn}+\Sigma_{in}\Sigma_{jm}\right] \\ &= \mathrm{tr}\left\{\dot{\Sigma}^{-1}\Sigma\right\}^2+2\,\mathrm{tr}\left\{\dot{\Sigma}^{-1}\Sigma\dot{\Sigma}^{-1}\Sigma\right\} \end{aligned}$$

Applying Property 1, we get:

$$E\left\{\left[(\mathbf{x}-\boldsymbol{\mu})^T\dot{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu})\right]^2\right\} = \mathrm{tr}\left\{\Sigma^{-1}\dot{\Sigma}\right\}^2+2\,\mathrm{tr}\left\{\Sigma^{-1}\dot{\Sigma}\Sigma^{-1}\dot{\Sigma}\right\} \tag{B.6}$$

For the fourth term of (B.5), we have:

$$\begin{aligned} E\left\{(\mathbf{x}-\boldsymbol{\mu})^T\dot{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu})\right\} &= E\left\{\mathrm{tr}\left\{(\mathbf{x}-\boldsymbol{\mu})^T\dot{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu})\right\}\right\} \\ &= \mathrm{tr}\left\{\dot{\Sigma}^{-1}E\left\{(\mathbf{x}-\boldsymbol{\mu})(\mathbf{x}-\boldsymbol{\mu})^T\right\}\right\} \\ &= \mathrm{tr}\left\{\dot{\Sigma}^{-1}\Sigma\right\} \end{aligned}$$

Applying Property 1, we get:

$$E\left\{(\mathbf{x}-\boldsymbol{\mu})^T\dot{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu})\right\} = -\,\mathrm{tr}\left\{\Sigma^{-1}\dot{\Sigma}\right\} \tag{B.7}$$

The fifth and sixth terms of (B.5) are zero, since they are odd-order moments of zero-mean Gaussian distributions. Collecting the above results, we have:

$$\boxed{E\left\{\left[\frac{d}{d\tau}\log f(\mathbf{x})\right]^2\right\} = \dot{\boldsymbol{\mu}}^T\Sigma^{-1}\dot{\boldsymbol{\mu}}+\frac{1}{2}\,\mathrm{tr}\left\{\Sigma^{-1}\dot{\Sigma}\Sigma^{-1}\dot{\Sigma}\right\}} \tag{B.8}$$

which is the desired result.

□

B.2 Specialization to Discrete-Time Crossing Point Estimation

We wish to express (B.8) in a simplified form, using the definitions for Σ and $\boldsymbol{\mu}$ defined in (4.17) and (4.16). We have:

$$\dot{\boldsymbol{\mu}}^T\Sigma^{-1}\dot{\boldsymbol{\mu}} + \frac{1}{2}\operatorname{tr}\left\{\Sigma^{-1}\dot{\Sigma}\Sigma^{-1}\dot{\Sigma}\right\} = \frac{1}{\sigma_s^4}\left[y^2\dot{\boldsymbol{\rho}}^T\Sigma^{-1}\dot{\boldsymbol{\rho}} + 2y\dot{y}\boldsymbol{\rho}^T\Sigma^{-1}\dot{\boldsymbol{\rho}} + \dot{y}^2\boldsymbol{\rho}^T\Sigma^{-1}\boldsymbol{\rho}\right] + \frac{1}{2\sigma_s^4}\left[\operatorname{tr}\left\{\Sigma^{-1}(\dot{\boldsymbol{\rho}}\boldsymbol{\rho}^T + \boldsymbol{\rho}\dot{\boldsymbol{\rho}}^T)\Sigma^{-1}(\dot{\boldsymbol{\rho}}\boldsymbol{\rho}^T + \boldsymbol{\rho}\dot{\boldsymbol{\rho}}^T)\right\}\right] \tag{B.9}$$

$$= \frac{1}{\sigma_s^2}\left[y^2e + 2y\dot{y}b + \dot{y}^2a\right] + \frac{1}{2\sigma_s^2}\operatorname{tr}\left\{\Sigma^{-1}\left[\dot{\boldsymbol{\rho}}b\boldsymbol{\rho}^T + \dot{\boldsymbol{\rho}}a\dot{\boldsymbol{\rho}}^T + \boldsymbol{\rho}e\boldsymbol{\rho}^T + \boldsymbol{\rho}b\dot{\boldsymbol{\rho}}^T\right]\right\} \tag{B.10}$$

Rearranging the $\operatorname{tr}\{\cdot\}$ expressions yeilds the desired result (4.33):

$$\boxed{\dot{\boldsymbol{\mu}}^T\Sigma^{-1}\dot{\boldsymbol{\mu}} + \frac{1}{2}\operatorname{tr}\left\{\Sigma^{-1}\dot{\Sigma}\Sigma^{-1}\dot{\Sigma}\right\} = \frac{1}{\sigma_s^2}\left[y^2e + 2y\dot{y}b + \dot{y}^2a\right] + b^2 + ae} \tag{B.11}$$

The final step from (4.33) to (4.34) is a straightforward substitution of each of (4.22) into (4.24). The resulting expression may be written using (4.25).

Appendix C

Proof of (4.24)

We wish to prove that the equation

$$\Sigma^{-1} = \Sigma_0^{-1} + \frac{\Sigma_0^{-1}\boldsymbol{\rho}\boldsymbol{\rho}^T\Sigma_0^{-1}}{\sigma_s^2 - \boldsymbol{\rho}^T\Sigma_0^{-1}\boldsymbol{\rho}} \tag{C.1}$$

is a valid expression for Σ^{-1}. As stated in Section 4.2.4, this statement is valid provided that Σ_0 is nonsingular (which is granted, since we assume Σ_0 is positive definite) and provided $\sigma_s^2 - \boldsymbol{\rho}(\tau)^T\Sigma_0^{-1}\boldsymbol{\rho}(\tau) \neq 0$ (which we will establish here.)

We may express Σ_0^{-1} by expanding (4.4) with the Sherman-Morrison-Woodbury formula [35]. The result is unconditionally valid for $\sigma_n^2 \neq 0$.

$$\Sigma_0^{-1} = \Sigma_s^{-1} - \Sigma_s^{-1}(\sigma_n^{-2}I + \Sigma_s^{-1})^{-1}\Sigma_s^{-1} \tag{C.2}$$

As the sum of positive definite matrices, the center term $(\sigma_n^{-2}I + \Sigma_s^{-1})$ and its inverse are positive definite. Likewise, the matrix quadratic $\Sigma_s^{-1}(\sigma_n^{-2}I + \Sigma_s^{-1})^{-1}\Sigma_s^{-1}$ is positive definite. Thus, the second term in the sum is necessarily positive, and for any $\boldsymbol{\rho}(\tau) \neq \mathbf{0}$ we have the strict inequality

$$\boldsymbol{\rho}(\tau)^T\Sigma_0^{-1}\boldsymbol{\rho}(\tau) < \boldsymbol{\rho}(\tau)^T\Sigma_s^{-1}\boldsymbol{\rho}(\tau) \tag{C.3}$$

We may immediately disregard the case in which $\boldsymbol{\rho}(\tau) = \mathbf{0}$, since it corresponds to the degenerate case in which the observation vector $\mathbf{x}$ is totally uncorrelated with $s(\tau)$ at a particular τ.

To complete the proof, we now show that $\boldsymbol{\rho}(\tau)^T\Sigma_s^{-1}\boldsymbol{\rho}(\tau) \leq \sigma_s^2$. This is trivial, since $\epsilon = \sigma_s^2 - \boldsymbol{\rho}(\tau)^T\Sigma_s^{-1}\boldsymbol{\rho}(\tau)$ is the variance of the MMSE estimator to $s(\tau)$ given the (unobservable)

sample vector s [37]. The quadratic expression is nonnegative, and since variances are by definition nonnegative, $\rho(\tau)^T\Sigma_s^{-1}\rho(\tau) \leq \sigma_s^2$.

Summarizing, we have shown:

$$\rho(\tau)^T\Sigma_0^{-1}\rho(\tau) < \rho(\tau)^T\Sigma_s^{-1}\rho(\tau) \leq \sigma_s^2 \tag{C.4}$$

therefore,

$$\boxed{\rho(\tau)^T\Sigma_0^{-1}\rho(\tau) \neq \sigma_s^2.} \tag{C.5}$$