Autonomous Software-Defined Radio Receivers for Deep Space Applications

Jon Hamkins and Marvin K. Simon, Editors

Jet Propulsion Laboratory
California Institute of Technology

DEEP SPACE COMMUNICATIONS AND NAVIGATION SERIES

DEEP SPACE COMMUNICATIONS AND NAVIGATION SERIES

Issued by the Deep Space Communications and Navigation Systems
Center of Excellence
Jet Propulsion Laboratory
California Institute of Technology

Joseph H. Yuen, Editor-in-Chief

Published Titles in this Series

Radiometric Tracking Techniques for Deep-Space Navigation
Catherine L. Thornton and James S. Border

Formulation for Observed and Computed Values of Deep Space Network Data Types for Navigation
Theodore D. Moyer

Bandwidth-Efficient Digital Modulation with Application to Deep-Space Communications
Marvin K. Simon

Large Antennas of the Deep Space Network
William A. Imbriale

Antenna Arraying Techniques in the Deep Space Network
David H. Rogstad, Alexander Mileant, and Timothy T. Pham

Radio Occultations Using Earth Satellites: A Wave Theory Treatment
William G. Melbourne

Deep Space Optical Communications
Hamid Hemmati, Editor

DEEP SPACE COMMUNICATIONS AND NAVIGATION SERIES

Spaceborne Antennas for Planetary Exploration
William A. Imbriale, Editor

Autonomous Software-Defined Radio Receivers for Deep Space Applications
Jon Hamkins and Marvin K. Simon, Editors

Autonomous Software-Defined Radio Receivers for Deep Space Applications

Jon Hamkins and Marvin K. Simon, Editors

Jet Propulsion Laboratory
California Institute of Technology

DEEP SPACE COMMUNICATIONS AND NAVIGATION SERIES

Autonomous Software-Defined Radio Receivers for
Deep Space Applications

2006

The research described in this publication was carried out at the
Jet Propulsion Laboratory, California Institute of Technology, under
a contract with the National Aeronautics and Space Administration.

Table of Contents

Foreword

The Deep Space Communications and Navigation Systems Center of Excellence (DESCANSO) was established in 1998 by the National Aeronautics and Space Administration (NASA) at the California Institute of Technology's Jet Propulsion Laboratory (JPL). DESCANSO is chartered to harness and promote excellence and innovation to meet the communications and navigation needs of future deep-space exploration.

DESCANSO's vision is to achieve continuous communications and precise navigation—any time, anywhere. In support of that vision, DESCANSO aims to seek out and advocate new concepts, systems, and technologies; foster key technical talents; and sponsor seminars, workshops, and symposia to facilitate interaction and idea exchange.

The Deep Space Communications and Navigation Series, authored by scientists and engineers with many years of experience in their respective fields, lays a foundation for innovation by communicating state-of-the-art knowledge in key technologies. The series also captures fundamental principles and practices developed during decades of deep-space exploration at JPL. In addition, it celebrates successes and imparts lessons learned. Finally, the series will serve to guide a new generation of scientists and engineers.

Joseph H. Yuen
DESCANSO Leader

Preface

The steady march of radio receiver technology over the decades has enabled smaller and increasingly capable radio terminals. The development of the transistor in the 1950s enabled the replacement of heavy tubes, which reduced receiver weight dramatically. To take advantage of the lighter transistor, engineers in the second half of the twentieth century engaged in a sustained effort to replace each analog component of the radio with a digital component. For example, the analog phase-locked loop comprising a linear multiplier, voltage-controlled oscillator, and resistor capacitor (RC) loop filter was replaced by a digital phase-locked loop comprising a digital multiplier, numerically controlled oscillator, and digitally implemented filter. Over the years, the digital domain was pushed further and further toward the front end of the receiver, until eventually the all-digital receiver was born in the 1990s. This receiver converts the intermediate frequency signal directly to digital samples, with all processing thereafter accomplished digitally.

With all operations done digitally, radios began to become more flexible as well, because loop bandwidths, gains, data rates, etc., were easier to adjust in the digital domain. This increasingly capable reconfigurable radio evolved into what is now called a software-defined radio (SDR), or cognitve radio, in which nearly all aspects of the radio are redefinable. A premier example of an SDR is the National Aeronautic and Space Administration (NASA) Electra radio, in which the baseband processing is entirely implemented in a reconfigurable field programmable gate array (FPGA). Virtually any channel code, modulation, and data rate may be accommodated via suitable reprogramming of this SDR.

The purpose of this monograph, and the natural next step in the evolution of radio receiver technology, is the development of techniques to *autonomously* configure an all-digital SDR receiver for whatever type of signal happens to hit its antenna. We describe automatic identification of the carrier frequency,

modulation index, data rate, modulation type, and pulse shape, based on observations of the received signal. These are functions that typically are configured manually by the user of an SDR, prior to reception, based on a priori knowledge. We also describe how the conventional receiver estimators for the signal-to-noise ratio, carrier phase, and symbol timing require knowledge of the modulation type, data rate, and so on, and we show how these conventional functions can be implemented in the absence of this information.

For each of the estimators described above, we develop the optimal solution using a maximum-likelihood (ML) approach, and we offer simplifications and low-complexity approximations. When the solutions are highly complex or intractable, we present ad hoc estimators. Finally, we show how the suite of estimators may be combined into a working radio receiver.

Although the title of this monograph indicates that the autonomous radio technology described herein is for deep-space applications—and it certainly applies there, for auto-configuration of both Deep Space Network (DSN) and situ relay radio receivers—the development here is actually quite general. Indeed, virtually any terrestrial radio with the capability to process more than one type of signal can take advantage of the theoretical development and algorithms presented in the monograph.

Jon Hamkins and Marvin K. Simon
Editors
2006

Acknowledgments

The work described was performed at the Jet Propulsion Laboratory, California Institute of Technology, under contract with the National Aeronautics and Space Administration. It was funded by NASA's Mars Technology Program, JPL's Research and Technology Development Program, and the Deep Space Communications and Navigation Center of Excellence at JPL.

The authors wish to thank Kenneth Andrews, Fabrizio Pollara, and Victor Vilnrotter for their many helpful discussions on modulation classification and other topics related to autonomous radio receivers, Ginny Azevedo for drawing many of the figures, and Pat Ehlers for preparing the published manuscript.

Jon Hamkins and Marvin K. Simon
Editors
2006

Contributors

David Bell received his B.S.E.E. from Cornell University in 1980 and M.S.E.E. from Stanford University in 1981. Mr. Bell joined the Jet Propulsion Laboratory (JPL) in 1981, working on deep-space and commercial satellite applications research and development. He currently is the Supervisor of the Communications Systems Engineering and Operations Group. Outside of JPL, Mr. Bell has taught satellite courses at the University of California, Los Angeles (UCLA), and with Pete Conrad has developed and started Universal Spacenet, a commercial satellite telemetry, tracking, and control (TT&C) service provider.

Ann Devereaux received her B.S. in Computer Engineering from the Massachusetts Institute of Technology (MIT) in 1989 and her M.S. in Communications Engineering from the University of Southern California (USC) in 1994. She has worked at JPL since 1989 in communications systems for spacecraft and deep-space ground elements as well as for commercial and military applications. She is the Supervisor of the Proximity Radios Group, which develops relay link radios and generic instrument interface and data processors for space flight use.

Dariush Divsalar received his Ph.D. in Electrical Engineering from UCLA in 1978. Since then, he has been with JPL, where he is a Principal Scientist. During the past 20 years, he has taught graduate courses at UCLA and Caltech. He has published over 150 papers, coauthored 3 books, and holds 10 United States (U.S.) patents. Recently, one of his papers was selected as one of the key research papers published by the Institute of Electrical and Electronics Engineers (IEEE) Communications Society during the past five decades. He has received over 25 National Aeronautics and Space Administration (NASA)

Tech Brief awards and a NASA Exceptional Engineering Achievement Medal. Dr. Divsalar is a Fellow of IEEE.

Sam Dolinar received his Ph.D. in Electrical Engineering from MIT in 1976, where his master's and doctoral theses were on optical communications. He worked at MIT Lincoln Laboratory before joining JPL in 1980. Dr. Dolinar has focused his research on channel coding and source coding for the deep-space channel, especially turbo codes and low-density parity-check codes during the past decade. He teaches data compression at Caltech.

Todd Ely received his Ph.D. in Astronautical Engineering from Purdue University in 1996. Prior to joining JPL in 1999, he was an officer in the U.S. Air Force and a Visiting Assistant Professor at Purdue University. Dr. Ely is currently a Senior Engineer in the Guidance, Navigation, and Control Section, where he is the Mars Network Navigation Lead, supporting navigation system trades for Mars and radio navigation technology development.

Edwin Grigorian received his B.S.E.E. in 1985 and his M.S.E.E. in 1989, both from the University of Alabama in Huntsville, and is currently pursuing his Ph.D. in Electrical Engineering at USC. Mr. Grigorian has been at JPL since 1990, where he is a senior member of the engineering staff developing analog/digital integrated circuits. Some of his recent work has included the design and integration of multi-watt fully integrated complementary metal oxide semiconductor (CMOS) power amplifiers and drivers for an L-band radar transmitter/receiver (T/R) module, design of field programmable gate array (FPGA) and low-power application-specific integrated circuit (ASIC)-based digital radios for ultra-high frequency (UHF) proximity links, and digital design of a neural prosthetic implant mixed-signal ASIC.

Jon Hamkins received his B.S. from the California Institute of Technology (Caltech) in 1990 and Ph.D. from the University of Illinois at Urbana-Champaign in 1996, both in Electrical Engineering. Dr. Hamkins has been at JPL since 1996, where he is the technical supervisor of the Information Processing Group, which performs research in autonomous radios, optical communications, information theory, channel coding, data compression, and synchronization.

Thomas Jedrey received his B.S. in Mathematics from the University of Maryland, College Park, Maryland, in 1979. He received his M.A. in Probability and Statistics from the University of Maryland in 1982 and his M.S. in Electrical Engineering from USC in 1990. He worked at the U.S. Army Harry Diamond Laboratories and at Gould Electronic Systems before joining JPL in 1986. He led the successful development of the Electra Proximity Radio,

and he is now the Deputy Manager of the Flight Communications Systems Section at JPL.

Igor Kuperman received his B.S. from California State Polytechnic University, Pomona, in 2001 and his M.S. from USC in 2005, both in Electrical Engineering. Mr. Kuperman has been at JPL since 2000, where he is a member of the engineering staff in the Communications Architectures and Research Section. Some of his recent work includes software and firmware design for digital UHF proximity radios and implementation of coding and data compression algorithms.

Alan Lee received his B.S. from the University of California, Berkeley, in 1988, M.S.E.E. from UCLA in 1990, and E.E.E. from USC in 1997. Mr. Lee has been at JPL since 1999, where he is the Supervisor of the Electronics Design Group for the Flight System Avionics Section.

Edgar Satorius received his Ph.D. in Electrical Engineering from Caltech in 1975, where his doctoral thesis was on electromagnetic propagation through non-linear plasmas. He worked at the Naval Ocean Systems Center, Dynatech, and Gould Electronic Systems before joining JPL in 1986. Dr. Satorius has focused his research on adaptive signal processing and communication and has several innovations in data equalization and direction finding during the past decade. He teaches digital signal processing at USC.

Hooman Shirani-Mehr received his B.S. from Sharif University of Technology, Tehran, Iran, in 2001 and M.S. from USC in 2006, both in Electrical Engineering. He currently is a Ph.D. student in Electrical Engineering at USC. His research interests are in communications and information theory.

Marvin Simon received his Ph.D. in Electrical Engineering from New York University in 1966. He was employed at the Bell Telephone Laboratories from 1966 to 1968. Since 1968 he has been with JPL, where he is currently a Principal Scientist. During this time period, he has performed digital communications research as applied to the design of NASA deep-space and near-Earth missions, with particular emphasis in the disciplines of synchronization, trellis coding, spread spectrum, and modulation and demodulation techniques. In the past, Dr. Simon also held a joint appointment in the Electrical Engineering Department at Caltech. Dr. Simon has published over 170 papers and authored and co-authored 12 books.

Andre Tkacenko received his B.S., M.S., and Ph.D. degrees in Electrical Engineering from Caltech in 1999, 2001, and 2004, respectively. Dr. Tkacenko has been at JPL since 2005, where he is a Research Engineer for the Digital Signal Processing Group. His primary fields of research include multirate systems theory, digital communications, and atmospheric modeling for antenna arrays.

Chapter 1
Introduction and Overview

Jon Hamkins and Marvin K. Simon

The National Aeronautics and Space Administration (NASA) has embarked on an ambitious project to develop new technology for a radio to receive a signal without much a priori knowledge about its defining characteristics [1]. As a first step in this direction, a suite of modules has been developed to autonomously recognize various signal attributes, including the angle of arrival, data rate, symbol timing, carrier frequency and phase, modulation index, modulation type, and signal-to-noise ratio (SNR). This chapter is an overview of the architecture of the autonomous radio receiver, describing what each module does and how the modules interact to produce the desired effect.

The primary application of this technology is in relaying communication signals from multiple deep-space assets. For example, one might want two or more rovers on a distant planet to relay data through an orbiter, as the two Mars Exploration Rovers have done via Mars Global Surveyor and Mars Odyssey [2]. Multiple landed assets communicating through relays will continue to be an important part of NASA's exploration plans throughout the next two decades [3,4]. Over a period of years, we may expect NASA and other space agencies to launch a set of diverse orbiters and landers,[1] and because technology continues to emerge, it is unlikely that they will all use the same data rates, protocols, error-correcting codes, and modulation types.

The advantage of an autonomous radio in this emerging scenario is that it can communicate to each asset that comes into view, automatically, without having to be reconfigured from Earth for each pass to account for differences in the signal characteristics. The radio would receive whatever each landed asset

[1] See `http://www.jpl.nasa.gov/missions/future_missions.cfm`.

sent. Since an orbital period may be a few hours, an orbiter may come within view of various landed assets several times a day, and the automatic reconfiguration would be this frequent. By comparison, manual radio reconfiguration would be a daunting task.

In addition to easing the scheduling and configuration burden, an autonomous radio also will gracefully handle unpredictable or anomalous events. For example, during entry, descent, and landing (EDL), a spacecraft can undergo large Doppler swings caused by rocket firings, parachute openings, backshell ejection, and a bouncing landing on the surface. Even when all scheduled events occur successfully, there may be Doppler uncertainty due to unpredictable properties of the atmosphere. Ideally, the communication link should operate whether or not each of the EDL events is successful, but the uncertainties involved typically lead to liberal link margins—for example, the Mars Exploration Rovers observed link margins that sometimes exceeded 10 dB. An autonomous radio could substantially reduce this margin because it would handle any Doppler swing nearly optimally.

Unfortunately, such flexible technology is not available on NASA's currently flying missions. In perhaps the most glaring example of this, NASA engineers discovered in 2000 that a receiver aboard Cassini, launched in 1997, would fail during the Huygens probe descent onto Titan because it did not properly account for the Doppler profile of the probe [5]. Increasing the loop bandwidth of the synchronization loops would have easily fixed the problem, but, unfortunately, these loop bandwidths were hard-wired to fixed values on the spacecraft. With superior engineering and enormous dedication, NASA and the European Space Agency were still able to save the mission by slightly altering the original trajectory, but this solution required forming a large and expensive international recovery team to find the appropriate recommendations on how to overcome the radio's severe limitations.

This chapter is an overview of the architecture of an autonomous radio of the type described above. In Section 1.1 we describe a general model for a received signal that will be used throughout the monograph, and we define many parameters one might desire to estimate from the signal. In Section 1.2 we describe in detail the differences between a conventional receiver, a reconfigurable receiver such as the first-generation Electra, and an autonomous receiver. In Sections 1.3 and 1.4 we describe a suite of individual modules that estimate or classify the signal parameters, along with a message-passing strategy to improve performance, and in Section 1.5 we describe a software implementation of these cooperative modules.

1.1 Preliminaries

1.1.1 Signal Model

We describe here a common signal model that is used throughout the monograph. The received bandpass waveform can be written as

$$r(t) = s(t) + n(t) \tag{1-1}$$

where $s(t)$ is the signal component and $n(t)$ is a passband additive white Gaussian noise (AWGN) process with two-sided power spectral density (PSD) $N_0/2$ W/Hz. We may write

$$\begin{aligned} r(t) &= \mathrm{Re}\left\{\tilde{r}(t)e^{j\omega_c t}\right\} \\ s(t) &= \mathrm{Re}\left\{\tilde{s}(t)e^{j\omega_c t}\right\} \\ n(t) &= \mathrm{Re}\left\{\tilde{n}(t)e^{j\omega_c t}\right\} \end{aligned} \tag{1-2}$$

where

$$\tilde{r}(t) = \tilde{s}(t) + \tilde{n}(t) \tag{1-3}$$

is the complex baseband representation of the bandpass signal $r(t)$ centered at a carrier frequency of ω_c rad/s. The complex baseband AWGN noise process can be expanded as $\tilde{n}(t) = \sqrt{2}[n_c(t) + jn_s(t)]$, where $n_c(t)$ and $n_s(t)$ are independent AWGN processes, each with two-sided power-spectral density $N_0/2$ W/Hz. Thus, we may rewrite the passband noise process as

$$n(t) = \sqrt{2}n_c(t)\cos(\omega_c t) - \sqrt{2}n_s(t)\sin(\omega_c t) \tag{1-4}$$

The transmitted signal $\tilde{s}(t)$ is assumed to be a single-channel amplitude- and phase-modulated signal with or without a residual carrier, of the form

$$\tilde{s}(t) = \sqrt{2P_d}\sum_{k=-\infty}^{\infty} d_k(t)p(t-kT)e^{j[\theta_c(t)-\pi/2]} + \sqrt{2P_c}e^{j[\theta_c(t)-\pi/2]} \tag{1-5}$$

where

- P_d and P_c are the powers of the real passband data and residual-carrier signals,[2] respectively
- $d_k(t) \triangleq A_k e^{j\theta_k g(t)}$ is the complex modulation for the kth symbol, where
  - A_k is the normalized amplitude satisfying $E[A_k^2] = 1$
  - θ_k is the phase modulation for the kth symbol
  - $g(t)$ represents the subcarrier modulation and is typically of the form $g(t) = \sin(\omega_{sc}t)$ or $g(t) = \text{sgn}[\sin(\omega_{sc}t)]$, where ω_{sc} is the subcarrier frequency in radians per second, or $g(t) = 1$ if there is no subcarrier
- $p(t)$ is a pulse shape satisfying $T^{-1}\int_0^T p^2(t)dt = 1$
- T is the symbol duration in seconds
- $\theta_c(t)$ is the carrier phase

Equation (1-5) represents a binary phase-shift keying (BPSK) signal when $\theta_k = c_k\pi/2$, $c_k \in \{-1, 1\}$, $A_k = 1$, $g(t) = 1$, and $p(t) \in \{-1, 1\}$ for all $t \in [0, T)$, in which case we may rewrite Eq. (1-5) as

$$\tilde{s}(t) = \sqrt{2P_t} \sum_{k=-\infty}^{\infty} p(t - kT)e^{j[\beta c_k + \theta_c(t) - \pi/2]} \tag{1-6}$$

where $P_t = P_c + P_d$ is the total passband signal power and $\beta = \tan^{-1}\sqrt{P_d/P_c}$ is the *modulation angle*, also referred to as the *modulation index*. For an M-ary phase-shift keying (M-PSK) signal with no residual carrier, Eq. (1-5) becomes

$$\tilde{s}(t) = \sqrt{2P_t} \sum_{k=-\infty}^{\infty} e^{j[\theta_k + \theta_c(t) - \pi/2]}p(t - kT) \tag{1-7}$$

where $\theta_k = [2q_k + (1 + (-1)^{M/2})/2]\pi/M$ is the phase modulation for the kth M-PSK symbol, with independent and uniformly distributed $q_k \in \{0, 1, \cdots, M–1\}$.

---

[2] If the power of a signal $x(t)$ is defined as $1/T'\int_0^{T'} x(t)x^*(t)dt$, then the power of the complex baseband signals is twice that of the corresponding passband signal. Thus, the complex baseband signal $\tilde{s}(t)$ in Eq. (1-5) has power $2(P_d + P_c)$, while the power of the corresponding passband signal $s(t)$ is $P_d + P_c$.

Quadrature amplitude modulation (QAM) may also be represented by Eq. (1-5) by appropriately defining $d_k(t)$.

At the receiver, the timing and carrier phase are initially unknown, and noise is present. If we also assume that the carrier frequency ω_c is imperfectly estimated as $\hat{\omega}_c$ at the front end of the receiver, then a residual frequency component $\omega_r = \omega_c - \hat{\omega}_c$ will remain after conversion to baseband, and the resulting signal will have the form

$$\tilde{r}(t) = \sqrt{2P_d} \sum_{k=-\infty}^{\infty} d_k(t) p[t - kT - \varepsilon T] e^{j[\omega_r t + \theta_c(t) - \pi/2]} + \sqrt{2P_c} e^{j[\omega_r t + \theta_c(t) - \pi/2]} + \tilde{n}(t) \tag{1-8}$$

where ε is the unknown fractional symbol timing. A priori, ε is uniformly distributed on $[0, 1)$ and $\theta_c(t)$ is uniformly distributed on $[0, 2\pi)$. For our purposes, we assume that the carrier phase $\theta_c(t)$ is slowly varying with respect to the data rate; thus, we shall hereafter drop the dependence on time in the notation.

1.1.2 Anatomy of the Received Signal

Figure 1-1 graphically indicates the dependence of the received signal on several factors. We group the signal dependence graph into three primary components: the forward error-correcting (FEC) code encoder, the modulator/amplifier, and the channel. Each of these is affected by several sub-factors, including the ones shown in Fig. 1-1 as well as others which we call out in italics in the more detailed discussion below.

1.1.2.1 FEC Code. The FEC code can be one of several *code types*. The code types standardized by the Consultative Committee for Space Data Systems (CCSDS) for deep-space [6] or in situ [7] communications include Reed–Solomon (RS) codes, convolutional codes, turbo codes, Bose–Chaudhuri–Hocquenghem (BCH) codes, and cyclic redundancy check (CRC) codes. Work is also progressing rapidly both academically and in various standards (Digital Video Broadcast/Satellite, the Institute of Electrical and Electronics Engineers (IEEE) 802.{11n,15.3a,16e}, and CCSDS deep space and in situ) on low-density parity-check codes and progressive parity-type codes such as tornado and raptor codes.

Associated with each FEC code is its *code rate*, which is the fraction of symbols carrying information, and its *code length*, which indicates the number of symbols in each codeword. For some code types, these parameters alone are nearly enough to completely identify the code. For example, the best-performing

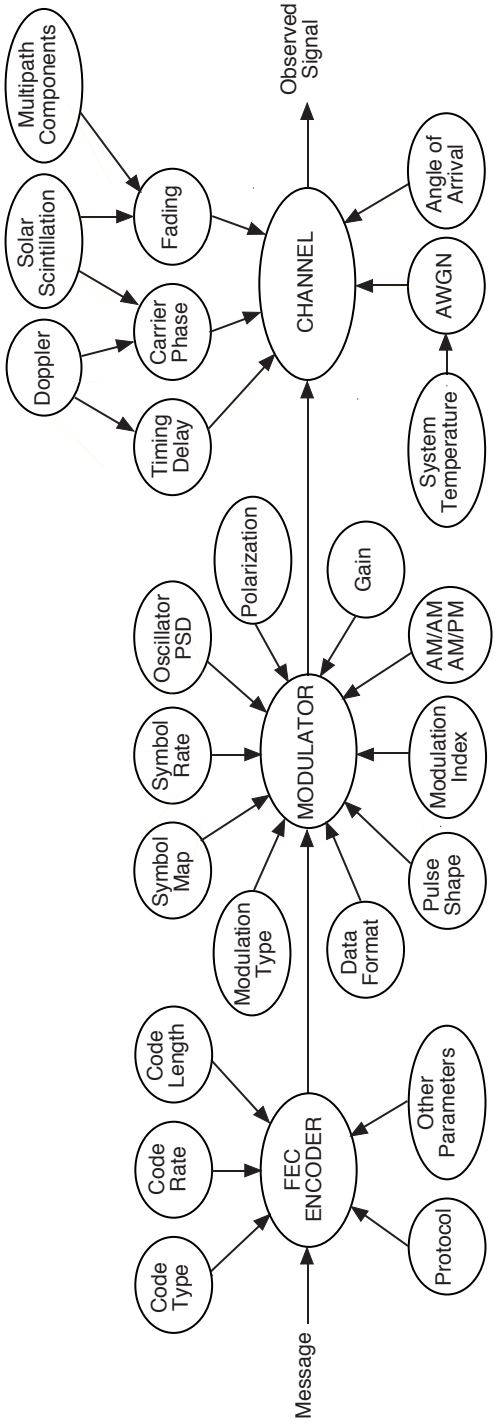


Fig. 1-1. Signal dependency graph.

convolutional codes for a given rate and constraint length are tabulated in textbooks (e.g., [8]), and applications using convolutional codes nearly always use codes from these tables. CRC codes of a given length also typically use standard generator polynomials [7]. RS codes are specified by their blocklength, rate, field generator polynomial, and code generator polynomials. The latter two can be one of several possibilities, but in practice space communication systems have primarily used the one that is specified in the CCSDS standard [6].

The techniques of *puncturing*, *shortening*, and *expurgating* are commonly used to modify a code. Puncturing raises the code rate, and it is the standard technique to obtain the CCSDS turbo codes of rate 1/4, 1/3, and 1/2 from the rate 1/6 mother code [6]. Shortening is often used with the standard RS(255,223) code—the full-length code with interleaving depth 5 has input-length $223 \times 8 \times 5 = 8920$, but missions often shorten this to 8800, a multiple of 32, which is a convenient quantity for spacecraft processors to handle. The (63,56) BCH code used for uplink commanding [9] is an expurgated (63,57) Hamming code, with odd-weight codewords removed.

In addition to the substantive factors mentioned above, there are a number of superficial factors that determine the FEC encoder output. The precise *bit ordering*, use of *trellis termination*, and *placement of frame headers, synchronization bits, and filler bits* are examples of these factors.

1.1.2.2 Modulator and Amplifier. The modulator uses the coded binary sequence from the output of the FEC encoder to modulate a carrier signal. This process depends on several factors. The *modulation type* identifies the signal constellation from which the transmitted symbols are chosen. BPSK, quadrature phase-shift keying (QPSK), quadrature amplitude modulation (QAM), and Gaussian-filtered minimum-shift keying (GMSK) are commonly used modulation types [10]. In the case of GMSK and other filtered modulation types, the bandwidth–time (BT) product is also needed to fully specify the modulation.

The assignment of FEC-encoded bits to symbols is defined by a *mapping*, which may be a static mapping such as a natural ordering, Gray code, or anti-Gray code, which maps each block of bit(s) to a symbol; or, the mapping may be dynamically controlled through a state machine, as it is with trellis-coded modulation [11].

The *symbol rate*, or baud, defines the number of discrete signal constellation elements transmitted per second. Within each symbol epoch, a *pulse shape* (rectangular, raised-cosine, etc.) is applied. With BPSK signaling, the *data format* may be non-return to zero (NRZ) or Manchester encoded. The *modulation index* determines the fraction of total power that is allocated to an unmodulated carrier signal.

The carrier signal to be modulated is generated by an imperfect oscillator, whose quality can be measured by its spectrum, or by distilling its spectrum to

a single quantity such as *Allan deviation*, *phase noise* at a given offset, or *drift rate*. Ultrastable oscillators can achieve a phase noise of −100 dBc/Hz at a 1-Hz offset [12], although not all missions have the mass budget to carry one onboard the spacecraft.

There are several signal-dependent factors in the amplifier as well. Nominally, the amplifier output is larger than the input by the *gain* of the amplifier. However, depending on the input, distortion may affect the amplitude or phase. *Amplitude-modulation-to-amplitude-modulation (AM/AM) distortion* occurs when the amplitude of the amplifier output is not proportional to the amplitude of the amplifier input. *Amplitude-modulation-to-phase-modulation (AM/PM) conversion* occurs when variations in the input amplitude result in unwanted phase modulation. Additionally, the *group delay* is the rate of change of the total phase shift with respect to angular frequency, and the *polarization* (right- or left-handed, circular or elliptical) describes the time-varying direction and amplitude of the electric field vector propagated from the transmitter.

1.1.2.3 Channel. Typically, deep-space communications channels are quite benign, with *AWGN* being the dominating impairment. If *fading* is present, it may be due to *multipath interference* or *solar scintillations* caused by a small Sun–Earth–probe angle. *Doppler* affects carrier and timing parameters. The *angle of arrival*, *symbol timing*, and *carrier phase* are also modeled in the channel component of the dependency graph shown in Fig. 1-1.

1.2 Radio Receiver Architectures

1.2.1 A Conventional Radio Receiver

A functional block diagram of a radio receiver and decoder is shown in Fig. 1-2. Factors that are known a priori in a conventional radio are shown in ellipses, while the tasks it performs are shown in rectangles. A conventional radio receiver has complete a priori knowledge of the signal-dependent factors relating to the FEC encoder and modulator/amplifier components shown in Fig. 1-1. Only the channel-related factors are not completely known—although even those may be partially known through the use of predicts.

Knowledge of the transmitted signal parameters greatly simplifies the design and implementation of the receiver. For example, if a residual carrier is present, then the carrier phase-tracking loop may be a simple phase-locked loop (PLL); hence, a Costas loop need not be implemented. Or, if the modulation type is known to be BPSK, then the receiver need not include any processing of the quadrature component of the signal. Every rectangular block in Fig. 1-2 is similarly simplified by knowing the basic properties of the transmitted signal.

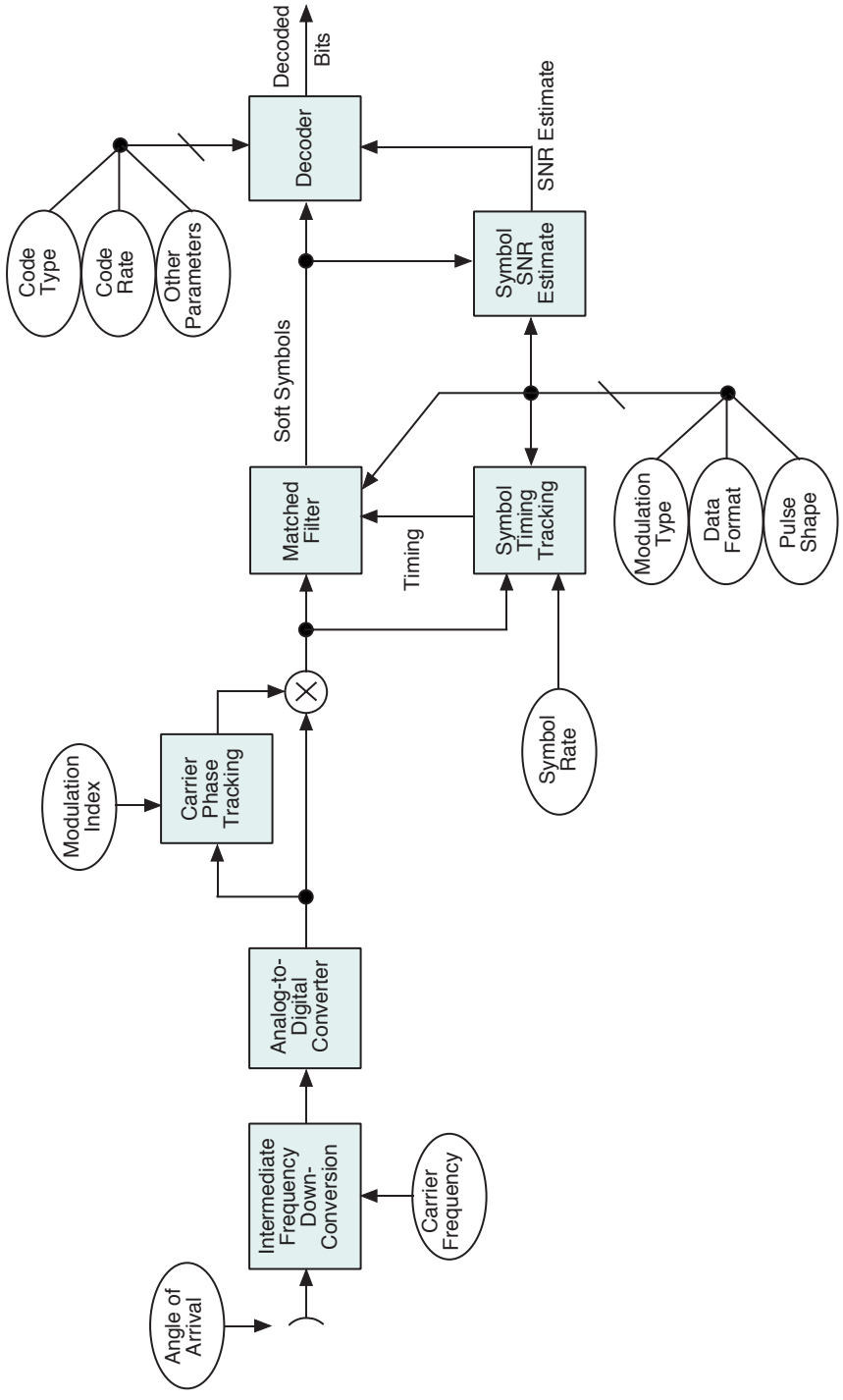


Fig. 1-2. Functional block diagram of a radio receiver. A conventional radio assumes a priori knowledge of all parameters shown in the ellipses, while an autonomous radio estimates them based on the incoming signal.

On the other hand, a conventional radio usually does not have much capability to receive signals types different from the single-signal type for which it was primarily designed, or, when it does have such capability, it requires specific pre-configuration according to a predetermined schedule. For example, if the radio can receive both suppressed-carrier and residual-carrier BPSK signals, it would typically carry both PLL and Costas loops, or a hybrid loop that incorporates both components and would have to be pre-configured to use the appropriate loop (or to set the relative gains in the hybrid loop) based on knowledge of when each type of signal will be arriving. This is the approach taken by the highly capable advanced receiver design [13] that eventually became the Deep Space Network Block V Receiver.

1.2.2 Electra

Electra is NASA's first highly capable software-defined radio [14]. Unlike other massively flexible radios, the Electra radio is an elegant, compact design based around a reprogrammable radiation-tolerant field programmable gate array (FPGA). The FPGA performs all the baseband processing for reception, including carrier tracking, timing recovery, and demodulation. It also includes all the baseband processing necessary for transmission.

Unlike the Block V Receiver, the massive capability of this radio is not achieved through multiple simultaneous implementations of tracking loops and demodulators for all the various signal types it might encounter in its lifetime. Rather, the radio is simply redefined in the same small footprint by reprogramming the baseband processor module. This compact, flexible design makes it ideally suited for in situ radios, and in fact, it is now the NASA standard in situ radio and will fly on the Mars Reconnaissance Orbiter, Mars Telecommunications Orbiter, and Mars Science Laboratory missions, among others.

1.2.3 An Autonomous Radio

The fundamental difference between a conventional radio, or even a software-defined radio such as Electra, and a truly autonomous radio is that an autonomous radio has the ability to recognize features of an incoming signal and to respond intelligently, without explicit pre-configuration or reprogramming to define the functions of the radio.

In an autonomous radio, the parameters shown in ellipses in the functional block diagram in Fig. 1-2 are assumed unknown a priori and must be determined based on the incoming signal. The quality of each of the estimators and classifiers of the autonomous radio is limited by its lack of knowledge of any of the other parameters. As such, the order in which the estimations/classifications are performed is critical. For example, it would not be feasible to classify the modulation type prior to classifying the data rate and obtaining the symbol tim-

ing. Using conventional estimation and tracking designs, one quickly gets into a chicken and egg problem, with nearly every estimator needing the output of the other estimators before it can make a maximum-likelihood (ML) estimate.

To resolve this problem, we have arranged the estimators/classifiers in the partially ordered set shown in Fig. 1-3, which defines the order in which they may be operated, at least sub-optimally, during the first iteration of estimation. Details about the order of estimation and the interaction of the estimators in the first and subsequent iterations are discussed in Chapter 11. There is a cluster of four estimators—data rate, SNR, pulse shape, and symbol timing—that are highly dependent on each other. For these, we propose a joint estimation algorithm, described in Chapter 7.

After the first estimate of the parameters is obtained, the estimates may be fed laterally and upward to other estimation modules to improve performance. For example, the modulation classifier operates noncoherently at first, without input from the phase-tracking loop, but once a phase estimate is available, it may switch to a better-performing coherent modulation classifier.

This approach yields a workable boot-strapping approach to estimating/classifying all of the parameters necessary for the proper operation of the entire receiver.

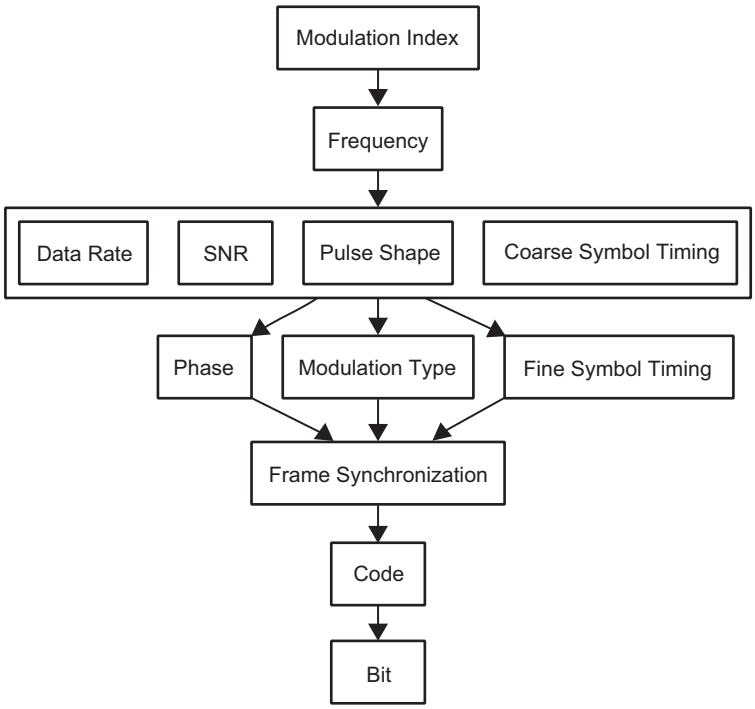


Fig. 1-3. The order of estimation in the autonomous radio.

1.3 Estimators and Classifiers of the Autonomous Radio

After identifying the proper order for the estimator modules shown in Fig. 1-3, a design for the individual modules can begin. At first glance, it may seem that some of these estimator modules are simply long-established, conventional designs. For example, phase-tracking loops have been designed and analyzed for nearly every reasonable signal type. However, the authors were unable to find any literature for the design of a phase-tracking loop for suppressed-carrier signals in which the modulation type is unknown. A loop is needed that works adequately for any phase-modulated signal, and which can improve its performance by later taking input from the modulation classifier when it starts producing an output.

The other seemingly standard modules had similar design challenges because of unknown signal attributes. Conventional implementations of frequency estimators, symbol-timing loops, and SNR estimators also assume the modulation type is known.

In addition, there are a number of estimators that are not conventional and occur only in an autonomous radio. These include the blocks that estimate or classify the data rate, modulation index, and modulation type. This monograph derives architectures for each of these from scratch, in most cases by formulating the ML criterion for the estimator and attempting to solve it analytically. This led to excellent solutions for modulation classification, SNR estimation, and frequency tracking. In some other cases, the ML solution was not tractable, and promising ad hoc schemes were identified.

We briefly summarize the status of the design and analysis of some of these estimators below.

1.3.1 Carrier Phase Tracking

In autonomous radio operation, it is desired that the receiver contain a carrier-synchronization structure that is generic in the sense that it is capable of tracking the carrier phase independently of the modulation. If the modulation is restricted to the M-PSK class, then it is possible to construct a universal structure that performs the carrier-synchronization function for all values of M. This structure is derived by first determining the maximum a posteriori (MAP) estimate of carrier phase based on an observation of the received signal, namely, M-PSK plus AWGN, and then using this to motivate a closed-loop carrier-synchronization loop. Such a structure, referred to as the *MAP estimation loop*, has been previously proposed in the literature for cases where the modulation is known beforehand. The most convenient form for use in the autonomous radio application is its simplification based on low SNR approximations applied to the nonlinearities inherent in the MAP phase estimate. When this is done, the error signal in the loop for M-PSK is of the form $\sin(M\phi)$, where ϕ is the phase error, which from

simple trigonometry can be written as $\sin(M\phi) = 2\sin[(M/2)\phi]\cos[(M/2)\phi]$. Thus, it is seen that the error signal in the loop for M-PSK is formed from the product of the error signal $\sin[(M/2)\phi]$ and the lock detector signal $\cos[(M/2)\phi]$ in the loop for $(M/2)$-PSK modulation. This simple relationship forms the basis for implementing a universal structure.

1.3.2 Modulation Classification

The autonomous radio determines the modulation type from the incoming signal. Approximations are derived of the ML classifier to discriminate between M-ary and M'-ary PSK transmitted on an AWGN channel and received noncoherently, partially coherently, or coherently, and when symbol timing is either known or unknown. A suboptimum classifier can be shown to be ten times less complex than the ML classifier and to have less than 0.1 dB of performance loss for symbol SNRs in the range $(-10,10)$ dB and any number of observed symbols. Other methods reduce complexity by a factor of 100 with less than 0.2 dB of performance loss. We also describe a classifier that does not require an estimate of the symbol SNR, and a new threshold optimization technique that improves the high-SNR performance of a previously published classifier. We discuss a classification error floor that exists for any classifier on any memoryless channel, even a noiseless one, by deriving a lower bound on the misclassification probability as a function of the number of observed samples.

1.3.3 Signal-to-Noise Ratio Estimation

In the design of receivers for autonomous operation, it is desirable that the estimation of SNR take place with as little known information as possible regarding other system parameters such as carrier phase and frequency, order of the modulation, data symbol stream, data format, etc. While the ML approach to the problem will result in the highest quality estimator, it typically results in a structure that is quite complex unless the receiver is provided with some knowledge of the data symbols typically obtained from data estimates made at the receiver (which themselves depend on knowledge of the SNR). Instead, we focused our attention on estimators that perform their functions without any data symbol knowledge and, despite their ad hoc nature, maintain a high level of quality and robustness with respect to other system parameter variations. One such ad hoc SNR estimator is the so-called *split-symbol moments estimator (SSME)* that forms its SNR estimation statistic from the sum and difference of information extracted from the first and second halves of each received data symbol. Our initial investigations focused on demonstrating that the scheme, which was previously investigated only for BPSK modulations, is readily applicable to the class of M-PSK ($M \geq 2$) modulations and furthermore showed that its performance is independent of the value of M! Even more generally, it was

pointed out that the complex symbol version of the SSME structure could also be used to provide SNR estimation for two-dimensional signal sets such as QAM. Performance results were obtained for a variety of different scenarios related to the degree of knowledge assumed for the carrier-frequency uncertainty and to what extent it is compensated for in obtaining the SNR estimate.

Following this, a modification of the conventional SSME architecture was discovered that provides significant improvement in performance (as measured by the variance of the estimator). The reconfiguration consists of partitioning the symbol interval into a larger (but even) number of subdivisions than the two that characterize the conventional SSME, where the optimum number of subdivisions depends on the SNR region in which the true SNR lies. It also was shown that these SNR regions can be significantly widened with very little loss in performance. Most important is the fact that, with this reconfiguration, the SNR estimator tracks the Cramer–Rao bound (with a fixed separation from it) on the variance of the estimator over the entire range of SNR values, whereas the conventional SSME deviates considerably from this bound at high SNR. Finally, an adaptive algorithm based on the modified SSME was developed that allows the system to automatically converge on the true SNR, starting with an initial guess (estimate) derived from a partition of two subdivisions.

1.3.4 Frequency Tracking

We present a robust frequency-tracking loop for a residual-carrier system that is capable of tracking the offset frequency without knowledge of received SNR. The proposed frequency-tracking loop can operate robustly not only over an AWGN channel but also over a Rayleigh fading channel. This loop does not require knowledge of carrier phase. We begin by deriving the likelihood function of the frequency, given the received observations. The derivative of this likelihood is then used as an error signal in a closed-loop structure, which therefore tends to converge near the ML estimate of the frequency. This design technique is similar to the one used for the MAP-motivated carrier phase-tracking loop. To reduce the implementation complexity with only a small loss in optimality, we simplified the derivation of the error signal.

1.4 An Iterative Message-Passing Architecture

As mentioned above, the autonomous radio begins by producing estimates at the highest level in Fig. 1-3 and then proceeding to progressively deeper levels. Initially, no estimator at an upper level can make use of any signal attribute estimated at a level beneath it. This limitation significantly impacts performance and is inherent to any non-iterative autonomous signal parameter estimation algorithm.

A fundamental innovation of the autonomous radio envisioned here is that, after each estimator completes its first estimate in the proper boot-strap order, the deeper-level estimators send soft information to the upper estimators. A second iteration then begins, wherein each estimator makes use of this additional extrinsic information to improve its performance. After several iterations, the message-passing system will reach a reasonable convergence. We have shown that such coupled systems are typically quite robust and can provide near-ML joint estimation/decoding [15–17].

We now informally describe a non-exhaustive list of the type of soft information that can be passed upward during the estimation iterations.

1.4.1 Messages from the Symbol-Timing Estimator

The symbol-timing module estimates the boundaries of symbol epochs and can produce a signal that indicates whether or not it is in lock. The lock indicator, which may be a soft value, can be fed up to the data rate classifier. For example, if the symbol-timing tracker is unable to lock onto symbol timing at one data rate, the data rate classifier can make use of that knowledge in reclassifying the data rate.

1.4.2 Messages from the Phase Tracker

The phase-tracking loop output can be used to generate a coherent reference that can be used to improve the symbol timing and SNR estimators, effectively improving the noncoherent performance to coherent performance.

1.4.3 Messages from the Modulation Classification

Estimates from the modulation classifier can assist in improving SNR, frequency correction, data rate, and symbol-timing estimators. The likelihood functions for each modulation have expected values that obey a known relationship to the SNR and symbol timing—for example, the modulation classification becomes more certain with increasing SNR and number of symbol observations. If the observed modulation-type likelihoods are inconsistent with the estimates from the SNR and symbol-timing modules, the likelihoods can be fed back to those modules so they can revise their estimates.

1.4.4 Messages from the Decoder

The output of the decoder includes likelihoods for each message bit. Depending on the code, it is usually simple to hard-limit these likelihoods and test if the result is a codeword. Typically, codes are designed so that the undetected

probability of codeword error is 10^{-10} or lower, which implies that if the output is a codeword, it is nearly certainly the correct one, and no further iterations of the radio are necessary.

If the correct codeword is not obtained, then the bit likelihoods can be used to generate a soft data-wipe of the received signal. This makes the signal more like a continuous wave (CW) signal, which will allow the SNR, frequency, and phase estimates to be substantially improved, which will in turn produce better inputs for the decoder to operate on in its second iteration. This behavior of coupled or iterative estimation has been observed before [16,18].

1.5 A Demonstration Testbed

NASA is developing a software demonstration testbed of the autonomous radio described in this chapter. The testbed contains two parts. In the first part, the attributes of the signal may be configured, including the data rate, pulse shape, data format, modulation type, and so forth. Channel effects such as SNR and Doppler also can be configured. Based on these settings, a simulated signal is generated. This is similar to the signals used in software simulations of the Electra modulator, for example.

The second part of the testbed implements the autonomous receiver estimators and classifiers. In most cases, these are either ML or motivated by low-complexity approximations to ML estimation or hypothesis testing. The testbed produces a graphical output that illustrates the performance of the various estimators and compares them to performance bounds, if such bounds are available.

References

[1] J. Hamkins, M. Simon, S. Dolinar, D. Divsalar, and H. Shirani-Mehr, "An Overview of the Architecture of an Autonomous Radio," *The Interplanetary Network Progress Report*, Jet Propulsion Laboratory, Pasadena, California, vol. 42-159, pp. 1–14, November 15, 2004.
http://ipnpr.jpl.nasa.gov/progress_report/42-159/159H.pdf

[2] C. D. Edwards, A. Barbieri, E. Brower, P. Estabrook, R. Gibbs, R. Horttor, J. Ludwinski, R. Mase, C. McCarthy, R. Schmidt, P. Theisinger, T. Thorpe, and B. Waggoner, "A Martian Telecommunications Network: UHF Relay Support of the Mars Exploration Rovers by the Mars Global Surveyor, Mars Odyssey, and Mars Express Orbiters, IAC-04-M.5.07," *International Astronautical Congress 2004*, pp. 1–11, October 2004.

[3] J. Guinn, "Phoenix Plans for Relay Use," *Mars Network Workshop*, June 2004.

[4] R. Barry, "MSL Plans for Relay Use," *Mars Network Workshop*, June 2004.

[5] J. Oberg, "Titan Calling," *IEEE Spectrum*, vol. 41, no. 10, pp. 28–33, 2004.

[6] Consultative Committee for Space Data Systems, *CCSDS 101.0-B-6: Telemetry Channel Coding*, Blue Book, issue 6, October 2002.

[7] Consultative Committee for Space Data Systems, *CCSDS 211.0-B-1: Proximity-1 Space Link Protocol*, Blue Book, issue 1, October 2002.

[8] S. Lin and D. J. Costello, Jr., *Error Control Coding: Fundamentals and Applications*, New Jersey: Prentice-Hall, 1983.

[9] Consultative Committee for Space Data Systems, *CCSDS 201.0-B-3: Telecommand Part 1—Channel Service*, Blue Book, issue 3, June 2000.

[10] J. G. Proakis, *Digital Communications*, New York: McGraw Hill, Inc., third ed., 1995.

[11] G. Ungerboeck, "Channel Coding with Multilevel/Phase Signals," *IEEE Transactions on Information Theory*, vol. IT-28, pp. 55–67, January 1982.

[12] S. W. Asmar, "Characteristic Trends of Ultrastable Oscillators for Radio Science Experiments," *The Telecommunications and Data Acquisition Progress Report 42-129, January–March 1997*, Jet Propulsion Laboratory, Pasadena, California, pp. 1–5, May 15, 1997.
http://ipnpr.jpl.nasa.gov/progress_report/42-129/129F.pdf

[13] S. Hinedi, "A Functional Description of the Advanced Receiver," *The Telecommunications and Data Acquisition Progress Report 42-100, October–December 1989*, Jet Propulsion Laboratory, Pasadena, California, pp. 131–149, February 15, 1990.
http://ipnpr.jpl.nasa.gov/progress_report/42-100/100K.PDF

[14] C. D. Edwards, T. C. Jedrey, E. Schwartzbaum, A. S. Devereaux, R. DePaula, M. Dapore, and T. W. Fischer, "The Electra Proximity Link Payload for Mars Relay Telecommunications and Navigation, IAC-03-Q.3.A06," *International Astronautical Congress 2003*, September–October 2003.

[15] D. Divsalar and F. Pollara, "Turbo Trellis Coded Modulation with Iterative Decoding for Mobile Satellite Communications," *IMSC*, 1997.

[16] J. Hamkins and D. Divsalar, "Coupled Receiver-Recoders for Low Rate Turbo Codes," *Proceedings of the IEEE International Symposium on Information Theory (ISIT)*, June 2003.

[17] J. Gunn, K. Barron, and W. Ruczczyk, "A Low-Power DSP Core-Based Software Radio Architecture," *IEEE Journal on Selected Areas in Communications*, vol. 17, pp. 574–590, April 1999.

[18] J. Hamkins, "A Joint Receiver–Decoder for Convolutionally Coded Binary Phase-Shift Keying (BPSK)," *The Telecommunications and Mission Operations Progress Report 42-139, July–September 1999*, Jet Propulsion Laboratory, Pasadena, California, pp. 1-23, November 15, 1999.
http://ipnpr.jpl.nasa.gov/progress_report/42-139/139B.pdf

Chapter 2
The Electra Radio

Edgar Satorius, Tom Jedrey, David Bell, Ann Devereaux,
Todd Ely, Edwin Grigorian, Igor Kuperman,
and Alan Lee

This chapter provides an overview of the Electra radio [1]. This is the first programmable software radio that has been developed for space missions. The radio currently accommodates digital binary phase shift keying (BPSK) modulation with both suppressed- and residual-carrier capabilities. The radio is designed to operate over a wide range of data rates from 1 kilobit per second (kbps) to 4 megabits per second (Mbps) and must accommodate frequency uncertainties up to 20 kHz with navigational Doppler tracking capabilities. As such, it is highly programmable and incorporates efficient front-end digital decimation architectures to minimize power consumption requirements. The Electra radio uses field programmable gate array (FPGA) technology to provide the real-time and programmable capabilities. Emphasis in this chapter is focused on the programmable features of the software algorithms implemented in the Electra transceiver as well as the hardware functional specifications.

The objective of the Electra radio, which is based on the original Micro Communications and Avionics Systems (MCAS) prototype [2], is to develop programmable telecommunications systems to meet the unique needs of the National Aeronautics and Space Administration (NASA) for low-power space and planet-surface communications. NASA is moving into an era of much smaller space exploration platforms that require low mass and power. This new era will usher in increasing numbers of miniature rovers, probes, landers, aerobots, gliders, and multiplatform instruments, all of which have short-range communications needs (in this context short-range is defined as non-deep-space links). Presently these short-range (or in situ) communications needs are being met by a combination

of modified commercial solutions (e.g., Sojourner) and mission-specific designs [e.g., Mars '96, Mars '98, Space Technology-2 (ST-2)]. The problem with commercial-based solutions is that they are high-power, high-mass, and single-application-oriented; achieve low levels of integration; and are designed for a benign operating environment. The problem with the mission-specific designs is that the resultant short-range communication systems do not provide the performance and capabilities to make their use for other missions desirable.

Electra is primarily targeted at potential Jet Propulsion Laboratory (JPL) users in the space exploration arena, such as the Mars Exploration Office, which will use this for various microspacecraft short-range communication links, such as orbit–lander, orbiter–rover, orbiter–microprobe, orbiter–balloon, orbiter–sample return canister), ST-4 (orbiter to/from lander link), ST-3 (inter-spacecraft links), and multiple proposed Discovery missions (e.g., balloons, gliders, probes). Electra also has applicability to any space mission that has a short-range communications requirement, such as International Space Station intravehicular and extravehicular wireless communications, X-33 wireless sensor and short-range ground links, Moon missions (e.g., Moonrise), etc. To this end, Electra has been designed to be compatible with the Proximity-1 Space Link Protocol [3].

The ultimate goal of Electra is to achieve a higher level of system integration, thus allowing significant mass, power, and size reductions, at lower cost, for a broad class of platforms. The realization of this goal has resulted in maximizing the transceiver functions performed in the digital domain. These digital functions are implemented with space-qualified FPGA technology. The receiver functions that must be in the analog domain consist primarily of the radio frequency (RF) up- and down-conversions. A space-qualified RF design has been developed through proper selection of parts.

The functionality of the digital portion of the Electra transceiver is exhibited in the block diagram presented in Fig. 2-1. Emphasis in this chapter will be focused primarily on the programmable digital portions of the Electra transceiver (receiver and digital modulator). The programmable digital receiver front-end processing is described in Section 2-1, and the Electra data demodulator is discussed in Section 2-2. Finally, the programmable digital modulator is described in Section 2-3.

2.1 Electra Receiver Front-End Processing

In this section, we describe the Electra receiver front end. With reference to Fig. 2-1, this comprises the automatic gain control (AGC), the analog-to-digital converter (ADC), and the digital downconverter/decimator. These are described separately in this section.

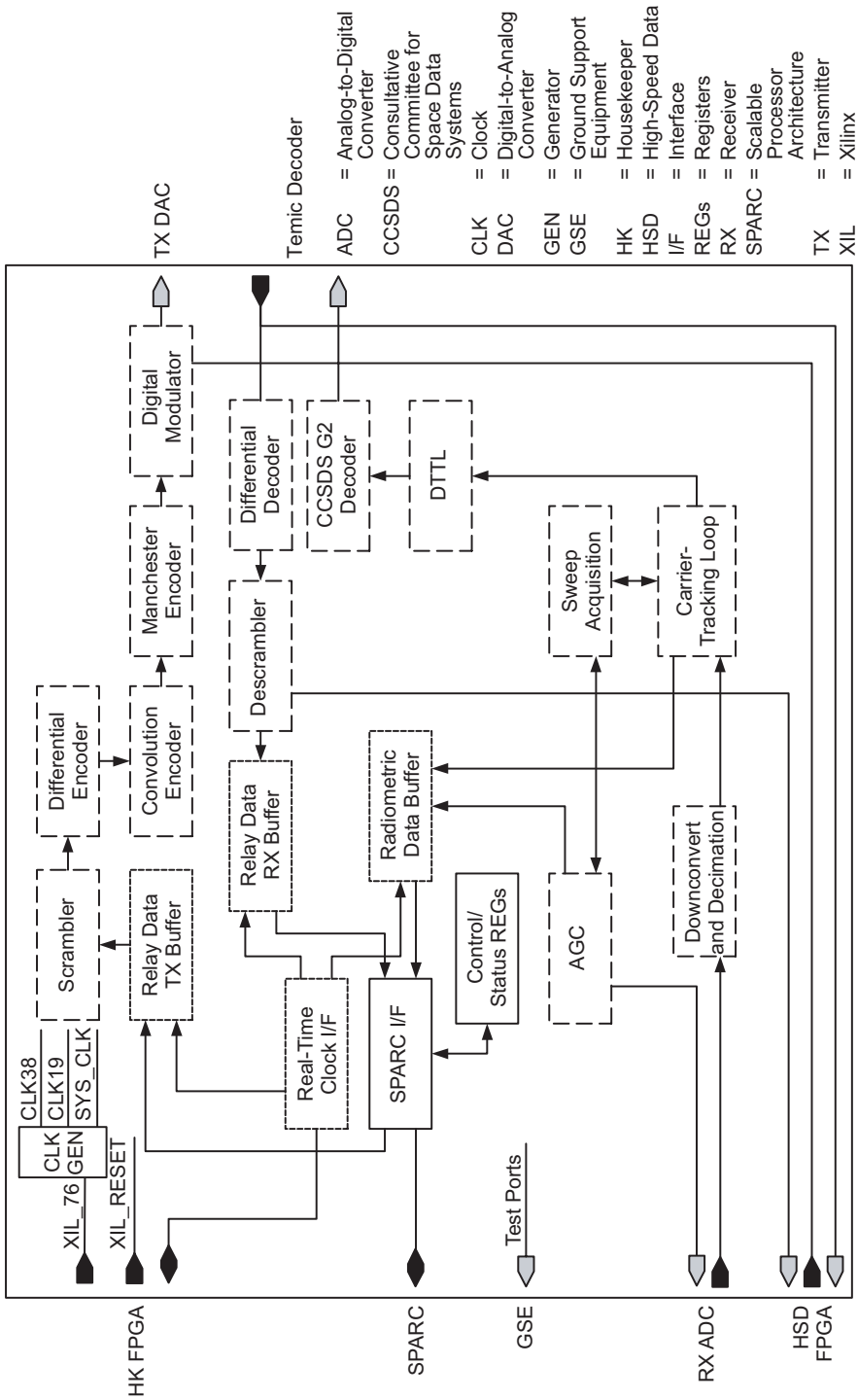


Fig. 2-1. Electra transceiver block diagram.

2.1.1 AGC

The AGC controls the voltage level input to the ADC based on a control voltage signal generated digitally in the FPGA. In particular, the AGC is based on a single feedback control loop design with the AGC control voltage extending back from the carrier-tracking loop (CTL) arm-filter outputs, as indicated in Fig. 2-2. The digital AGC error signal, E_{AGC}, is generated from the CTL arm-filter outputs, I and Q, via

$$E_{\mathrm{AGC}} = K_{\mathrm{gain}} \cdot (1 - \sqrt{I^2 + Q^2}) \tag{2-1}$$

where K_{gain} controls the time constant of the AGC as well as the variance of the resulting amplitude gain estimate. Note that K_{gain} is the only programmable AGC constant. Typically, $K_{\mathrm{gain}} = 2^{-15}$ provides a reasonable compromise between a fast AGC response time (<10 ms) and a low noise gain estimate.

The AGC error signal, Eq. (2-1), is chosen such that the AGC forces the complex magnitude of the CTL arm-filter outputs, $\sqrt{I^2 + Q^2}$, to be unity on average. This in turn helps to regulate the CTL loop bandwidth over a reasonably wide range of input signal levels. The error signal, Eq. (2-1), is integrated in the AGC loop filter, i.e.,

$$V_{\mathrm{out}}[k+1] = V_{\mathrm{out}}[k] + E_{\mathrm{AGC}}[k] \tag{2-2}$$

and the magnitude of the result, $|V_{\mathrm{out}}|$, is used to generate the AGC gain, K_{AGC}, via the nonlinear transfer curve, $f(\cdot)$, i.e.,

$$K_{\mathrm{AGC}}\ (\mathrm{dB}) = f\big(|V_{\mathrm{out}}|\big) \tag{2-3}$$

This gain then is used to scale the AGC input.

A critical issue with this approach is the impact of the AGC on the operation of the ADC as well as the internal digital arithmetic implemented in the FPGA. Ideally the input ADC voltage is scaled to achieve an optimal trade-off between ADC quantization noise and clipping distortion. In contrast, the AGC loop attempts to maintain the complex magnitude of the CTL arm-filter outputs to be unity on average. Thus, there is no guarantee that this criterion of unity root-mean-square (rms) CTL arm-filter outputs will enable the ADC to operate at its optimal input scaling (loading) point or even prevent the ADC from saturating. To alleviate this situation, fixed gains are distributed throughout the digital data paths. These gains are programmable, dependent upon the data rate, and used for purposes of minimizing the effects of digital quantization noise and saturation.

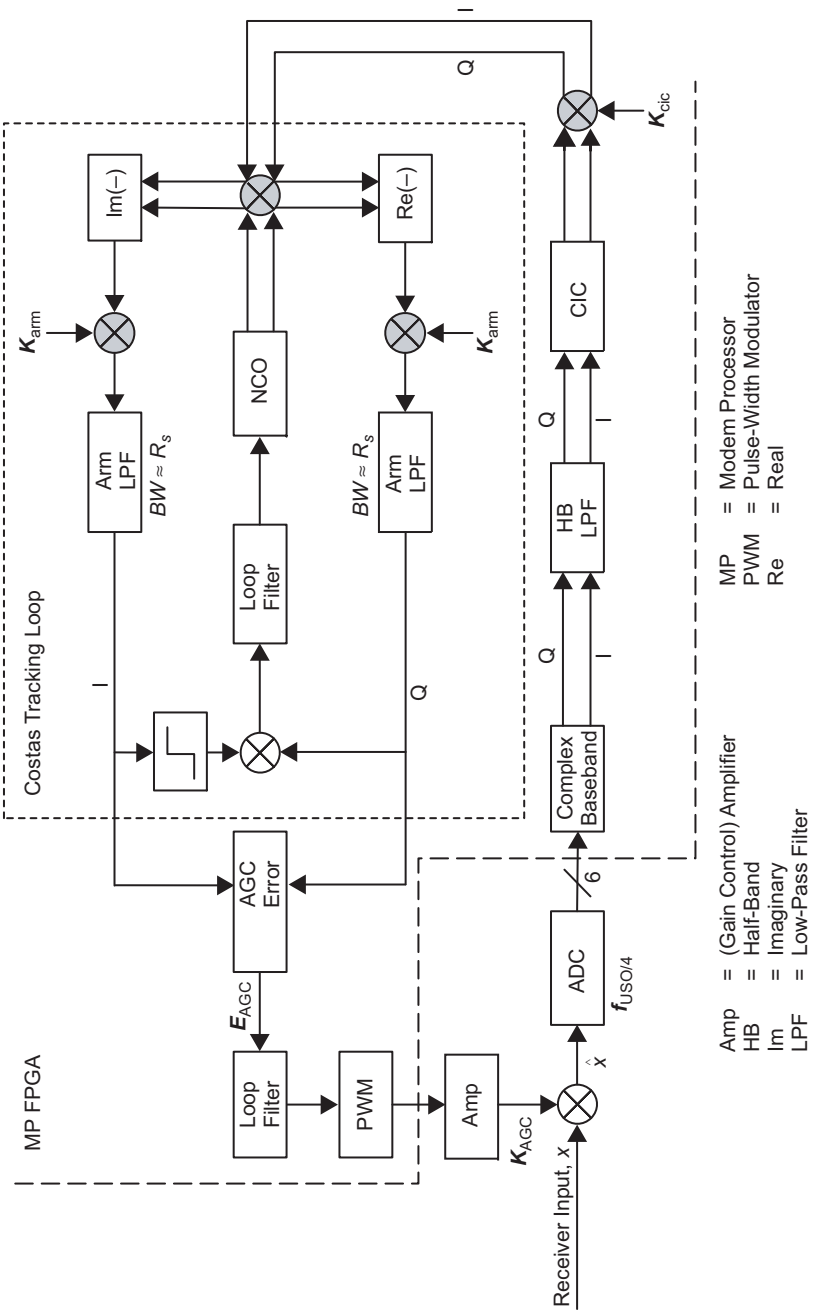


Fig. 2-2. AGC control loop.

2.1.2 ADC

The Electra receiver employs first-order, bandpass sampling wherein the intermediate frequency (IF) band is mapped directly down to digital baseband. Denoting the IF and ADC sampling frequencies by f_{IF} and F_s, respectively, then as long as the frequency band $f_{\mathrm{IF}} - F_s/4 \le f \le f_{\mathrm{IF}} + F_s/4$ coincides with one of the image bands, $kF_s/2 \le f \le (k+1)F_s/2$ for some integer k, the input at f_{IF} will be mapped into the baseband interval $0 \le f \le F_s/2$ as a result of bandpass sampling. This leads to the following condition on f_{IF} and F_s:

$$f_{\mathrm{IF}} = (2n+1) \cdot \frac{F_s}{4} \tag{2-4}$$

where n is a positive integer. Choosing f_{IF} and F_s to satisfy Eq. (2-4) guarantees that the IF frequency will be mapped down to the center of the Nyquist band, i.e., down to $F_s/4$. Furthermore, to maintain the lowest possible ADC sample rate and avoid aliasing, it is desired that F_s just exceed twice the IF filter bandwidth ($2B_{\mathrm{IF}}$).

In designing the bandpass sampling system, the ADC sampling rate was chosen to accommodate an integral, power-of-two number of samples per symbol at all symbol rates: 4.096 megasymbols per second (Msps), 2.048 Msps, $\cdots$, 1 kilosymbol per second (ksps). To achieve a minimum of 4 samples per symbol at the highest symbol rate of 4.096 Msps, an ADC sampling rate in excess of 16.384 MHz is required. In the Electra receiver, F_s is a sub-multiple of the master clock frequency that is provided by an ultra-stable oscillator (USO). A nominal USO frequency, F_{USO}, is 76.72 MHz, and thus the lowest sub-multiple of 76.72 MHz that meets the 4.096-MHz requirement is $F_s = 19.18$ MHz, as indicated in Fig. 2-1. This in turn results in a digital IF frequency of $F_s/4 = 4.795$ MHz. Given F_s, admissible IF frequencies are obtained from Eq. (2-4). An IF near the standard 70 MHz is desired. The closest admissible IF frequency satisfying Eq. (2-4) occurs when $2n + 1 = 15$, corresponding to $f_{\mathrm{IF}} = 15 \cdot 19.18/4$ MHz $= 71.775$ MHz.

Note that different ADC sampling rates may be accommodated simply by choosing different sub-multiples K of the USO frequency, i.e., $F_s = F_{\mathrm{USO}}/K$. This would allow the Electra receiver the flexibility of operating the ADC at slower rates for lower data rate missions, which in turn saves power. However, this flexibility is quite limited since changing the data rate typically requires a change in the IF frequency by virtue of Eq. (2-4), and the analog IF frequency is fixed in the Electra design. For example, if $K = 8$, then $F_s = F_{\mathrm{USO}}/8 = 9.57$ MHz, and the nominal IF frequency, $f_{\mathrm{IF}} = 71.775$ MHz is no longer an odd multiple of F_s. This of course would result in severe digital aliasing.

2.1.3 Digital Downconversion and Decimation

Programmable digital downconversion and decimation directly follow the ADC. The digital complex baseband downconversion scheme depicted in Fig. 2-3 is used and comprises Eq. (2-1), digital complex mixing from $F_s/4 = 4.795$ MHz down to baseband, followed by Eq. (2-2), and digital decimation via a first-order, cascaded integrator-comb (CIC) filter [4,5]. Note that the digital mixing functions do not require multiplication, and furthermore the CIC filters are multiplierless; thus, the entire structure can be implemented efficiently in the FPGA.[1]

The decimation factor M is programmable and is dependent upon the input data rate. To accommodate symbol-timing recovery, M typically is chosen so that there are at least 16 samples per symbol after decimation, except at the highest data rates. So at 1.024 Msps, 2.048 Msps, or 4.096 Msps, M will nominally be set to 1 (no decimation), in which case the remainder of the digital receiver (CTL, symbol-timing recovery, etc.) will run at the input sampling rate, F_s. As the data rate is lowered below $R_s = 1.024$ Msps down to 8 ksps, M is increased proportionately such that

$$\frac{F_s}{R_s \cdot M} = \frac{F_s}{1.024 \text{ MHz}} \tag{2-5}$$

Below $R_s = 8$ ksps, M remains fixed at 128 to accommodate Doppler offsets. Note that as M increases up to 128, more of the input noise to the ADC is filtered out by the CIC filters, thereby reducing the total CIC output power. This is compensated by the AGC as well as programmable, fixed gains following the CIC filters.

2.2 Electra Demodulation

In this section, the various programmable elements of the demodulation process are presented, including the frequency-acquisition and carrier-recovery loop (Section 2.2.1), the Doppler frequency extraction for navigation (Section 2.2.2), and the symbol-timing recovery (Section 2.2.3). In addition, a description of the Electra symbol signal-to-noise (SNR) estimator and convolutional node synchronization (sync) algorithm also are provided (Section 2.2.4). The remaining demodulation functions depicted in Fig. 2-1 (de-scrambling and decoding) are primarily non-programmable, standard functions and thus are not discussed here. In fact, the decoder is implemented outside of the FPGA.

---

[1] Additional, fixed digital (multiplierless) half-band filters [5] are also included in the digital downconverter to remove images—especially when digital decimation is not used ($M = 1$).

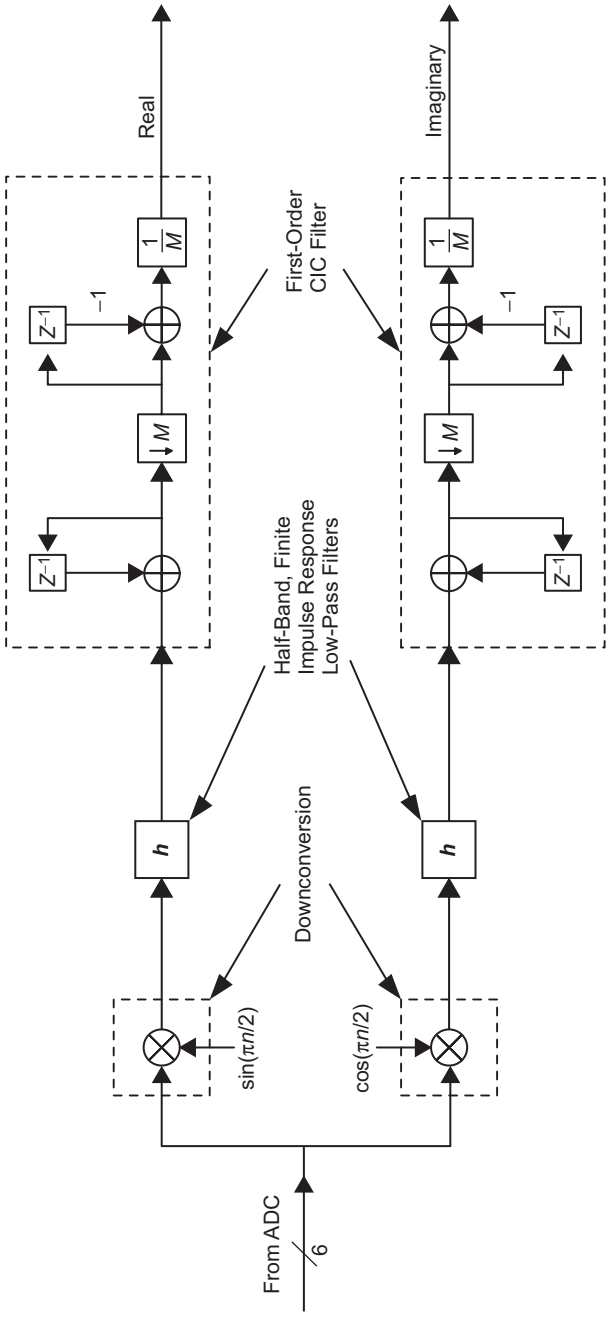


Fig. 2-3. Digital complex basebanding and decimation.

2.2.1 Frequency-Acquisition and Carrier-Tracking Loop

The Electra frequency acquisition and CTL are fully programmable and are designed to acquire and track the frequency and phase of the received signal. The signal can be suppressed-carrier BPSK, residual-carrier BPSK with a modulation index of $\pi/3$, or unmodulated.[2] The CTL operates at all of the required symbol rates from 1 ksps to 4 Msps, signal-to-noise ratios, and CIC-filter-decimated sampling rates. It tracks the carrier reliably when the received signal strength varies over many orders of magnitude. The tracking-loop bandwidth is programmable from 10 Hz to 10 kHz to meet the tracking and acquisition requirements for various communications scenarios. In addition, the frequency acquisition acquires and tracks received signals with maximum frequency offsets of ± 20 kHz, which are typical of ultra-high frequency (UHF), but is programmable to accommodate larger offsets, e.g., at S-band (around 2 GHz) or X-band (around 8 GHz). The CTL also supports navigation by supplying the instantaneous phase of the received signal.

Figure 2-4 shows the block diagram of the Electra CTL, including frequency acquisition. The loop follows the ADC, digital downconverter, and CIC decimation filter as indicated in Fig. 2-1. The complex baseband loop input is multiplied by the complex output of the numerically controlled oscillator (NCO). The product of the complex multiplication is split into the real and the imaginary data paths.

Both the real and imaginary signals are filtered by a pair of identical low-pass arm filters, $G(f)$, with a programmable cut-off frequency. After the arm filters, one or both of the arm filter outputs is used to form the input to the loop filter, $F(f)$, depending on whether the tracking loop is operated in the suppressed-carrier- (Costas) or the residual-carrier-tracking mode (labeled PLL mode in Fig. 2-4). There are three switches (SWs) in the CTL. SW1 and SW2 are selected depending on whether the loop is operated in the Costas loop or the residual-carrier-tracking mode. SW3 is used in the Costas-loop mode. Its position is chosen depending on whether the tracking loop is in the acquisition mode or the tracking mode.

Aside from SW1–3, the CTL loop bandwidth, B_L, as well as the arm filters are programmable. The arm filters are discrete implementations of a first-order low-pass Butterworth filter with programmable cut-off frequency. The arm filters are used to reduce noise in the carrier-tracking loop, but the cut-off frequency should not be so low that the signal power is reduced appreciably by the arm filters. It is found that the cut-off frequency that minimizes the tracking-loop error for the arm filters is approximately equal to the received symbol rate, R_s, for non-return to zero (NRZ)-coded data. Therefore, for the Electra receiver, the

[2] Currently, the Electra FPGA is being re-programmed to also accommodate suppressed-carrier quadrature phase-shift keying (QPSK).

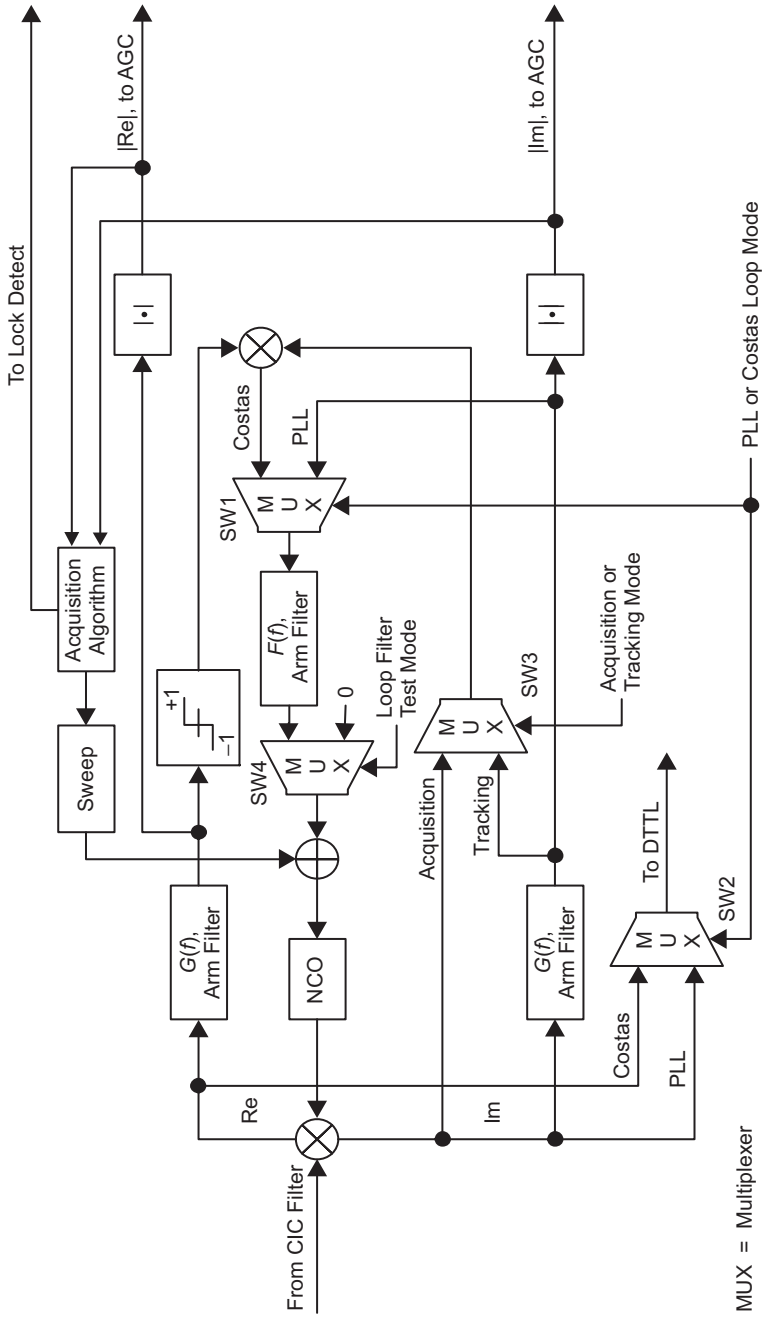


Fig. 2-4. Block diagram of the Electra carrier-tracking loop.

cut-off frequency needs to be programmable between $f_s/128$ and $f_s/4$, where $f_s = F_s/M$ is the decimated sampling rate.

The tracking-loop bandwidth, B_L, should be chosen small enough to provide sufficient loop SNR,

$$\rho \equiv \frac{P \times S_L}{N_0 B_L} \tag{2-6}$$

(P is the signal power input to the loop, S_L is the loop squaring loss, and $N_0/2$ is the two-sided noise power spectral density), and yet large enough to reduce acquisition time. Maintaining a loop SNR ρ in the 15- to 20-dB range requires that B_L vary between 10 Hz and 10 kHz, depending on the data rate and the symbol energy-to-noise spectral level, $E_s/N_0 = P/(N_0 \cdot R_s)$. For example, $B_L = 10$ kHz is acceptable at $R_s = 4.096$ Msps and $E_s/N_0 = 0$ dB (corresponding to a 23-dB loop SNR when $S_L = 3$ dB), whereas $B_L = 10$ Hz is more appropriate at $R_s = 1$ kHz.

The CTL bandwidth thus is chosen as the largest value that satisfies the 15- to 20-dB loop SNR requirement (assuming $E_s/N_0 = 0$ dB). These values for B_L are pre-computed as a function of the data rate and are programmed into the FPGA. In general, two sets of B_L settings are programmed: one for tracking and one for acquisition. The frequency-acquisition algorithm is fully programmable and uses simple saw-tooth sweeping. In particular, the frequency sweep starts at F_{init} (a programmable input parameter) and is incremented in frequency steps, f_{step}, i.e., $F_{acq_{n+1}} = F_{acq_n} + f_{\text{step}}$. The f_{step} is also a programmable input parameter that is typically set to $0.375 \cdot B_L$. At each frequency step, either the real arm output from the CTL, I_n (residual-carrier mode), or the difference between the magnitude of the real and imaginary arm outputs from the CTL, $|I_n| - |Q_n|$ (suppressed-carrier mode), is accumulated over N_{dwell} samples (at the decimated sampling rate) and compared to a threshold Z_{thresh}.

Both N_{dwell}, the number of samples per dwell, and Z_{thresh} are programmable input parameters that are pre-computed based on data rate and modulation type (residual or suppressed carrier). Once Z_{thresh} is exceeded, the sweep is terminated and data demodulation begins. If the threshold is not exceeded, the sweep continues to a maximum of $N \times f_{\text{step}}$ (programmable input parameter dependent upon the sweep range and CTL bandwidth B_L), at which point the sweep returns to the starting frequency, F_{init}, and is repeated until the threshold test succeeds. By re-programming the various input parameters (f_{step}, N, N_{dwell}, and Z_{thresh}), a wide range of Doppler search uncertainties and data rates can be accommodated.

2.2.2 Navigation: Doppler Phase Measurement

Missions like the Mars Relay Orbiter will be required to provide Doppler estimates derived from the received signal. This section describes how Electra Doppler frequency estimates are obtained. The method described herein is applicable to either the suppressed- or residual-carrier CTL mode of operation. The technique basically derives the Doppler frequency estimate from the difference between two instantaneous phase outputs from the phase register of the NCO. The resulting frequency estimate is equivalent to counting the elapsed phase cycles over a fixed time interval, T.

The Electra Doppler frequency estimate, f_{est}, is given by

$$f_{est} \equiv \frac{\theta(t+T) - \theta(t)}{2\pi T} \tag{2-7}$$

where T is the time between two instantaneous NCO phase measurements, $\theta(t)$ and $\theta(t+T)$. Assuming that T is sufficiently large such that $\theta(t+T)$ and $\theta(t)$ are independent (i.e., $T \gg 1/B_L$), then the variance of f_{est}, σ_f^2, is approximated by

$$\sigma_f^2 = \frac{2\sigma_\theta^2}{(2\pi T)^2} \tag{2-8}$$

where σ_θ^2 denotes the variance of the NCO phase measurements. Simulation results have verified the accuracy of Eq. (2-8) for both the suppressed- and residual-carrier CTL modes [2]. As seen from Eq. (2-8), the standard deviation of the Doppler frequency estimation error is inversely proportional to T. Thus, T should be as large as possible while still yielding a meaningful frequency estimate. Typically, T is on the order of 10s to 60s for a 10-minute pass [6]. It should be noted that a model suitable for operational use requires a more detailed development than is presented here [6].

2.2.3 Symbol-Timing Recovery

The Electra symbol-tracking loop is based on a digital data-transition tracking loop (DTTL) with a window size of half-symbol period [7]. The block diagram shown in Fig. 2-5 includes both the DTTL and the DTTL lock-detection logic and provides the word lengths used in the DTTL implementation, e.g., the notation "⟨6,1,t⟩" indicates that the data path is 6 bits wide with 1 integer bit and twos complement representation. The notation "⟨6,1,u⟩" also indicates that the data path is 6 bits wide with 1 integer bit but that the arithmetic representation is unsigned. The primary function of the DTTL is to synchronize the receiver

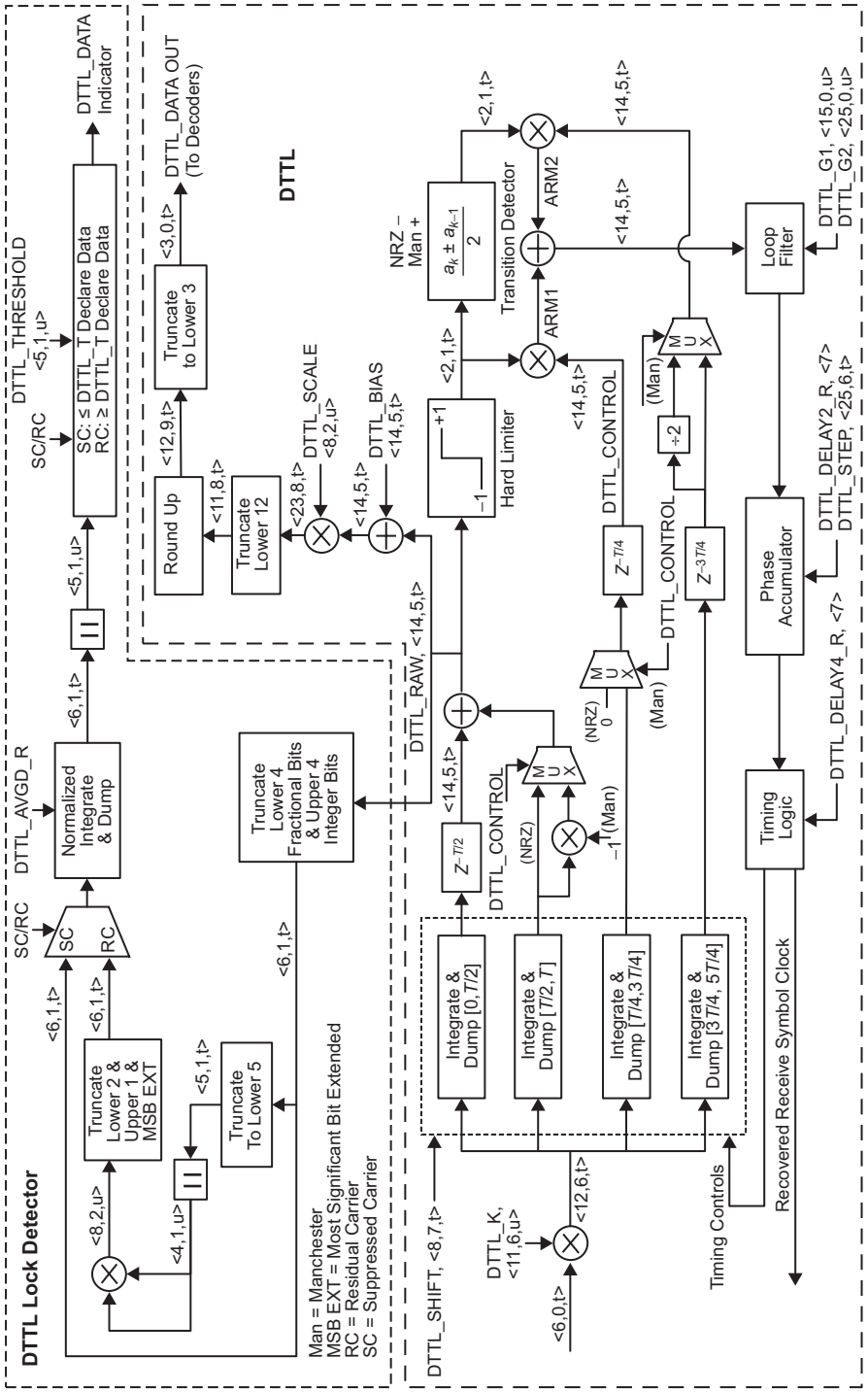


Fig. 2-5. Block diagram of the Electra symbol synchronization architecture.

logic to the received bit stream with the aid of the symbol transitions. The loop is designed to track the symbol boundaries for both NRZ and Manchester-coded data. As a result, a window size of 1/2 a symbol is chosen for the design. The DTTL lock-detection logic is used to determine if the received bit stream has only the carrier or carrier with active symbols. It is used by the downstream Viterbi decoders and to determine when the output of the DTTL has valid symbols.

The DTTL begins with a scaling stage. The only purpose for this stage is to normalize the amplitude of the input to the DTTL to 1. The scaling factor is entered via a programmable input parameter called "DTTL_K." The values for this parameter are obtained via simulation. It is a function of data rate and signal-to-noise ratio. Varying this scaling factor can change the bandwidth of the DTTL. The constant is pre-computed as a function of data rate and is programmed into the FPGA.

The next stage of the DTTL contains a set of integrators. Each integrator is designed to integrate the symbol level over a different segment of the symbol period. Since the window size of this DTTL is 1/2 a symbol, four integrators must be used. Two of them are used for detecting symbol transitions at symbol boundaries, and two are used for the mid-symbol boundaries. The timing of these integrators is controlled by the DTTL timing logic. "DTTL_SHIFT" is a programmable input parameter used to scale the integrator output by $1/2^{\text{DTTL\_SHIFT}}$. Again, DTTL_SHIFT is pre-computed as a function of data rate and is programmed into the FPGA.

The DTTL employs a second-order loop that requires the programmable loop bandwidth input parameter, B_{LD}. Like the CTL bandwidth parameter, the DTTL bandwidth should be chosen small enough to provide sufficient DTTL SNR yet large enough to track data rate changes—which is especially important given that the master (e.g., USO) clock is not necessarily an integral multiple of the data rate. Based on measurements, B_{LD} is computed as a function of data rate and is programmed into the FPGA. There are various other programmable DTTL parameters used to control the integrators and the DTTL phase accumulator (Fig. 2-5). All of these are pre-computed as a function of data rate and programmed into the FPGA.

Finally we describe the DTTL lock detector. A typical Proximity-1 session [3] starts with carrier-only transmission. Therefore, carrier-lock detection alone cannot determine the presence of valid data. This is the purpose for introducing the DTTL-lock detector. It is used to calculate the "power" of the received bit stream. The difference between the current power level and that for the carrier-only case hence can be used to determine the presence of valid data. The top half of Fig. 2-5 depicts the design of the DTTL lock detector. The DTTL lock detector is essentially a power-threshold detector. It converts the signal

level to signal power and compares that against a user input value to determine the lock detection.

Due to the difference in format between the data from the residual-carrier mode (Manchester coded) and those from the suppressed-carrier mode (NRZ coded), the raw data from the DTTL are reformatted before its power can be calculated. The next step is to calculate the respective signal power. The output of this stage is sent to an integrate-and-dump (or sum-and-dump) logic that produces an averaged power measurement on the incoming bit stream. The size of the sample space used for calculating the average is determined by the programmable input parameter "DTTL_AVGD_R." There are only 8 possible values for "DTTL_AVGD_R" ranging from 64 samples to 8192 samples. These are precomputed as a function of data rate and are programmed into the FPGA.

The absolute value of the output from the integrate-and-dump is checked against a programmable threshold called "dttl_threshold." Taking the absolute value of the integrate-and-dump output causes the drop of the sign bit and reduces the data size by 1 bit. The output of the integrate-and-dump is always positive, and the sign bit is superfluous. Depending on whether the received signal is in the residual-carrier mode or the suppressed-carrier mode, the threshold comparison is performed differently. In the case of the suppressed-carrier mode (NRZ), the averaged signal power must be less than or equal to the threshold to be indicative of the valid data. For the residual-carrier case (Manchester), the averaged signal power must be greater than or equal to the threshold to indicate the presence of valid data. The final DTTL data detection flag (dttl_lock) is gated by carrier lock to avoid false indication. The Viterbi decoder is constantly monitoring this flag to determine when the data are valid for processing.

2.2.4 Viterbi Node Sync and Symbol SNR Estimation

The Viterbi decoder is utilized to provide error correction of the received symbols that were convolutionally encoded at the transmitter. Proper Viterbi decoding requires that the correct pair of demodulated data symbols be assigned to the Viterbi code segment. This is termed Viterbi node sync. The method used by Electra to achieve node sync is based on re-encoding [8] as depicted in Fig. 2-6. The Viterbi decoding and re-encoding functions are performed outside of the FPGA on the commercial Temic chip. As is seen from Fig. 2-6, the key behind node sync is the proper choice of the threshold, T. Based on extensive testing, we find that T is a function of the symbol SNR, E_s/N_0. A typical empirically determined threshold curve is defined as follows:

$$T = \begin{cases} -50 \cdot E_s/N_0 + 730, & \text{if } E_s/N_0 \leq 7.5 \\ 355, & \text{if } E_s/N_0 > 7.5 \end{cases} \tag{2-9}$$

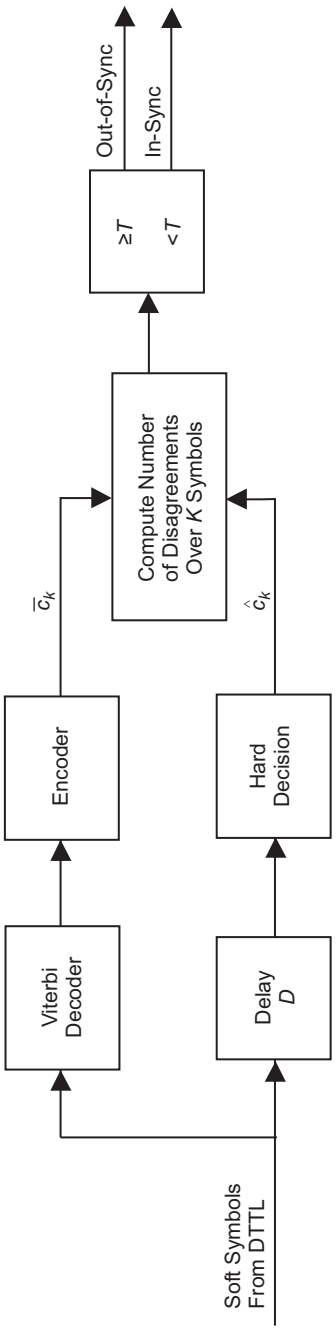


Fig. 2-6. Viterbi code node sync based on re-encoding.

Thus, once E_s/N_0 is estimated (as described below), it is converted to an error threshold (count) T via Eq. (2-9).

A flow diagram of the node sync algorithm that has been incorporated into Electra is presented in Fig. 2-7. There are three basic components of the algorithm: (1) the acquisition portion, wherein initial symbol-sequence phasing is determined, (2) the tracking portion, wherein lock status is continuously updated, and (3) threshold computation, which uses a combination of symbol SNR estimates and empirically derived estimates based on measurements of the decoder errors. These are described herein.

Viterbi sync acquisition starts once the DTTL is in lock. A large error-count threshold of $T_\Phi = 255$ is used to prevent the hardware (Temic) chip from prematurely halting its operation. The first steps in the acquisition process are computation of errors for the two node phases. This is done sequentially. The results of these error counts are denoted by the "Integrate $\Phi_0(S_0, N_\Phi)$" and "Integrate $\Phi_1(S_i, N_\Phi)$" blocks in Fig. 2-7. N_Φ is the number of bits used in the integration process (a multiple of 1000 bits), and S_i denotes error count for each code phase, Φ_i. If the difference in the error counts is sufficiently large, $|S_0 - S_1| \geq \alpha_0$ (α_0 is a pre-determined threshold between 0 and 255) *and* the smallest of the two error counts, $S_{\Phi\,\min}$, is less than a second pre-determined threshold α_1 (between 0 and 2048), then the node phase is set to the phase associated with the minimum error count and acquisition is completed. If not, then the decoder resets and the above process is repeated.

Once acquisition is declared, the node sync algorithm enters the tracking phase. The first steps in the tracking portion are to (1) initialize an un-lock counter U_L to zero and (2) initialize a running error-count average, X_A. During tracking, the number of bits used in computing the error counts is set to a multiple, $K = 1, 2, 4, 8, \cdots$, of N_Φ used for acquisition. Consequently, X_A is initialized to $K \cdot S_{\Phi\,\min}$. This is used in setting the tracking-error-count threshold T_S, as will be described in the next paragraph. Once this initialization occurs, then the error count is measured (S_S in Fig. 2-7) and compared against T_S. If $S_S < T_S$, then U_L is maintained at 0 and the error count average, X_A, is updated via $X_A = (X_A + S_S)/2$. Note that this represents a simple, exponential running average of the form

$$X_A(n) = \alpha \cdot X_A(n-1) + (1-\alpha) \cdot S_S(n) \tag{2-10}$$

where $\alpha = 1/2$ and n is a sequential time index. This average converges quite quickly (in just a few steps) and provides a reasonably smoothed estimate of the error count. If $S_S \geq T_S$ in Fig. 2-7, then U_L is incremented by 1 and compared against a threshold, L_T, between 0 and 15. As long as $U_L \leq L_T$, then the tracking loop proceeds. If not, then the decoder resets and acquisition is re-initiated.

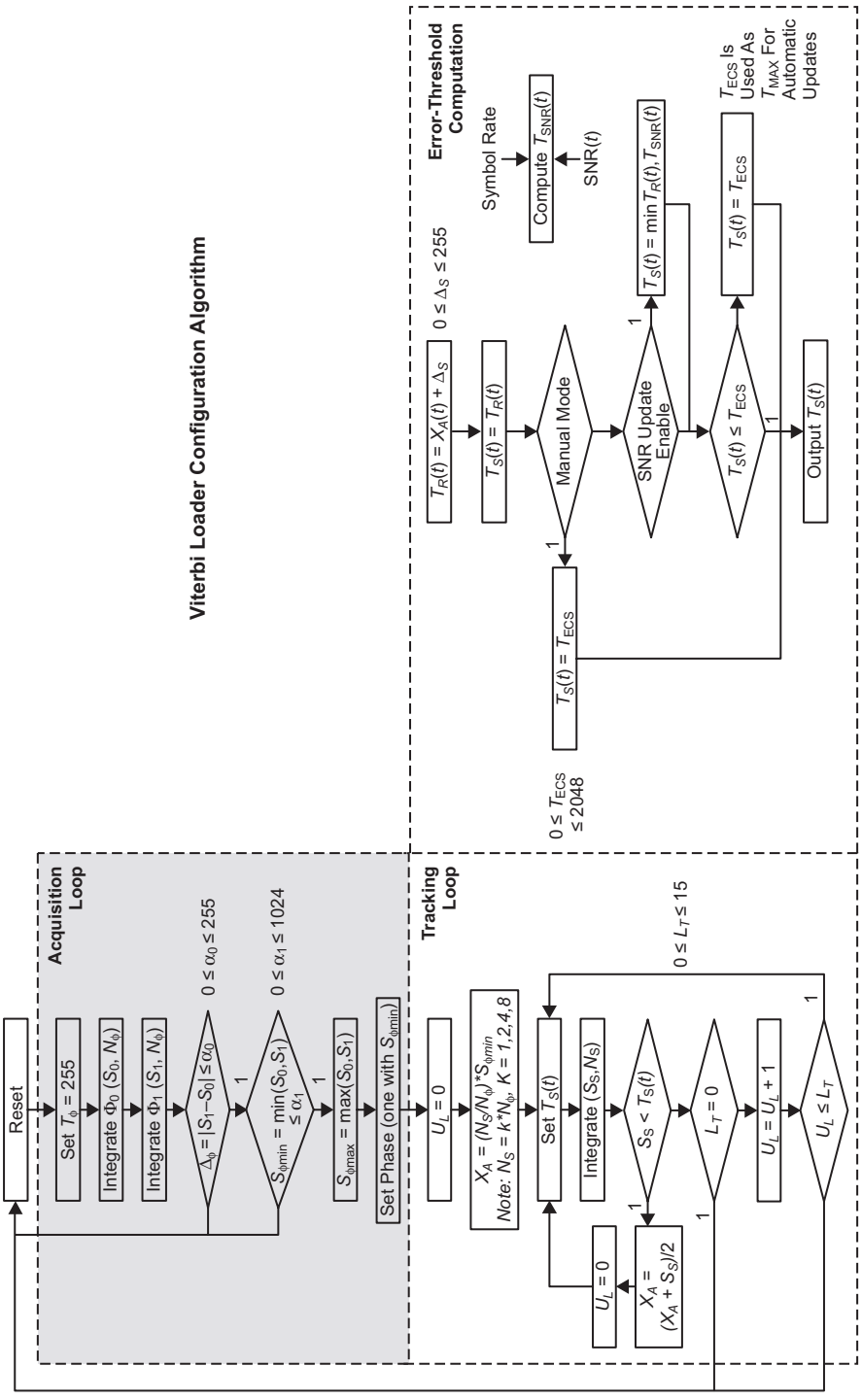


Fig. 2-7. Flow diagram of the Viterbi code node synchronizer.

The tracking-error-count threshold T_S is computed based on both the averaged error count X_A and the symbol SNR estimate. Initially, the threshold is set equal to $T_R \equiv X_A + \Delta_S$ (Δ_S is a pre-determined constant between 0 and 255). Then, assuming the symbol SNR estimator is enabled, a threshold $T_{\rm SNR}$ is computed from the above empirical relationship, Eq. (2-9). The tracking-error-count threshold T_S then is set to the minimum of T_R and $T_{\rm SNR}$ (if the symbol SNR estimator is not enabled, $T_S = T_R$). In addition, a manual override threshold, $0 \leq T_{ECS} \leq 2048$, is available if desired.

The various parameters defined above (e.g., T_{ECS}, α_0, α_1, N_Φ, etc.) are programmable and typically are determined from calibration measurements made on a given Electra unit. The symbol SNR, E_s/N_0, on the other hand, must be estimated from the received data. Techniques for estimating symbol SNR are described in Chapter 6. A different technique has been incorporated into the current Electra radio based more on FPGA constraints than on performance considerations. However, the Electra FPGA will be re-programmed to accommodate a better symbol SNR estimator based on the techniques described in Chapter 6.

A block diagram of the symbol SNR estimator currently used in Electra is presented in Fig. 2-8. The one-bit control signal, pll_mode, indicates whether suppressed carrier (pll_mode = 0) or residual carrier (pll_mode = 1) is being received. In the former case, the carrier-tracking loop is set to the Costas mode and the input data to the SNR indicator comprise time-averaged samples of the magnitude I- and Q-arm filter outputs (see Figs. 2-2 and 2-4). When pll_mode = 1, the carrier-tracking loop is set to the phase-locked loop (PLL) mode and the input data to the SNR indicator comprise time-averaged samples of the I- and residual-I-arm filter outputs. In practice, symbol SNR estimates are obtained via a table lookup procedure. Specifically, smoothed measurements of I_amp_k/Q_amp_k are collected at different symbol SNRs (see Fig. 2-8). A table of ratios then is created by curve fitting the measurements (two such tables are created, corresponding to suppressed- or residual-carrier mode). The resulting table comprises 64 ratios corresponding to symbol SNRs ranging from 0 dB to $(10 - 1/64)$ dB in 1/64-dB steps. Given the measurements, I_amp_k, Q_amp_k, the lookup procedure comprises the following steps: (1) Find $\min_j |\rho * Q\_amp_k * \text{Table}\ (j) - I\_amp_k|$, where "Table (j)" denotes the 64-element table of ratios and ρ is a constant that is nominally set to 1 but can be fine-tuned so that the single table can accommodate the different data rates, and (2) given the table index j_0 corresponding to the smallest magnitude residual in the first step, compute the symbol SNR estimate (dB) from $\hat{E}_s/N_0 = j_0/8 + j_0/32 + \text{bias}$. The bias is nominally zero but is also fine-tuned for different data rates.

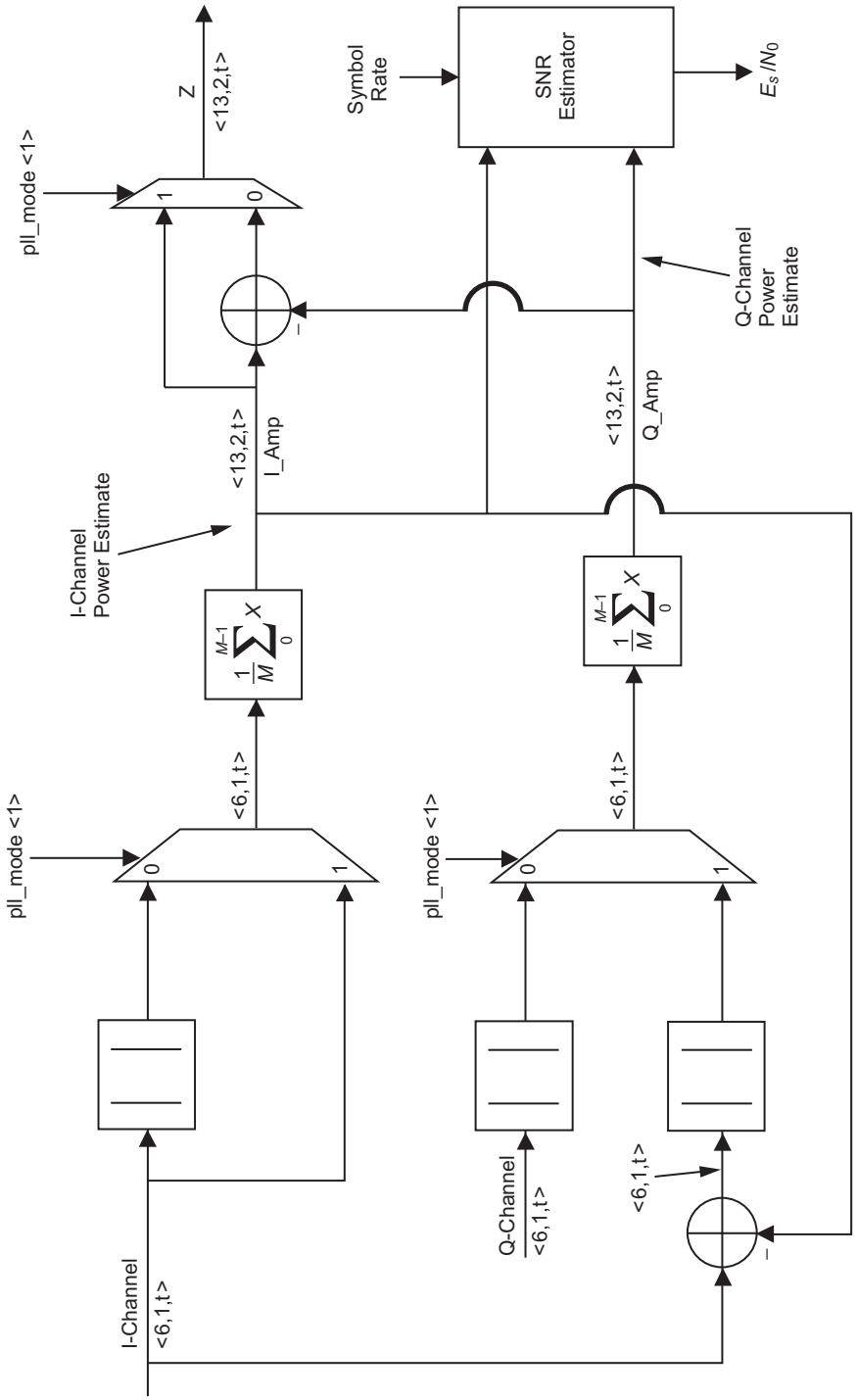


Fig. 2-8. Symbol SNR architecture.

As currently implemented in the residual-carrier mode, ρ is always equal to 1 for all data rates. The table entries "Table (j)" and bias offsets are obtained during system calibration and are uploaded to the Electra FPGA.

2.3 Electra Digital Modulator

The Electra modulator is fully programmable and is implemented on the FPGA. A block diagram of the modulator is depicted in Fig. 2-9. The coding (differential, convolutional encoding) and scrambling functions also are implemented on the FPGA. However, these typically are not programmed and thus are not described herein. Instead, we describe the programmable digital modulator section extending from the coders to the digital-to-analog converter (DAC) outputs. It is noted that in order to support carrier-only operations as required for the Proximity-1 protocol, a user-controlled variable called "enc_enable" is provided to force the final output of the encoder chain to a constant when needed. The output of the encoder is forced to 0 when setting "enc_enable" to 0. It is also noted that the modulator can accommodate dual-channel data inputs (a_n, b_n as depicted in Fig. 2-9). For BPSK modulation, only one input is required (and thus only one coder). However, the modulator also can accommodate QPSK modulation and in fact currently is being reprogrammed to include the suppressed-carrier, QPSK mode. The general purpose software that implements a protocol on an actual spacecraft is described in [9].

Manchester encoding typically is enabled for residual-carrier transmissions. It follows the coder chain as indicated in Fig. 2-9 and is considered part of the digital modulator. When enabled, the encoder converts NRZ waveforms to biphase-level-represented waveforms. In either case (Manchester or NRZ), the output is initially a binary (1,0) stream. After Manchester or NRZ encoding, the encoded binary data are converted to bipolar data. Typically, a logical zero is mapped into a phase of zero radians, and a logical one is mapped into a phase of π radians. This requires that for suppressed carrier, BPSK, the in-phase channel depicted in Fig. 2-9 (with $\cos\beta = 0$) is always zero. However, for hardware implementation, it is difficult to output an algebraic zero to the analog modulator (following the DACs). Thus, for the suppressed-carrier BPSK mode, the data are put on both I and Q channels ($a_n = b_n$), which essentially rotates the BPSK modulation by 45 deg. For the residual-carrier mode, the in-phase channel in Fig. 2-9 (prior to multiplication by $\cos\beta$) is always set equal to 1.

Prior to describing the different operational modes of the Electra digital modulator, we first point out that there are several programmable parameters that determine the power output from the modulator as well as gain and bias shifts on the two output channels from the digital modulator that are input to the DACs and analog modulator (Fig. 2-9). Specifically, for suppressed-carrier BPSK,

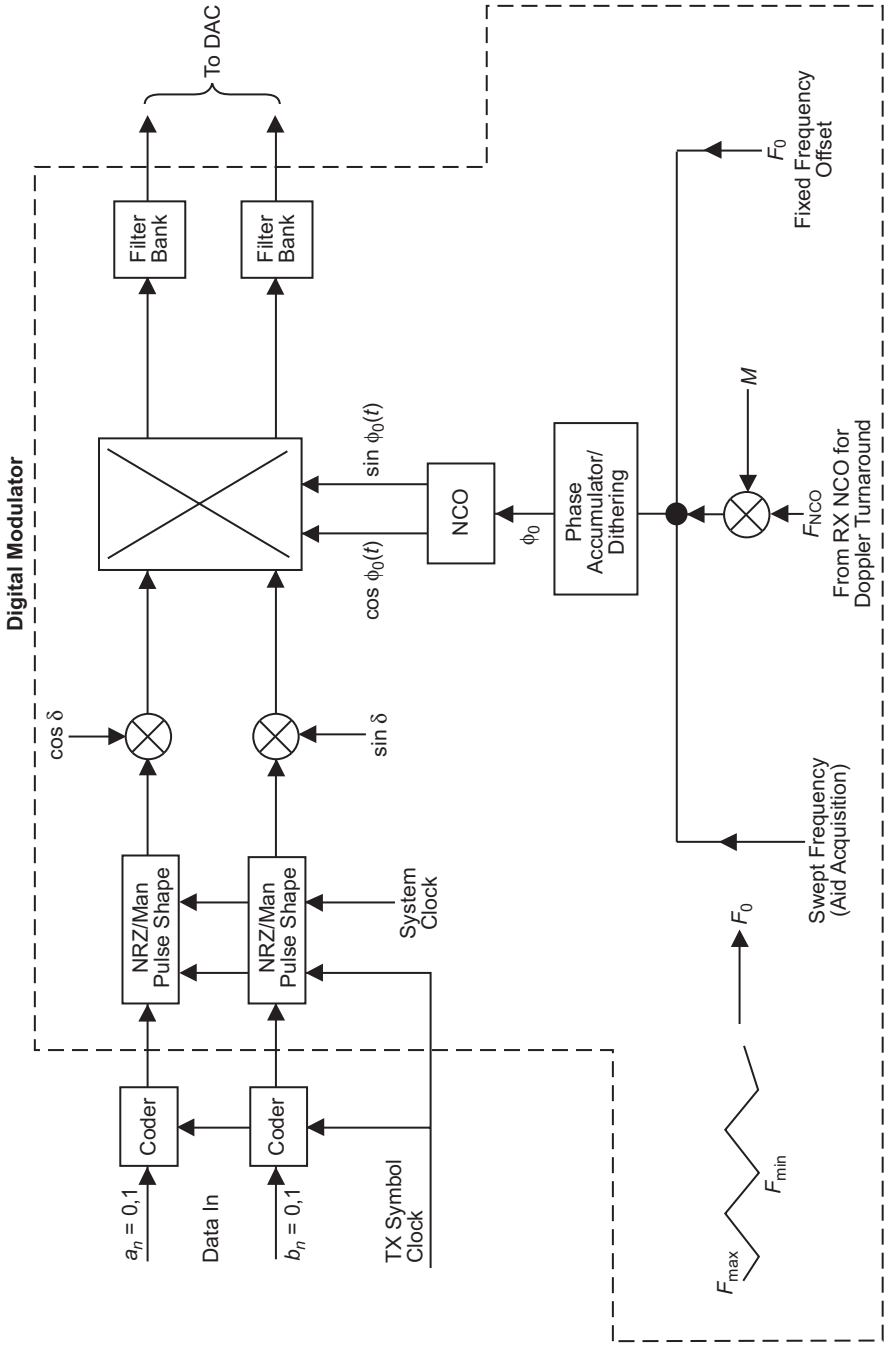


Fig. 2-9. Electra modulator block diagram.

the bipolar input levels to the complex multiplier depicted in Fig. 2-9 are identical and are computed from

$$\pm 0.5 \cdot \sqrt{T_{pow}} \tag{2-11}$$

where T_{pow} is a programmable input parameter controlling the modulator output power. For residual-carrier BPSK, the input levels to the complex multiplier are computed from

$$\text{In-phase} = 0.5 \cdot \sqrt{T_{pow}} \cdot \cos\beta \tag{2-12}$$

$$\text{Quadrature} = \pm 0.5 \cdot \sqrt{T_{pow}} \cdot \sin\beta \tag{2-13}$$

where β is the programmable modulation index parameter (nominally set to $\pi/3$ radians). Note that the in-phase component is unipolar and is the source of the residual carrier. The quadrature channel represents the data component of the residual-carrier waveform. In addition to β and T_{pow}, programmable gain and bias shifts are incorporated on the two output channels from the digital modulator that are input to the DACs. These are critical during system calibration in minimizing mixer images and carrier bleed-through at the output of the analog modulator.

Six operational modes have been implemented in the Electra digital modulator:

(1) Data only

(2) Doppler turnaround

(3) Offset frequency with no Doppler turnaround

(4) Offset frequency with Doppler turnaround

(5) Frequency sweeping with no Doppler turnaround

(6) Frequency sweeping with Doppler turnaround

These different operational modes are controlled via a user-controlled variable modulator_mode (3 bits wide). For mode 0, the modulator NCO is effectively turned off by forcing the NCO outputs to be constant. Mode 1 is the basic Doppler turnaround mode where the input frequency F_{NCO} to the demodulator CTL NCO phase accumulator is multiplied by a turnaround ratio (Fig. 2-9). The turnaround ratio is a programmable input parameter. For Mode 2, the modulator NCO generates in-phase and quadrature sinewave components based on a

fixed carrier offset that is a programmable input parameter. Doppler turnaround is disabled for Mode 2, whereas for Mode 3 it is enabled along with the fixed carrier offset. As indicated in Fig. 2-9, the input to the NCO phase accumulator for Mode 3 is the sum of the turnaround and fixed offset frequencies. Modulator modes 4 and 5 are similar to modes 2 and 3 except that, instead of a fixed-frequency waveform, the modulator NCO generates a swept-frequency waveform that can be used to aid frequency acquisition at the receiver. The transmit sweep parameters (sweep rate, frequency limits, etc.) are all programmable input parameters. These different modes allow a large range of operational scenarios for the Electra transceiver.

References

[1] C. D. Edwards, T. C. Jedrey, E. Schwartzbaum, A. S. Devereaux, R. DePaula, M. Dapore, and T. W. Fischer, "The Electra Proximity Link Payload for Mars Relay Telecommunications and Navigation, IAC-03-Q.3.A06," *International Astronautical Congress 2003*, September–October 2003.

[2] M. Agan, A. Gray, E. Grigorian, D. Hansen, E. Satorius, and C. Wang, "Micro Communications and Avionics Systems First Prototype (MCAS1): A Low Power, Low Mass In Situ Transceiver," *The Telecommunications and Mission Operations Progress Report 42-138, April–June 1999*, Jet Propulsion Laboratory, Pasadena, California, pp. 1–35, August 15, 1999. http://ipnpr.jpl.nasa.gov/progress_report/42-138/138I.pdf

[3] Consultative Committee for Space Data Systems, *CCSDS 211.0-B-1: Proximity-1 Space Link Protocol*, Blue Book, issue 1, October 2002.

[4] E. Hogenauer, "An Economical Class of Digital Filters for Decimation and Interpolation," *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 29, no. 2, pp. 155–162, 1981.

[5] A. Kwentus, Z. Jiang, and A. N. Willson, Jr., "Application of Filter Sharpening to Cascaded Integrator-Comb Decimation Filters," *IEEE Transactions on Signal Processing*, vol. 45, no. 2, pp. 457–467, 1997.

[6] T. Ely, "Two Way Electra Phase and Integrated Doppler," *AIAA GNC Conference*, 2005, to appear.

[7] W. C. Lindsey and M. K. Simon, *Telecommunication Systems Engineering*, Englewood Cliffs, New Jersey: Prentice-Hall, 1973; reprinted by Dover Press, New York, 1991.

[8] U. Mengali, R. Pellizzoni, and A. Spalvieri, "Soft-Decision-Based Node Synchronization for Viterbi Decoders," *IEEE Transactions on Communications*, vol. 43, pp. 2532–2539, September 1995.

[9] K. Peters, *Electra Baseband Processor Module (BPM) Payload Controller Software Release Description Document*, JPL D-22719 (internal document), Jet Propulsion Laboratory, Pasadena, California, September 29, 2004.

Chapter 3
Modulation Index Estimation

Marvin K. Simon and Jon Hamkins

Of the many possible modes available for communicating information over a coherent communication channel, one that is quite common is to allocate a portion of the total transmitted power P_t to a discrete carrier for purposes of carrier synchronization. In the case of binary phase-shift keying (BPSK) modulation, the simplest way to accomplish this is to employ a phase modulation index β other than 90 deg. When this is done, the fraction of power allocated to the discrete carrier becomes $P_c = P_t \cos^2 \beta$ with the remaining fractional power $P_d = P_t \sin^2 \beta$ available for data detection. When using this signaling mode, one must assure oneself that the power spectrum of the data modulation is such that it does not interfere with the extraction of the discrete carrier by an appropriate carrier-tracking loop such as a phase-locked loop (PLL). In other words, the discrete carrier should be inserted at a point where the power spectrum of the data modulation is minimum, preferably equal to zero. In the case of digital data, this rules out direct modulation of the carrier with a non-return-to-zero (NRZ) data stream whose spectrum is maximum at zero frequency, which at radio frequency (RF) would correspond to the carrier frequency. Instead one can first modulate the data onto a subcarrier whose frequency is selected significantly higher than the data rate so that the sidebands of the data modulation are sufficiently reduced by the time they reach the carrier frequency. Alternatively, one can use a data format such as biphase-L (Manchester coding), whose power spectrum is identically equal to zero at zero frequency, and directly modulate the carrier.

On other occasions it might be preferable to use a coherent communication mode where carrier synchronization is established directly from the data-bearing signal, e.g., using a Costas loop. In this case, none of the transmitted power is allocated to a discrete carrier, and the system is said to operate in a suppressed-

carrier mode, which in the case of BPSK corresponds to $\beta = 90$ deg. Although a Costas loop operates with a less efficient performance (e.g., larger mean-squared phase-tracking error) than a PLL, it offers the advantage of not requiring a discrete carrier to lock onto, and thus all of the transmitted power can be used for the purpose of data detection.

Given that either of the transmission modes discussed above is possible, in the case of autonomous receiver operation it is essential to have a means of estimating the modulation index or, equivalently, the ratio of transmitted carrier to data power. In this chapter, in Section 3.1 we first pursue the maximum-likelihood (ML) estimation approach to estimating modulation index along with appropriate approximation of the nonlinearities that result to allow for practical implementations at low and high signal-to-noise ratio (SNR) scenarios. In Section 3.2, we consider modulation index estimation for the case where carrier synchronization has not yet been established, i.e., the carrier phase is random. Here the ML estimation problem is too difficult to handle analytically and so we propose an ad hoc scheme instead. Finally, in Section 3.3, we describe how this scheme may be applied when the modulation type, symbol timing, and data rate are also unknown.

3.1 Coherent Estimation

3.1.1 BPSK

We begin by considering BPSK modulation where the received signal is given in complex baseband by Eqs. (1-3) and (1-6), or in passband by

$$\begin{aligned} r(t) &= \sqrt{2P_t}\sin\left(\omega_c t + \beta \sum_{n=-\infty}^{\infty} c_n p(t-nT)\right) + n(t) \\ &= \sqrt{2P_t\cos^2\beta}\sin\omega_c t + \sqrt{2P_t\sin^2\beta}\sum_{n=-\infty}^{\infty} c_n p(t-nT)\cos\omega_c t + n(t) \\ &= \sqrt{2P_c}\sin\omega_c t + \sqrt{2P_d}\sum_{n=-\infty}^{\infty} c_n p(t-nT)\cos\omega_c t + n(t) \end{aligned} \tag{3-1}$$

where, in addition to the aforementioned parameter definitions, $\{c_n\}$ is a binary sequence, which for our purposes may be treated as independent, identically distributed (iid) data taking on values ± 1 with equal probability; $p(t)$ is the pulse shape, also taking on values ± 1; ω_c is the carrier frequency in rad/s;

$1/T$ is the data (symbol) rate; and $n(t)$ is a bandpass additive white Gaussian noise (AWGN) source with two-sided power spectral density $N_0/2$ W/Hz. Based on the above AWGN model, then for an observation of K data intervals, the conditional probability of the received signal given the data and the modulation index is given by

$$\begin{aligned} p\big(r(t)|\{c_n\},\beta\big) &= \frac{1}{\sqrt{\pi N_0}} \exp\left(-\frac{1}{N_0}\int_0^{KT} \Big[r(t) - \sqrt{2P_c}\sin\omega_c t \right. \\ &\qquad \left. -\sqrt{2P_d}\sum_{n=-\infty}^{\infty} c_n p(t-nT)\cos\omega_c t\Big]^2 dt\right) \\ &= C\exp\left(\frac{2\sqrt{2P_c}}{N_0}\int_0^{KT} r(t)\sin\omega_c t\,dt\right) \\ &\qquad \times \exp\left(\frac{2\sqrt{2P_d}}{N_0}\int_0^{KT} r(t)\sum_{n=-\infty}^{\infty} c_n p(t-nT)\cos\omega_c t\,dt\right) \end{aligned} \tag{3-2}$$

where C is a constant that has no bearing on the modulation index estimation to be performed. With some additional manipulation, Eq. (3-2) can be put in the form

$$\begin{aligned} p\big(r(t)|\{c_n\},\beta\big) &= C\exp\left(\frac{2\sqrt{2P_t}\cos\beta}{N_0}\int_0^{KT} r(t)\sin\omega_c t\,dt\right) \\ &\qquad \times \prod_{k=1}^{K-1}\exp\left(\frac{2\sqrt{2P_t}\sin\beta}{N_0}c_k\int_{kT}^{(k+1)T} r(t)p(t-kT)\cos\omega_c t\,dt\right) \end{aligned} \tag{3-3}$$

Averaging over the iid data sequence gives what is referred to as the conditional-likelihood function (CLF), namely,

$$\begin{aligned} p\big(r(t)|\beta\big) &= C\exp\left(\frac{2\sqrt{2P_t}\cos\beta}{N_0}\int_0^{KT} r(t)\sin\omega_c t\,dt\right) \\ &\qquad \times \prod_{k=1}^{K-1}\cosh\left(\frac{2\sqrt{2P_t}\sin\beta}{N_0}\int_{kT}^{(k+1)T} r(t)p(t-kT)\cos\omega_c t\,dt\right) \end{aligned} \tag{3-4}$$

Next, taking the logarithm of Eq. (3-4), we obtain the log-likelihood function (LLF)

$$\Lambda \triangleq \ln p\big(r(t)|\beta\big) = \frac{2\sqrt{2P_t}\cos\beta}{N_0}\int_0^{KT} r(t)\sin\omega_c t\,dt + \sum_{k=0}^{K-1}\ln\cosh\left(\frac{2\sqrt{2P_t}\sin\beta}{N_0}\int_{kT}^{(k+1)T} r(t)p(t-kT)\cos\omega_c t\,dt\right) \quad (3\text{-}5)$$

where we have ignored the additive constant $\ln C$.

Finally, differentiating the LLF with respect to β and equating the result to zero, we get

$$\frac{d}{d\theta}\ln p\big(r(t)|\beta\big) = -\frac{2\sqrt{2P_t}\sin\beta}{N_0}\int_0^{KT} r(t)\sin\omega_c t\,dt + \sum_{k=0}^{K-1}\tanh\left(\frac{2\sqrt{2P_t}\sin\beta}{N_0}\int_{kT}^{(k+1)T} r(t)p(t-kT)\cos\omega_c t\,dt\right) \times \frac{2\sqrt{2P_t}\cos\beta}{N_0}\int_{kT}^{(k+1)T} r(t)p(t-kT)\cos\omega_c t\,dt = 0 \quad (3\text{-}6)$$

from which the ML estimate of β, namely, $\hat{\beta}$, is the solution to the transcendental equation

$$\int_0^{KT} r(t)\sin\omega_c t\,dt = \Big(\cot\hat{\beta}\Big)\sum_{k=0}^{K-1}\tanh\left(\frac{2\sqrt{2P_t}\sin\hat{\beta}}{N_0}\int_{kT}^{(k+1)T} r(t)p(t-kT)\cos\omega_c t\,dt\right) \times \frac{2\sqrt{2P_t}}{N_0}\int_{kT}^{(k+1)T} r(t)p(t-kT)\cos\omega_c t\,dt \quad (3\text{-}7)$$

In order to arrive at an estimation algorithm that is practical to implement, one must now make suitable approximations to the nonlinearity in Eq. (3-7) cor-

responding to low and high data detection SNR conditions. For large arguments, the hyperbolic tangent nonlinearity can be approximated as

$$\tanh x \cong \operatorname{sgn} x \tag{3-8}$$

Applying this approximation to Eq. (3-7), we arrive at the simple result

$$\cot \hat{\beta} = \frac{\int_0^{KT} r(t) \sin \omega_c t \, dt}{\sum_{k=0}^{K-1} \left| \int_{kT}^{(k+1)T} r(t) p(t - kT) \cos \omega_c t \, dt \right|} \tag{3-9}$$

which for rectangular pulses simplifies further to

$$\cot \hat{\beta} = \frac{\int_0^{KT} r(t) \sin \omega_c t \, dt}{\sum_{k=0}^{K-1} \left| \int_{kT}^{(k+1)T} r(t) \cos \omega_c t \, dt \right|} \tag{3-10}$$

The result in Eq. (3-10) is intuitively satisfying since, in the absence of noise, it becomes

$$\cot \hat{\beta} = \frac{\int_0^{KT} \sqrt{2P_c} \sin^2 \omega_c t \, dt}{\sum_{k=0}^{K-1} \left| \int_{kT}^{(k+1)T} c_k \sqrt{2P_d} \cos^2 \omega_c t \, dt \right|} = \frac{\sqrt{2P_c}(KT/2)}{\sqrt{2P_d}(KT/2)} = \sqrt{\frac{P_c}{P_d}} \tag{3-11}$$

For small arguments, the hyperbolic tangent nonlinearity can be approximated as

$$\tanh x \cong x \tag{3-12}$$

Applying this approximation to Eq. (3-7), we arrive at the simple result

$$\cos \hat{\beta} = \frac{N_0 \int_0^{KT} r(t) \sin \omega_c t \, dt}{2\sqrt{2P_t} \sum_{k=0}^{K-1} \left(\int_{kT}^{(k+1)T} r(t) p(t - kT) \cos \omega_c t \, dt \right)^2} \tag{3-13}$$

which for rectangular pulses simplifies further to

$$\cos\hat{\beta} = \frac{N_0 \int_0^{KT} r(t)\sin\omega_c t\,dt}{2\sqrt{2P_t}\sum_{k=0}^{K-1}\left(\int_{kT}^{(k+1)T} r(t)\cos\omega_c t\,dt\right)^2} \tag{3-14}$$

Unfortunately, there is no guarantee that the right-hand side of Eq. (3-13) will be less than or equal to unity and thus a solution to this equation may not always exist.

3.1.2 *M*-PSK

For M-phase shift keying (M-PSK) modulation ($M > 2$), the received signal can be represented in complex baseband using Eqs. (1-3) and (1-7), or in passband by

$$r(t) = \sqrt{2P_c}\sin\omega_c t + \sqrt{2P_d}\cos\left(\omega_c t + \sum_{n=-\infty}^{\infty}\theta_n p(t-nT)\right) + n(t) \tag{3-15}$$

where $\theta_n = [2q_n + (1+(-1)^{M/2})/2]\pi/M$ is the data modulation for the nth M-PSK symbol, with independent and uniformly distributed $q_n \in \{0, 1, \cdots, M-1\}$. The CLF analogous to Eq. (3-3) now becomes

$$p\big(r(t)|\{\theta_k\},\beta\big) = C\exp\left(\frac{2\sqrt{2P_c}}{N_0}\int_0^{KT} r(t)\sin\omega_c t\,dt\right)$$
$$\times\prod_{k=0}^{K-1}\exp\left(\frac{2\sqrt{2P_d}}{N_0}\int_{kT}^{(k+1)T} r(t)\cos\big(\omega_c t + \theta_k p(t-kT)\big)\,dt\right) \tag{3-16}$$

Once again averaging over the data symbols, then because of the symmetry of the constellation around the circle, i.e., for each phase value there is one that is π radians away from it, we obtain

$$p\big(r(t)|,\beta\big) = C\exp\left(\frac{2\sqrt{2P_t}\cos\beta}{N_0}\int_0^{KT} r(t)\sin\omega_c t\,dt\right)$$
$$\times\prod_{k=0}^{K-1}\frac{2}{M}\sum_{q=0}^{(M/2)-1}\cosh\left[\frac{2\sqrt{2P_t}\sin\beta}{N_0}\right.$$
$$\left.\times\int_{kT}^{(k+1)T} r(t)\cos\left(\omega_c t+\frac{(2q+1)\pi}{M}p(t-kT)\right)\,dt\right] \quad (3\text{-}17)$$

where we have artificially introduced the modulation index β to have the same meaning as in the BPSK case. Once again taking the logarithm of Eq. (3-17), we obtain the LLF

$$\Lambda \triangleq \ln p\big(r(t)|\beta\big) = \frac{2\sqrt{2P_t}\cos\beta}{N_0}\int_0^{KT} r(t)\sin\omega_c t\,dt + \sum_{k=0}^{K-1}\ln\frac{2}{M}$$
$$\times\sum_{q=0}^{(M/2)-1}\cosh\left(\frac{2\sqrt{2P_t}\sin\beta}{N_0}\int_{kT}^{(k+1)T} r(t)\cos\left(\omega_c t+\frac{(2q+1)\pi}{M}p(t-kT)\right)\,dt\right) \quad (3\text{-}18)$$

Finally, differentiating Eq. (3-18) with respect to β and equating the result to zero results in the transcendental equation

$$\int_0^{KT} r(t)\sin\omega_c t\,dt =$$
$$\left(\cot\hat{\beta}\right)\sum_{k=0}^{K-1}\frac{\sum_{q=0}^{(M/2)-1} x_k(q)\sinh\left(\frac{2\sqrt{2P_t}\sin\hat{\beta}}{N_0}x_k(q)\right)}{\sum_{q=0}^{(M/2)-1}\cosh\left(\frac{2\sqrt{2P_t}\sin\hat{\beta}}{N_0}x_k(q)\right)} \quad (3\text{-}19)$$

$$x_k(q) \triangleq \int_{kT}^{(k+1)T} r(t)\cos\left(\omega_c t+\frac{(2q+1)\pi}{M}p(t-kT)\right)\,dt$$

whose solution is the ML estimate of the modulation index. As for the BPSK case, to get an implementable estimator we must invoke suitable approximations to the nonlinearities in Eq. (3-19).

For large arguments, the hyperbolic sine and cosine nonlinearities can be approximated as

$$\begin{aligned} \sinh x &\cong \frac{1}{2}\exp\left(|x|\right)\operatorname{sgn} x \\ \cosh x &\cong \frac{1}{2}\exp\left(|x|\right) \end{aligned} \tag{3-20}$$

from which we obtain

$$\int_0^{KT} r(t)\sin\omega_c t\,dt = \left(\cot\hat{\beta}\right)\sum_{k=0}^{K-1}\frac{\sum\limits_{q=0}^{(M/2)-1}|x_k(q)|\exp\left(\dfrac{2\sqrt{2P_t}\sin\hat{\beta}}{N_0}|x_k(q)|\right)}{\sum\limits_{q=0}^{(M/2)-1}\exp\left(\dfrac{2\sqrt{2P_t}\sin\hat{\beta}}{N_0}|x_k(q)|\right)} \tag{3-21}$$

Noting further that for large SNR the summations in Eq. (3-21) are dominated by their largest term, we can make the further simplification

$$\sum_{q=0}^{(M/2)-1}|x_k(q)|\exp\left(\frac{2\sqrt{2P_t}\sin\hat{\beta}}{N_0}|x_k(q)|\right)\cong$$

$$|x_k(q)|_{\max}\exp\left(\frac{2\sqrt{2P_t}\sin\hat{\beta}}{N_0}|x_k(q)|_{\max}\right) \tag{3-22}$$

$$\sum_{q=0}^{(M/2)-1}\exp\left(\frac{2\sqrt{2P_t}\sin\hat{\beta}}{N_0}|x_k(q)|\right)\cong\exp\left(\frac{2\sqrt{2P_t}\sin\hat{\beta}}{N_0}|x_k(q)|_{\max}\right) \tag{3-23}$$

where

$$|x_k(q)|_{\max}\triangleq\max_q|x_k(q)| \tag{3-24}$$

Finally, applying Eq. (3-22) to Eq. (3-21) gives the desired simplified solution for the ML estimate of modulation index for M-PSK, namely,

$$\cot\hat{\beta} = \frac{\int_0^{KT} r(t)\sin\omega_c t\,dt}{\sum_{k=0}^{K-1}|x_k(q)|_{\max}} \tag{3-25}$$

$$= \frac{\int_0^{KT} r(t)\sin\omega_c t\,dt}{\sum_{k=0}^{K-1}\max_q\left|\int_{kT}^{(k+1)T} r(t)\cos\left(\omega_c t + \frac{(2q+1)\pi}{M}p(t-kT)\right)\,dt\right|} \tag{3-26}$$

which for rectangular pulses simplifies further to

$$\cot\hat{\beta} = \frac{\int_0^{KT} r\,(t)\sin\omega_c t\,dt}{\sum_{k=0}^{K-1}|x_k\,(q)|_{\max}} = \frac{\int_0^{KT} r\,(t)\sin\omega_c t\,dt}{\sum_{k=0}^{K-1}\max_q\left|\int_{kT}^{(k+1)T} r\,(t)\cos\left(\omega_c t + \frac{(2q+1)\pi}{M}\right)\,dt\right|} \tag{3-27}$$

For low SNR, we can apply the small argument approximations

$$\begin{aligned}\sinh &\cong x\\ \cosh x &\cong 1\end{aligned} \tag{3-28}$$

Note that these approximations are consistent with the approximation of the hyperbolic tangent nonlinearity given in Eq. (3-12). Thus, applying the approximations in Eq. (3-28) to Eq. (3-19) results in the ML estimate

$$\begin{aligned}&\cos\hat{\beta}\\ &= \frac{N_0\int_0^{KT} r\,(t)\sin\omega_c t\,dt}{2\sqrt{2P_t}\sum_{k=0}^{K-1}\frac{2}{M}\sum_{q=0}^{(M/2)-1}x_k^2\,(q)}\\ &= \frac{MN_0\int_0^{KT} r\,(t)\sin\omega_c t\,dt}{4\sqrt{2P_t}\sum_{k=0}^{K-1}\sum_{q=0}^{(M/2)-1}\left(\int_{kT}^{(k+1)T} r\,(t)\cos\left(\omega_c t + \frac{(2q+1)\pi}{M}p\,(t-kT)\right)\,dt\right)^2}\end{aligned} \tag{3-29}$$

which has the same difficulty as that in the discussion following Eq. (3-14).

3.2 Noncoherent Estimation

In the noncoherent case, the modulation index estimate must be formed in the absence of carrier synchronization. For simplicity, we again begin the investigation for BPSK modulation. The received signal is again modeled as in Eq. (3-1) with the addition of an unknown (assumed to be uniformly distributed) carrier phase to both the discrete and data-modulated carriers. Thus, analogous to Eq. (3-4), we now have the CLF

$$p\big(r(t)|\beta,\theta_c\big) = C\exp\left(\frac{2\sqrt{2P_t}\cos\beta}{N_0}\int_0^{KT} r\left(t\right)\sin\left(\omega_c t\ +\theta_c\right)dt\right)$$

$$\times\prod_{k=1}^{K-1}\cosh\left(\frac{2\sqrt{2P_t}\sin\beta}{N_0}\int_{kT}^{(k+1)T} r\left(t\right)p\left(t-kT\right)\cos\left(\omega_c t\ +\theta_c\right)\,dt\right) \quad (3\text{-}30)$$

The next step would be to average over the uniformly distributed carrier phase, which is an analytically intractable task. Even after approximating the nonlinearities as was done in the coherent case, performing this average is still analytically intractable. Thus, we abandon our search for the ML estimate and instead propose the following ad hoc approach.

Consider demodulating the received signal of Eq. (3-1), including now the unknown carrier phase θ_c, with the in-phase (I) and quadrature (Q) carriers (arbitrarily assumed to have zero phase relative to the unknown carrier phase of the received signal)

$$\begin{aligned} r_c\left(t\right) &= \sqrt{2}\cos\omega_c t \\ r_s\left(t\right) &= \sqrt{2}\sin\omega_c t \end{aligned} \quad (3\text{-}31)$$

Then, the outputs of these demodulations become

$$\begin{aligned} y_c\left(t\right) &= r\left(t\right)r_c\left(t\right) = \sqrt{P_c}\sin\theta_c + \sqrt{P_d}\sum_{n=-\infty}^{\infty} c_n p\left(t-nT\right)\cos\theta_c + n_c\left(t\right) \\ y_s\left(t\right) &= r\left(t\right)r_s\left(t\right) = \sqrt{P_c}\cos\theta_c - \sqrt{P_d}\sum_{n=-\infty}^{\infty} c_n p\left(t-nT\right)\sin\theta_c + n_s\left(t\right) \end{aligned} \quad (3\text{-}32)$$

where

$$\begin{aligned} n_c(t) &= n(t)\left(\sqrt{2}\cos\omega_c t\right) \\ n_s(t) &= n(t)\left(\sqrt{2}\sin\omega_c t\right) \end{aligned} \tag{3-33}$$

Integrating $y_c(t)$ and $y_s(t)$ over K symbol durations and summing the squares of these integrations gives

$$\left(\int_0^{KT} y_c(t)\,dt\right)^2 + \left(\int_0^{KT} y_s(t)\,dt\right)^2 = (KT)^2 P_t \cos^2\beta$$

$$+ (KT)^2 P_t \sin^2\beta \left(\frac{1}{K}\sum_{k=0}^{K-1}\frac{1}{T}\int_{kT}^{(k+1)T} c_k p(t-kT)\,dt\right)^2 + N_1(t) \tag{3-34}$$

where $N_1(t)$ is composed of $S\times N$ and $N\times N$ terms. For sufficiently large K, the data-dependent term becomes vanishingly small, in which case Eq. (3-34) simplifies to

$$\left(\int_0^{KT} y_c(t)\,dt\right)^2 + \left(\int_0^{KT} y_s(t)\,dt\right)^2 = (KT)^2 P_t\cos^2\beta + N_1(t) \tag{3-35}$$

Next, noting that the first terms in Eq. (3-32) are constant with time, form the difference signals

$$\begin{aligned} y_c(t) - y_c(t-T) &= \sqrt{P_t}\sin\beta \sum_{n=-\infty}^{\infty}(c_n - c_{n-1})\,p(t-nT)\cos\theta_c \\ &\quad + n_c(t) - n_c(t-T) \\ y_s(t) - y_s(t-T) &= -\sqrt{P_t}\sin\beta \sum_{n=-\infty}^{\infty}(c_n - c_{n-1})\,p(t-nT)\sin\theta_c \\ &\quad + n_s(t) - n_s(t-T) \end{aligned} \tag{3-36}$$

Now first squaring these signals and then integrating them over K symbol durations, the sum of these integrations becomes

$$\int_0^{KT} \left(y_c(t) - y_c(t-T)\right)^2 dt + \int_0^{KT} \left(y_s(t) - y_s(t-T)\right)^2 dt =$$

$$P_t \sin^2\beta \sum_{k=0}^{K-1} \int_{kT}^{(k+1)T} \left(2 - 2c_k c_{k-1}\right) p^2\left(t - kT\right) dt + N_2\left(t\right) \quad (3\text{-}37)$$

where again $N_2(t)$ is composed of $S \times N$ and $N \times N$ terms. Once again, for sufficiently large K, the data-dependent term becomes vanishingly small and, assuming for convenience rectangular pulses, Eq. (3-37) simplifies to

$$\int_0^{KT} \left(y_c(t) - y_c(t-T)\right)^2 dt + \int_0^{KT} \left(y_s(t) - y_s(t-T)\right)^2 dt =$$

$$2KTP_t \sin^2\beta + N_2(t) \quad (3\text{-}38)$$

Finally then, from observation of Eqs. (3-35) and (3-38), it is reasonable to propose the ad hoc noncoherent estimator of modulation index

$$\cot\hat{\beta} = \sqrt{\frac{2\left[\left(\int_0^{KT} y_c(t)dt\right)^2 + \left(\int_0^{KT} y_s(t)dt\right)^2\right]}{KT\left[\int_0^{KT}\left(y_c(t) - y_c(t-T)\right)^2 dt + \int_0^{KT}\left(y_s(t) - y_s(t-T)\right)^2 dt\right]}} \quad (3\text{-}39)$$

Clearly, in the absence of noise this estimator produces the true value of the modulation index. Also, it has an advantage over Eqs. (3-14) and (3-29) in that the SNR need not be known to compute it. The architecture given by Eq. (3-39) is shown in Fig. 3-1.

3.3 Estimation in the Absence of Knowledge of the Modulation, Data Rate, Symbol Timing, and SNR

The modulation index estimators in Section 3.1 do not require an SNR estimate, and the ones in Section 3.2 require neither an SNR estimate nor a carrier phase estimate. However, they both require explicit knowledge of the phase-shift keying (PSK) modulation size, data rate, and symbol timing, as seen by the use (either explicitly or implicitly) of the parameters M and T and precise integration limits in Eqs. (3-10), (3-14), (3-27), (3-29), and (3-39).

In this section, we extend the ad hoc modulation index estimator of Eq. (3-39) for BPSK signals to a general M-PSK modulation where M is un-

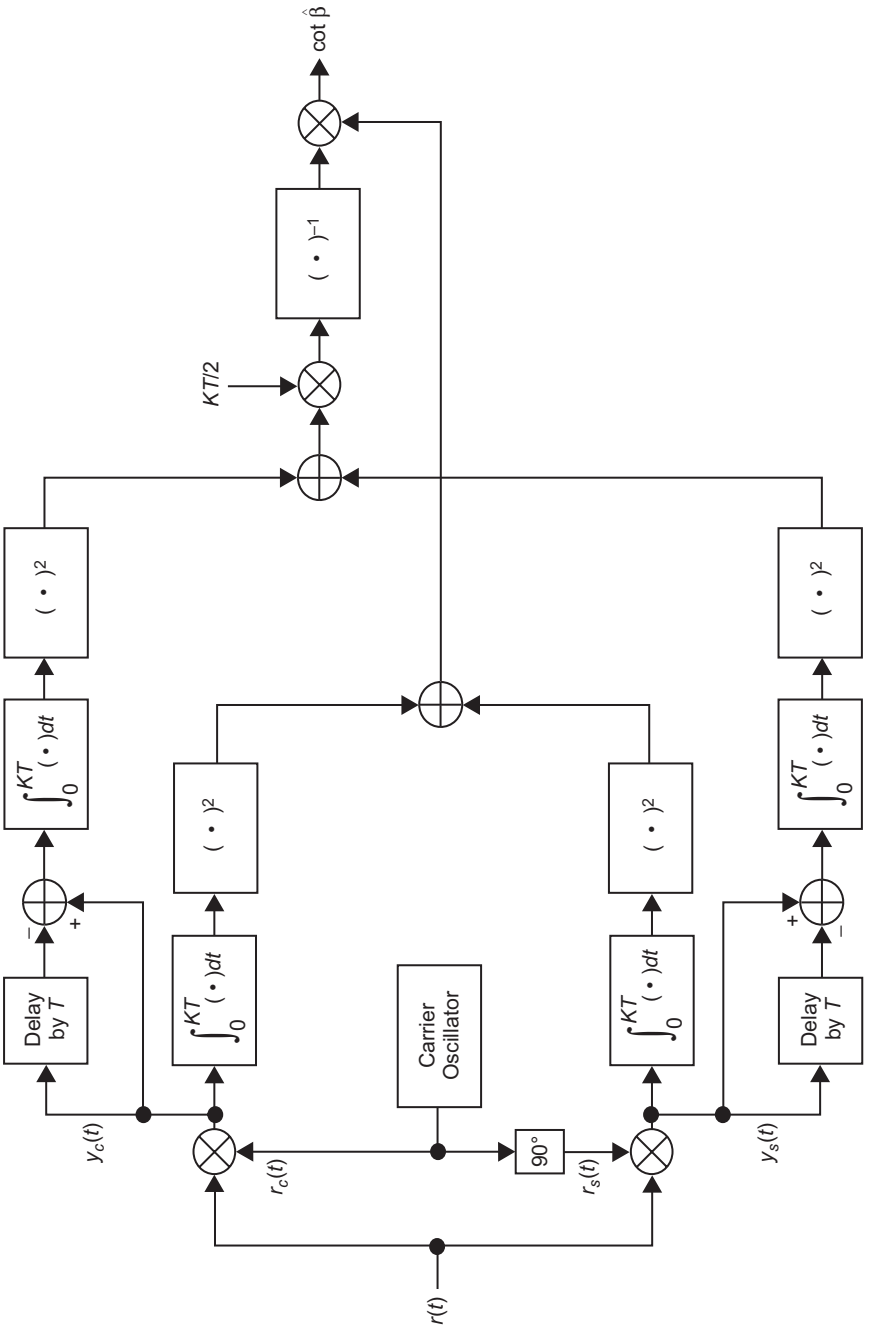


Fig. 3-1. An open-loop modulation index estimator.

known, and where the symbol rate $(1/T)$ and fractional symbol timing (ε) are also unknown. We assume that T takes on values in a finite set $\mathcal{T}$, and we define

$$T^* \triangleq \max_T \{T \in \mathcal{T}\} \tag{3-40}$$

The received signal can be represented as

$$\begin{aligned} r(t) = & \sqrt{2P_c}\sin(\omega_c t + \theta_c) \\ & + \sqrt{2P_d}\cos\left(\omega_c t + \theta_c + \sum_{n=-\infty}^{\infty} \theta_n p(t - nT - \varepsilon T)\right) + n(t) \end{aligned} \tag{3-41}$$

which is the same as Eq. (3-15) except that we have introduced unknown parameters θ_c and ε, and we now allow for the possibility of BPSK as well, so that $\theta_n = [2q_n + (1 + (-1)^{M/2})/2]\pi/M$ is the data modulation for the nth M-PSK symbol, with independent and uniformly distributed $q_n \in \{0, 1, \cdots, M-1\}$.

After mixing with in-phase and quadrature signals [see Eq. (3-31)], we have

$$\begin{aligned} y_c(t) = r(t)\, r_c(t) = & \sqrt{P_c}\sin\theta_c + \sqrt{P_d}\cos\left[\theta_c + \sum_{n=-\infty}^{\infty} \theta_n p(t - nT - \varepsilon T)\right] \\ & + n_c(t) \end{aligned} \tag{3-42}$$

$$\begin{aligned} y_s(t) = r(t)\, r_s(t) = & \sqrt{P_c}\cos\theta_c - \sqrt{P_d}\sin\left[\theta_c + \sum_{n=-\infty}^{\infty} \theta_n p(t - nT - \varepsilon T)\right] \\ & + n_s(t) \end{aligned}$$

where $n_c(t)$ and $n_s(t)$ are described by Eq. (3-33), as before.

Following the same strategy for ad hoc estimation as in Section 3.2, we integrate over a long duration. In this case, the integration limits are not necessarily aligned to the symbols, and T^* is used in place of the (unknown) T, to obtain

$$\begin{aligned} \frac{1}{KT^*}\int_0^{KT^*} y_c(t)dt = & \sqrt{P_c}\sin\theta_c \\ + \frac{1}{KT^*}\int_0^{KT^*} & \left[\sqrt{P_d}\cos\left(\theta_c + \sum_{n=-\infty}^{\infty} \theta_n p(t - nT - \varepsilon T)\right) + n_c(t)\right] dt \end{aligned} \tag{3-43}$$

and

$$\frac{1}{KT^*}\int_0^{KT^*} y_s(t)dt = \sqrt{P_c}\cos\theta_c$$
$$+\frac{1}{KT^*}\int_0^{KT^*}\left[\sqrt{P_d}\sin\left(\theta_c+\sum_{n=-\infty}^{\infty}\theta_n p(t-nT-\varepsilon T)\right)+n_s(t)\right]dt \qquad (3\text{-}44)$$

Using the facts that (1) M-PSK, for all M even, has the property that if θ_n is an allowable modulation angle, $\theta_n+\pi$ is as well, (2) each point in the signal constellation is equally likely, (3) both NRZ and Manchester satisfy $p(t)\in\{-1,1\}$ for all t, and (4) $\cos(\alpha)=-\cos(\alpha+\pi)$ and $\sin(\alpha)=-\sin(\alpha+\pi)$, it follows that the integrals in Eqs. (3-43) and (3-44) each approach zero as $K\to\infty$. Thus, for sufficiently large K, we may write

$$\left(\frac{1}{KT^*}\int_0^{KT^*} y_c(t)dt\right)^2 \cong P_c\sin^2\theta_c \qquad (3\text{-}45)$$

$$\left(\frac{1}{KT^*}\int_0^{KT^*} y_s(t)dt\right)^2 \cong P_c\cos^2\theta_c \qquad (3\text{-}46)$$

or

$$\left(\frac{1}{KT^*}\int_0^{KT^*} y_c(t)dt\right)^2+\left(\frac{1}{KT^*}\int_0^{KT^*} y_s(t)dt\right)^2 \cong P_c \qquad (3\text{-}47)$$

It remains to obtain an estimate of P_d. We may form the difference

$$\begin{aligned}
y_c(t)-y_c(t-T^*) =& \sqrt{P_d}\left[\cos\left(\theta_c+\sum_{n=-\infty}^{\infty}\theta_n p(t-nT-\varepsilon T)\right)\right.\\
&\left.-\cos\left(\theta_c+\sum_{n=-\infty}^{\infty}\theta_n p(t-T^*-nT-\varepsilon T)\right)\right]\\
&+n_c(t)-n_c(t-T^*)\\
=&\sqrt{P_d}\left\{-2\sin\left[\frac{1}{2}\left(2\theta_c+\sum_{n=-\infty}^{\infty}(\theta_n+\theta_{n+l})p(t-nT-\varepsilon T)\right)\right]\right.\\
&\left.\times\sin\left[\frac{1}{2}\sum_{n=-\infty}^{\infty}(\theta_n-\theta_{n+l})p(t-nT-\varepsilon T)\right]\right\}\\
&+n_c(t)-n_c(t-T^*)
\end{aligned}$$

where $l \triangleq T^*/T$. Although l is unknown, since T is unknown, we will see that this parameter will drop out of the final metric. Forming a similar expression for the difference of $y_s(t)$ terms, squaring, and ignoring noise terms, we obtain

$$(y_c(t) - y_c(t - T^*))^2 \cong 4P_d \left\{ \sin^2 \left[\frac{1}{2} \left(2\theta_c + \sum_{n=-\infty}^{\infty} (\theta_n + \theta_{n+l}) p(t - nT - \varepsilon T) \right) \right] \right.$$

$$\left. \times \sin^2 \left[\frac{1}{2} \left(\sum_{n=-\infty}^{\infty} (\theta_n - \theta_{n+l}) p(t - nT - \varepsilon T) \right) \right] \right\}$$

$$(y_s(t) - y_s(t - T^*))^2 \cong 4P_d \left\{ \cos^2 \left[\frac{1}{2} \left(2\theta_c + \sum_{n=-\infty}^{\infty} (\theta_n + \theta_{n+l}) p(t - nT - \varepsilon T) \right) \right] \right.$$

$$\left. \times \sin^2 \left[\frac{1}{2} \left(\sum_{n=-\infty}^{\infty} (\theta_n - \theta_{n+l}) p(t - nT - \varepsilon T) \right) \right] \right\}$$

Thus,

$$\frac{1}{2KT^*} \int_0^{KT^*} \left[\big(y_c(t) - y_c(t - T^*)\big)^2 + \big(y_s(t) - y_s(t - T^*)\big)^2 \right] dt \cong$$

$$\frac{2P_d}{KT^*} \int_0^{KT^*} \sin^2 \left[\frac{1}{2} \big((\theta_n - \theta_{n+l}) p(t - nT - \varepsilon T) \big) \right] dt \cong P_d \quad (3\text{-}48)$$

where we have used $\sin^2 x = (1/2)(1 - 2\cos x)$ and the fact that the integration of the cosine term approaches zero for sufficiently large K.

Thus, an ad hoc estimator of the modulation index $\beta = \cot^{-1} \sqrt{P_c/P_d}$ is given by

$$\hat{\beta} = \cot^{-1} \left[\sqrt{ \frac{2 \left[\left(\int_0^{KT^*} y_c(t) dt \right)^2 + \left(\int_0^{KT^*} y_s(t) dt \right)^2 \right]}{KT^* \int_0^{KT^*} \left[\big(y_c(t) - y_c(t - T^*)\big)^2 + \big(y_s(t) - y_s(t - T^*)\big)^2 \right] dt} } \right] \quad (3\text{-}49)$$

which is identical to Eq. (3-39) when T is replaced by T^*. Thus, the same architecture shown in Fig. 3-1 may be used when modulation type, data rate, symbol timing, and SNR are unknown, by replacing T with T^*.

3.4 Noncoherent Estimation in the Absence of Carrier Frequency Knowledge

Consider now demodulating the received signal of Eq. (3-1) with I and Q references as in Eq. (3-31) with ω_c replaced by $\omega_c - \Delta\omega$, where $\Delta\omega$ denotes the uncertainty in the knowledge of the true carrier frequency ω_c. Then, the outputs of these demodulations are given by Eq. (3-32) with θ_c replaced by $\Delta\omega t + \theta_c$. Squaring $y_c(t)$ and $y_s(t)$ and summing these squares gives

$$y_c^2(t) + y_s^2(t) =$$

$$P_c + P_d \left(\sum_{n=-\infty}^{\infty} p^2(t - nT) + \sum_{n=-\infty}^{\infty} \sum_{\substack{m=-\infty \\ n \neq m}}^{\infty} a_n a_m p(t - nT)\, p(t - mT) \right) + N_1(t) \tag{3-50}$$

where as before $N_1(t)$ is composed of $S \times N$ and $N \times N$ terms. Integrating the sum in Eq. (3-50) over K symbol durations and recognizing that for sufficiently large K the time average over the data-dependent term can be replaced by the statistical average, which for random data equates to zero, we obtain the simplified result

$$\int_0^{KT} \left[y_c^2(t) + y_s^2(t) \right] dt = KT(P_c + P_d) + \overbrace{\int_0^{KT} N_1(t)\, dt}^{N_A} = KTP_t + N_A \tag{3-51}$$

Next, form the complex signal

$$\tilde{y}(t) = y_s(t) + j y_c(t)$$

$$= \left[\sqrt{P_c} + j\sqrt{P_d} \sum_{n=-\infty}^{\infty} a_n p(t - nT) \right] e^{j(\Delta\omega t + \theta_c)} + n_s(t) + j n_c(t) \tag{3-52}$$

and multiply it by its complex conjugate delayed by one symbol $\tilde{y}^*(t-T)$, which gives

$$\tilde{y}(t)\,\tilde{y}^*(t-T) = \left[P_c + P_d \sum_{n=-\infty}^{\infty} a_n a_{n-1} p(t-nT)\right] e^{j\Delta\omega T} + j\sqrt{P_c P_d} \sum_{n=-\infty}^{\infty} (a_n - a_{n-1})\, p(t-nT) e^{j\Delta\omega T} + N_2(t) \quad (3\text{-}53)$$

where again $N_2(t)$ is composed of $S \times N$ and $N \times N$ terms. Now integrate this complex product over K symbol intervals, once again ignoring the averages over the data-dependent terms, valid for large K. Thus,

$$\int_0^{KT} \tilde{y}(t)\,\tilde{y}^*(t-T)\,dt = KTP_c e^{j\Delta\omega T} + \overbrace{\int_0^{KT} N_2(t)\,dt}^{N_B(t)} \quad (3\text{-}54)$$

Finally, from observation of Eqs. (3-51) and (3-54), it is reasonable to propose the ad hoc noncoherent estimator of modulation angle

$$\cos\beta = \sqrt{\frac{\left|\int_0^{KT} \tilde{y}(t)\,\tilde{y}^*(t-T)\,dt\right|}{\int_0^{KT} |\tilde{y}(t)|^2\,dt}} \quad (3\text{-}55)$$

Chapter 4
Frequency Correction

Dariush Divsalar

Over the years, much effort has been spent in the search for optimum synchronization schemes that are robust and simple to implement [1,2]. These schemes were derived based on maximum-likelihood (ML) estimation theory. In many cases, the derived open- or closed-loop synchronizers are nonlinear. Linear approximation provides a useful tool for the prediction of synchronizer performance.

In this semi-tutorial chapter, we elaborate on these schemes for frequency acquisition and tracking. Various low-complexity frequency estimator schemes are presented in this chapter. The theory of ML estimation provides the optimum schemes for frequency estimation. However, the derived ML-based scheme might be too complex for implementation. One approach is to use theory to derive the best scheme and then try to reduce the complexity such that the loss in performance remains small. Organization of this chapter is as follows: In Section 4.1, we show the derivation of open- and closed-loop frequency estimators when a pilot (residual) carrier is available. In Section 4.2, frequency estimators are derived for known data-modulated signals (data-aided estimation). In Section 4.3, non-data-aided frequency estimators are discussed. This refers to the frequency estimators when the data are unknown at the receiver.

4.1 Frequency Correction for Residual Carrier

Consider a residual-carrier system where a carrier (pilot) is available for tracking. We consider both additive white Gaussian noise (AWGN) and Rayleigh fading channels in this section.

4.1.1 Channel Model

Let $\tilde{r}_c[k]$ be the kth received complex sample of the output of a low-pass filtered pilot. The observation vector $\tilde{\mathbf{r}}_\mathbf{c}$ with components $\tilde{r}_c[k]$; $k = 0, 1, \cdots, N-1$ can be modeled as

$$\tilde{r}_c[k] = Ae^{j(2\pi\Delta f k T_s + \theta_c)} + \tilde{n}[k] \tag{4-1}$$

where the $\tilde{r}_c[k]$ samples are taken every T_s seconds (sampling rate of $1/T_s$). In the above equation, $\tilde{n}[k]$, $k = 0, 1, \cdots, N-1$, are independent, identically distributed (iid) zero-mean, complex Gaussian random variables with variance σ^2 per dimension. The frequency offset to be estimated is denoted by Δf, and θ_c is an unknown initial carrier phase shift that is assumed to be uniformly distributed in the interval $[0, 2\pi)$ but constant over the N samples. For an AWGN channel, $A = \sqrt{2P_c}$ is constant and represents the amplitude of the pilot samples. For a Rayleigh fading channel, we assume A is a complex Gaussian random variable, where $|A|$ is Rayleigh distributed and $\arg A \triangleq \tan^{-1}\left(\mathrm{Im}(A)/\mathrm{Re}(A)\right)$ is uniformly distributed in the interval $[0, 2\pi)$, where $\mathrm{Im}(\cdot)$ denotes the imaginary operator and $\mathrm{Re}(\cdot)$ denotes the real operator.

4.1.2 Optimum Frequency Estimation over an AWGN Channel

We desire an estimate of the frequency offset Δf based on the received observations given by Eq. (4-1). The ML estimation approach is to obtain the conditional probability density function (pdf) of the observations, given the frequency offset. To do so, first we obtain the following conditional pdf:

$$P(\tilde{\mathbf{r}}_\mathbf{c}|\Delta f, \theta_c) = C_0 e^{-(1/2\sigma^2)Z} \tag{4-2}$$

where C_0 is a constant, and

$$Z = \sum_{k=0}^{N-1} \left| \tilde{r}_c[k] - Ae^{j(2\pi\Delta f k T_s + \theta_c)} \right|^2 \tag{4-3}$$

Define

$$Y = \sum_{k=0}^{N-1} \tilde{r}_c[k] e^{-j(2\pi\Delta f k T_s)} \tag{4-4}$$

Then Z can be rewritten as

$$Z = \sum_{k=0}^{N-1} \left|\tilde{r}_c[k]\right|^2 - 2A\mathrm{Re}(Ye^{-j\theta_c}) + \sum_{k=0}^{N-1} A^2 \tag{4-5}$$

The first and the last terms in Eq. (4-5) do not depend on Δf and θ_c. Denoting the sum of these two terms by C_1, then Z can be written as

$$Z = C_1 - 2A|Y|\cos(\theta_c - \arg Y) \tag{4-6}$$

Using Eq. (4-6), the conditional pdf of Eq. (4-2) can be written as

$$P(\tilde{\mathbf{r}}_\mathbf{c}|\Delta f, \theta_c) = C_2 \exp\left[\frac{A}{\sigma^2}|Y|\cos(\theta_c - \arg Y)\right] \tag{4-7}$$

where $C_2 = Ce^{-(C_1/2\sigma^2)}$. Averaging Eq. (4-7) over θ_c produces

$$P(\tilde{\mathbf{r}}_\mathbf{c}|\Delta f) = C_2 I_0\left(\frac{A|Y|}{\sigma^2}\right) \tag{4-8}$$

where $I_0(\cdot)$ is the modified Bessel function of zero order and can be represented as

$$I_0(x) = \frac{1}{2\pi}\int_0^{2\pi} e^{x\cos(\psi)} d\psi \tag{4-9}$$

Since $I_0(x)$ is an even convex cup $\cup$ function of x, maximizing the right-hand side of Eq. (4-8) is equivalent to maximizing $|Y|$. Thus, the ML metric for estimating the frequency offset can be obtained by maximizing the following metric:

$$\lambda(\Delta f) = |Y| = \left|\sum_{k=0}^{N-1} \tilde{r}_c[k]e^{-j(2\pi\Delta f k T_s)}\right| \tag{4-10}$$

4.1.3 Optimum Frequency Estimation over a Rayleigh Fading Channel

We desire an estimate of the frequency offset Δf over a Rayleigh fading channel. The ML approach is to obtain the conditional pdf of the observations,

given the frequency offset. To do so, first we start with the following conditional pdf:

$$P(\tilde{\mathbf{r}}_\mathbf{c}|A, \Delta f, \theta_c) = C_0 e^{-(1/2\sigma^2)Z} \tag{4-11}$$

where C_0 is a constant, and Z and Y are defined as in Eqs. (4-3) and (4-4). Since A is now a complex random variable, then Z can be rewritten as

$$Z = \sum_{k=0}^{N-1} \left|\tilde{r}_c[k]\right|^2 - 2\mathrm{Re}(YAe^{-j\theta_c}) + \sum_{k=0}^{N-1} |A|^2 \tag{4-12}$$

The first terms in Eq. (4-12) do not depend on A. Averaging the conditional pdf in Eq. (4-11) over A, assuming the magnitude of A is Rayleigh distributed and its phase is uniformly distributed, we obtain

$$P(\tilde{\mathbf{r}}_\mathbf{c}|\Delta f, \theta_c) = C_3 \exp\left(\frac{C_4}{2\sigma^2}|Y|^2\right) \tag{4-13}$$

where C_3 and C_4 are constants, and Eq. (4-13) is independent of θ_c. Thus, maximizing the right-hand side of Eq. (4-13) is equivalent to maximizing $|Y|^2$ or equivalently $|Y|$. Thus, the ML metric for estimating the frequency offset can be obtained by maximizing the following metric:

$$\lambda(\Delta f) = |Y| = \left|\sum_{k=0}^{N-1} \tilde{r}_c[k] e^{-j(2\pi \Delta f k T_s)}\right| \tag{4-14}$$

which is identical to that obtained for the AWGN channel case.

4.1.4 Open-Loop Frequency Estimation

For an open-loop estimation, we have

$$\widehat{\Delta f} = \underset{\Delta f}{\mathrm{argmax}}\, \lambda(\Delta f) \tag{4-15}$$

However, this operation is equivalent to obtaining the fast Fourier transform (FFT) of the received sequence, taking its magnitude, and then finding the maximum value, as shown in Fig. 4-1.

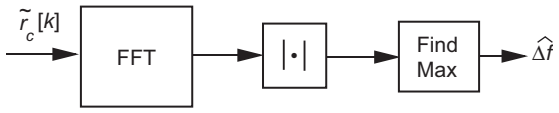


Fig. 4-1. Open-loop frequency estimation, residual carrier.

4.1.5 Closed-Loop Frequency Estimation

The error signal for a closed-loop estimator can be obtained as

$$e = \frac{\partial}{\partial \Delta f}\lambda(\Delta f) \tag{4-16}$$

We can approximate the derivative of $\lambda(\Delta f)$ for small ε as

$$\frac{\partial}{\partial \Delta f}\lambda(\Delta f) = \frac{\lambda(\Delta f + \varepsilon) - \lambda(\Delta f - \varepsilon)}{2\varepsilon} \tag{4-17}$$

Then, we can write the error signal as (in the following, any positive constant multiplier in the error signal representation will be ignored)

$$e = |Y(\Delta f + \varepsilon)| - |Y(\Delta f - \varepsilon)| \tag{4-18}$$

where

$$Y(\Delta f + \varepsilon) = \sum_{k=0}^{N-1} \tilde{r}_c[k] e^{-j(2\pi \Delta f k T_s)} e^{-j(2\pi \varepsilon k T_s)} \tag{4-19}$$

The error-signal detector for a closed-loop frequency correction can be implemented based on the above equations. The block diagram is shown in Fig. 4-2, where in the figure $\alpha = e^{-j2\pi\varepsilon T_s}$.

Now rather than using the approximate derivative of $\lambda(\Delta f)$, we can take the actual derivative of $\lambda^2(\Delta f) = |Y|^2$, which gives the error signal

$$e = \mathrm{Im}(Y^* U) \tag{4-20}$$

where

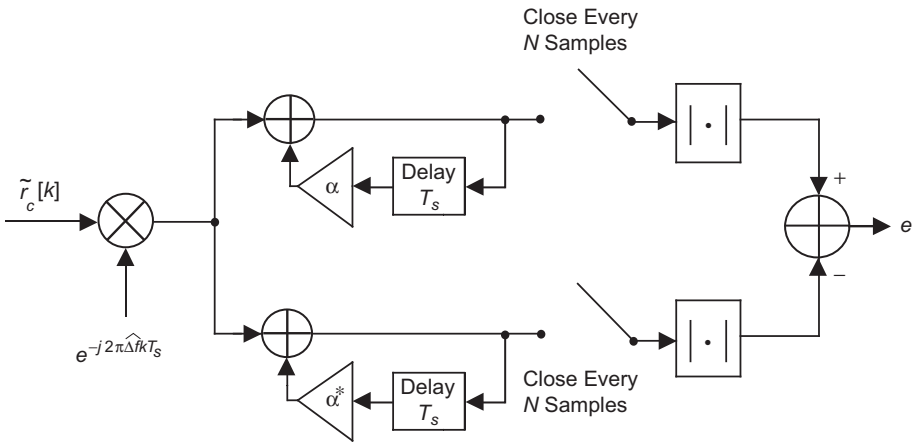


Fig. 4-2. Approximate error signal detector, residual carrier.

$$U = \sum_{k=0}^{N-1} \tilde{r}_c[k] k e^{-j(2\pi \Delta f k T_s)} \tag{4-21}$$

Note that the error signal in Eq. (4-20) can also be written as

$$e = \mathrm{Im}(Y^* U) = |Y - jU|^2 - |Y + jU|^2 \tag{4-22}$$

or for a simple implementation we can use

$$e = |Y - jU| - |Y + jU| \tag{4-23}$$

The block diagram of the error signal detector based on Eq. (4-23) is shown in Fig. 4-3.

The corresponding closed-loop frequency estimator is shown in Fig. 4-4. The dashed box in this figure and all other figures represents the fact that the hard limiter is optional. This means that the closed-loop estimators can be implemented either with or without such a box.

4.1.5.1. Approximation to the Optimum Error Signal Detector. Implementation of the optimum error signal detector is a little bit complex. To reduce the complexity, we note that

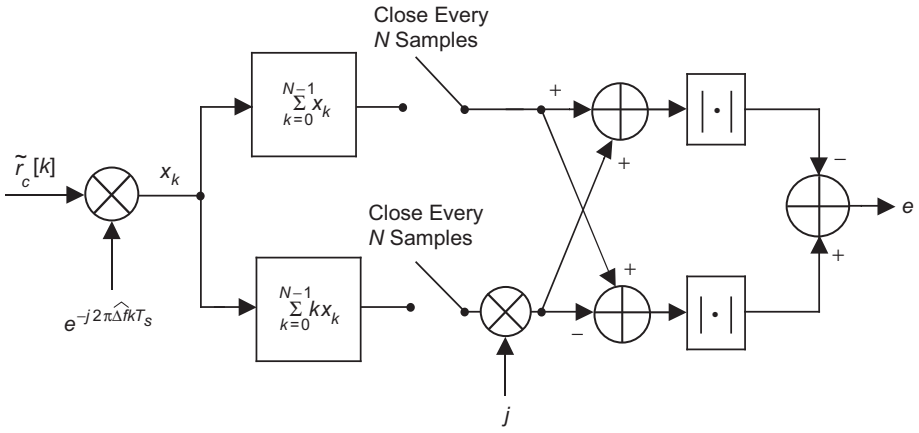


Fig. 4-3. Exact error signal detector, residual carrier.

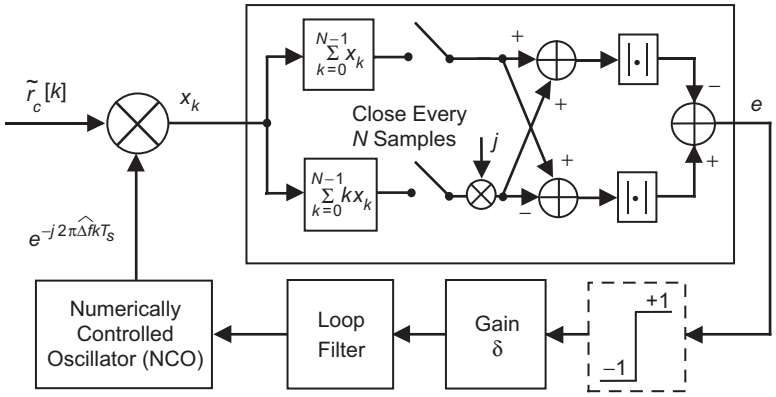


Fig. 4-4. Closed-loop frequency estimator, residual carrier.

$$e = \mathrm{Im}(Y^*U) = \sum_{i=0}^{N-1} \mathrm{Im}(X^*_{0,i}X_{i+1,(N-1)}) \cong C_5\mathrm{Im}(X^*_{0,(N/2)-1}X_{(N/2),N-1}) \tag{4-24}$$

where

$$X_{m,n} = \sum_{k=m}^{n} \tilde{r}_c[k]e^{-j(2\pi\Delta fkT_s)} \tag{4-25}$$

The closed-loop frequency estimator with the approximate error signal detector given by Eq. (4-24) is shown in Fig. 4-5. The parameters $N_w = N/2$ (the

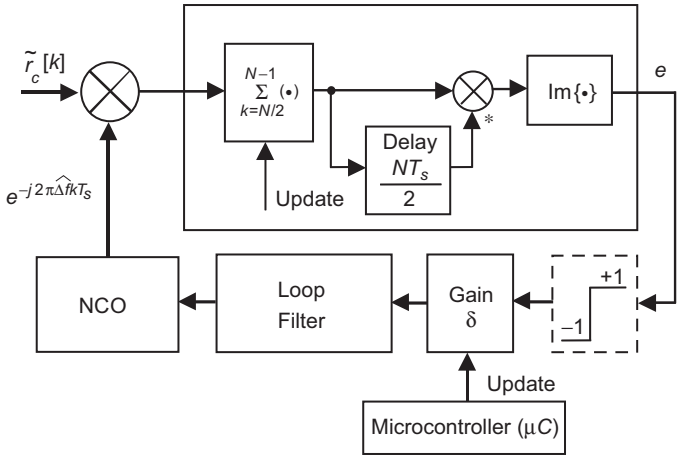


Fig. 4-5. Low-complexity closed-loop frequency correction, residual carrier.

number of samples to be summed, i.e., the window size) and δ (gain) should be optimized and updated after the initial start to perform both the acquisition and tracking of the offset frequency.

4.1.5.2. Digital Loop Filter. The gain δ that was shown in the closed-loop frequency-tracking system is usually part of the digital loop filter. However, here we separate them. Then the digital loop filter without gain δ can be represented as

$$F(z) = 1 + \frac{b}{1 - z^{-1}} \tag{4-26}$$

The corresponding circuit for the digital loop filter is shown in Fig. 4-6. Now in addition to the gain δ, the parameter b also should be optimized to achieve the best performance.

4.1.5.3. Simulation Results. Performance of the closed-loop frequency estimator in Fig. 4-5 was obtained through simulations. First, the acquisition of the closed-loop estimator for a 10-kHz frequency offset is shown in Fig. 4-7. Next the standard deviation of the frequency error versus the received signal-to-noise ratio (SNR) for various initial frequency offsets was obtained. The results of the simulation are shown in Fig. 4-8.

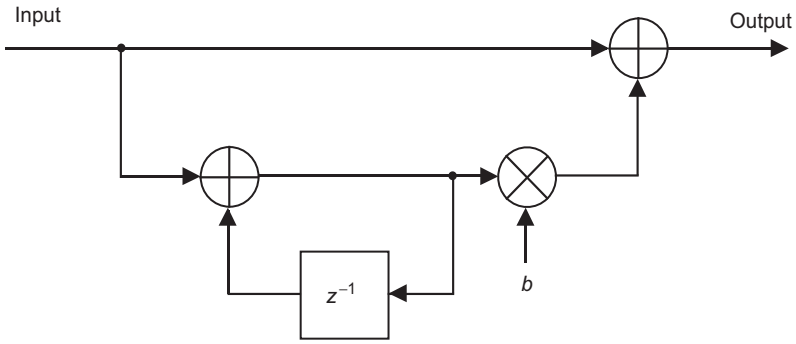


Fig. 4-6. Loop filter for frequency-tracking loops.

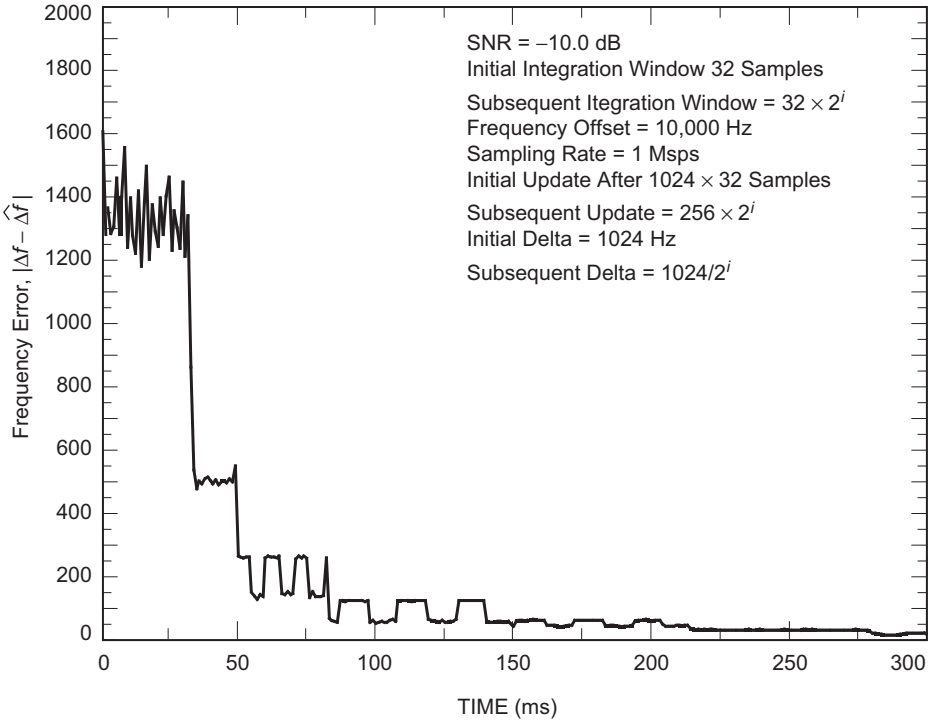


Fig. 4-7. Frequency acquisition performance.

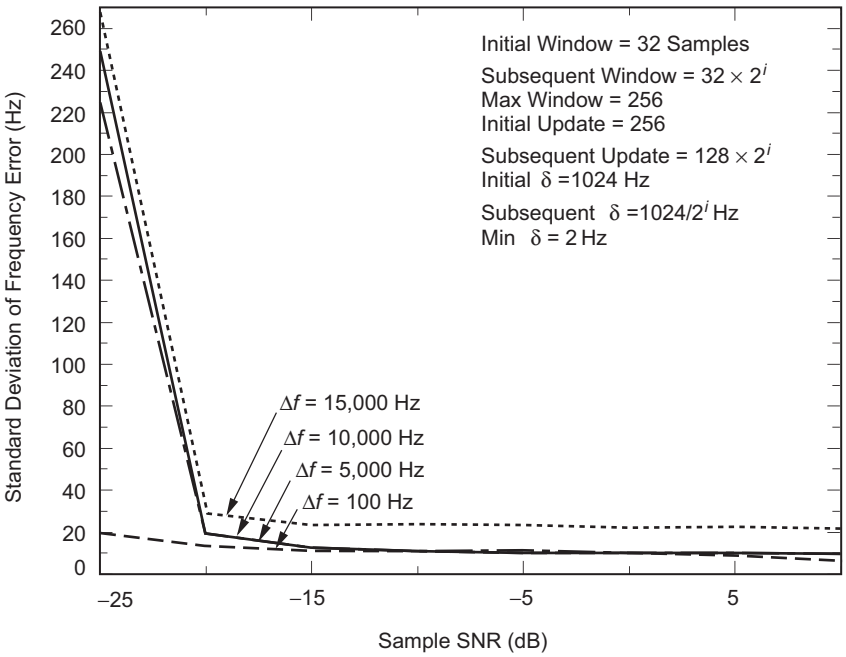


Fig. 4-8. Standard deviation of frequency error.

4.2 Frequency Correction for Known Data-Modulated Signals

Consider a data-modulated signal with no residual (suppressed) carrier. In this section, we assume perfect knowledge of the symbol timing and data (data-aided system). Using again the ML estimation, we derive the open- and closed-loop frequency estimators.

4.2.1. Channel Model

We start with the received baseband analog signal and then derive the discrete-time version of the estimators. Let $\tilde{r}(t)$ be the received complex waveform, and a_i be the complex data representing an M-ary phase-shift keying (M-PSK) modulation or a quadrature amplitude modulation (QAM). Let $p(t)$ be the transmit pulse shaping. Then the received signal can be modeled as

$$\tilde{r}(t) = \sum_{i=-\infty}^{\infty} a_i p(t - iT) e^{j(2\pi \Delta f t + \theta_c)} + \tilde{n}(t) \tag{4-27}$$

where T is the data symbol duration and $\tilde{n}(t)$ is the complex AWGN with two-sided power spectral density N_0 W/Hz per dimension. The conditional pdf of the received observation given the frequency offset Δf and the unknown carrier phase shift θ_c can be written as

$$p(\tilde{\mathbf{r}}|\Delta f, \theta_c) = C_6 e^{-(1/N_0)\int_{-\infty}^{\infty} |\tilde{r}(t) - \sum_{i=-\infty}^{\infty} a_i p(t-iT) e^{j(2\pi \Delta f t + \theta_c)}|^2 dt} \tag{4-28}$$

where C_6 is a constant. Note that

$$\left| \tilde{r}(t) - \sum_{i=-\infty}^{\infty} a_i p(t-iT) e^{j(2\pi \Delta f t + \theta_c)} \right|^2 = |\tilde{r}(t)|^2 + \left| \sum_{i=-\infty}^{\infty} a_i p(t-iT) \right|^2$$

$$- 2 \sum_{i=-\infty}^{\infty} \mathrm{Re}\left\{ a_i^* \tilde{r}(t) p(t-iT) e^{-j(2\pi \Delta f t + \theta_c)} \right\} \tag{4-29}$$

The first two terms do not depend on Δf and θ_c. Then we have

$$p(\tilde{\mathbf{r}}|\Delta f, \theta_c) = C_7 e^{(2/N_0)\mathrm{Re}\left\{ \sum_{i=-\infty}^{\infty} a_i^* z_i(\Delta f) e^{-j\theta_c} \right\}} \tag{4-30}$$

where C_7 is a constant and

$$z_i(\Delta f) = \int_{iT}^{(i+1)T} \tilde{r}(t) p(t-iT) e^{-j(2\pi \Delta f t)} dt \tag{4-31}$$

The conditional pdf in Eq. (4-30) also can be written as

$$p(\tilde{\mathbf{r}}|\Delta f, \theta_c) = C_7 \exp\left[\frac{2}{N_0} |Y| \cos(\theta_c - \arg Y) \right] \tag{4-32}$$

where

$$Y = \sum_{i=-\infty}^{\infty} a_i^* z_i(\Delta f) \tag{4-33}$$

Averaging Eq. (4-32) over θ_c produces

$$P(\tilde{\mathbf{r}}|\Delta f) = C_8 I_0\left(\frac{2}{N_0}|Y|\right) \tag{4-34}$$

where C_8 is a constant. Again, since $I_0(x)$ is an even convex cup $\cup$ function of x, maximizing the right-hand side of Eq. (4-34) is equivalent to maximizing $|Y|$ or equivalently $|Y|^2$. Thus, the ML metric for estimating the frequency offset over the N data symbol interval can be obtained by maximizing the following metric:

$$\lambda(\Delta f) = |Y| = \left|\sum_{k=0}^{N-1} a_k^* z_k(\Delta f)\right| \tag{4-35}$$

4.2.2 Open-Loop Frequency Estimation

For an open-loop estimation, we have

$$\widehat{\Delta f} = \underset{\Delta f}{\operatorname{argmax}}\, \lambda(\Delta f) \tag{4-36}$$

but this operation is equivalent to multiplying the received signal by $e^{-j(2\pi\Delta f t)}$, passing it through the matched filter (MF) with impulse response $p(-t)$, and sampling the result at $t = (k+1)T$, which produces the sequence of z_k's. Next, sum the z_k's, take its magnitude, and then find the maximum value by varying the frequency Δf between $-\Delta f_{max}$ and Δf_{max}, where Δf_{max} is the maximum expected frequency offset. The block diagram to perform these operations is shown in Fig. 4-9.

4.2.3 Closed-Loop Frequency Estimation

The error signal for closed-loop tracking can be obtained as

$$e = \frac{\partial}{\partial \Delta f}\lambda(\Delta f) \tag{4-37}$$

We can approximate the derivative of $\lambda(\Delta f)$ for small ε as in Eq. (4-17). Then we can approximate the error signal as

$$e = |Y(\Delta f + \varepsilon)| - |Y(\Delta f - \varepsilon)| \tag{4-38}$$

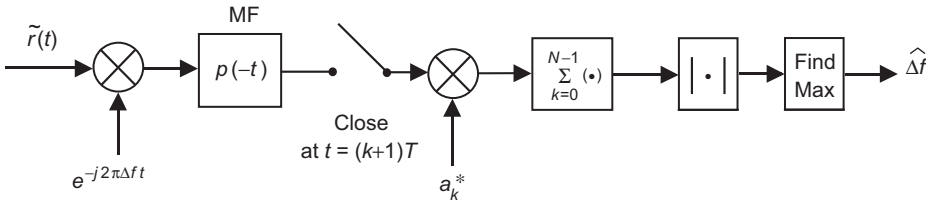


Fig. 4-9. Open-loop frequency estimation for suppressed carrier, known data.

where

$$Y(\Delta f + \varepsilon) = \sum_{k=0}^{N-1} a_k^* z_k(\Delta f + \varepsilon) \tag{4-39}$$

The error signal detector for the closed-loop frequency correction is implemented using the above equations and is shown in Fig. 4-10. In the figure, DAC denotes digital-to-analog converter.

Now again, rather than using the approximate derivative of $\lambda(\Delta f)$, we can take the derivative of $\lambda^2(\Delta f) = |Y|^2$ to obtain the error signal as

$$e = \mathrm{Im}(Y^* U) \tag{4-40}$$

and

$$U = \sum_{k=0}^{N-1} a_k^* u_k(\Delta f) \tag{4-41}$$

where

$$u_i(\Delta f) = \int_{iT}^{(i+1)T} \tilde{r}(t) tp(t - iT) e^{-j(2\pi\Delta f t)} dt \tag{4-42}$$

Thus, $u_k(\Delta f)$ is produced by multiplying $\tilde{r}(t)$ by $e^{-j2\pi\Delta f t}$ and then passing it through a so-called derivative matched filter (DMF)—also called a frequency-matched filter (FMF)—with impulse response $tp(-t)$, and finally sampling the result of this operation at $t = (k+1)T$. Note that the error signal in Eq. (4-40) also can be written as

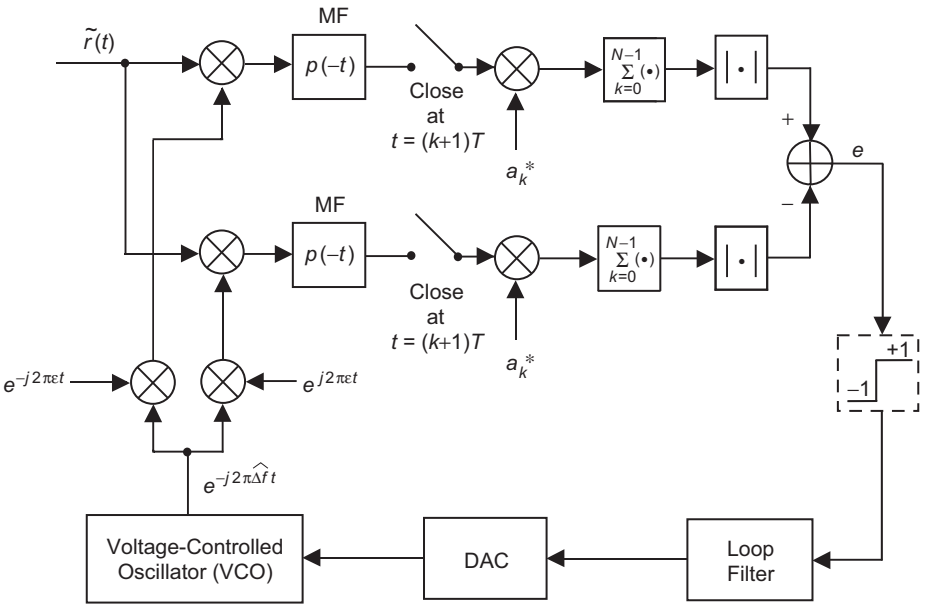


Fig. 4-10. Error signal detector and closed-loop block diagram for suppressed carrier, known data.

$$e = \mathrm{Im}(Y^*U) = |Y - jU|^2 - |Y + jU|^2 \tag{4-43}$$

or, simply, we can use

$$e = |Y - jU| - |Y + jU| \tag{4-44}$$

The block diagram of the closed-loop frequency estimator using the error signal detector given by Eq. (4-40) is shown in Fig. 4-11. Similarly, the block diagram of the closed-loop frequency estimator using the error signal detector given by Eq. (4-44) is shown in Fig. 4-12.

The closed-loop frequency estimator block diagrams shown in this section contain mixed analog and digital circuits. An all-digital version of the closed-loop frequency estimator in Fig. 4-11 operating on the received samples $\tilde{r}[k]$ is shown in Fig. 4-13. In the figure, p_k represents the discrete-time version of the pulse shaping $p(t)$. We assume that there are n samples per data symbol duration T. An all-digital version of other closed-loop estimators can be obtained similarly.

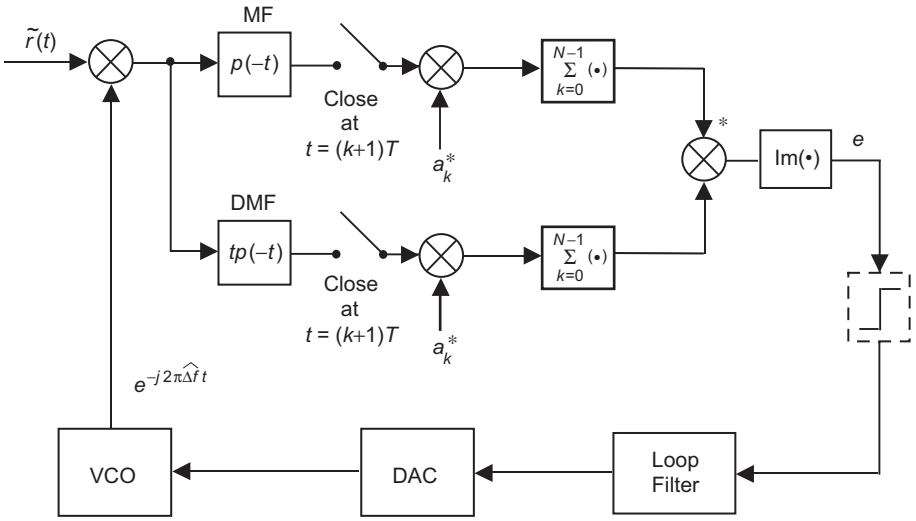


Fig. 4-11. Closed-loop estimator with error signal detector for suppressed carrier, known data, Eq. (4-40).

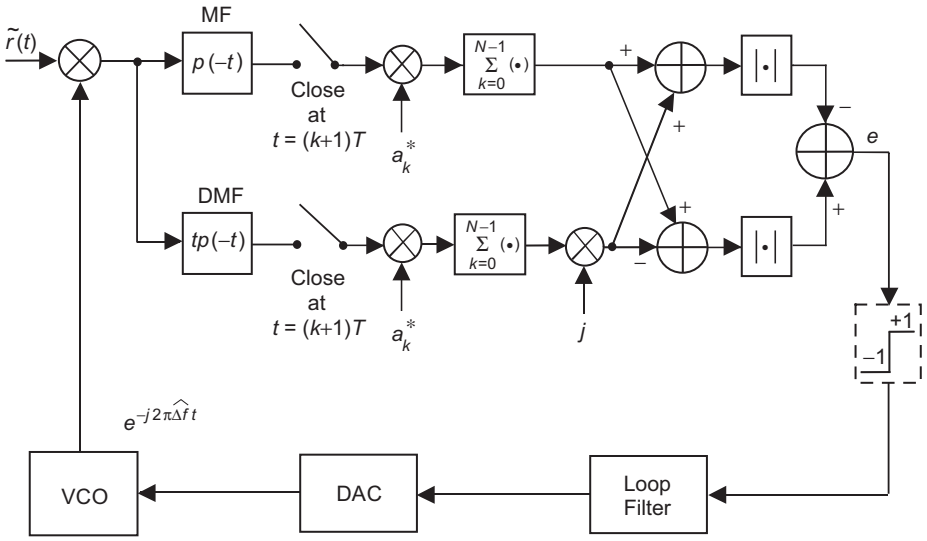


Fig. 4-12. Closed-loop estimator with error signal detector for suppressed carrier, known data, Eq. (4-44).

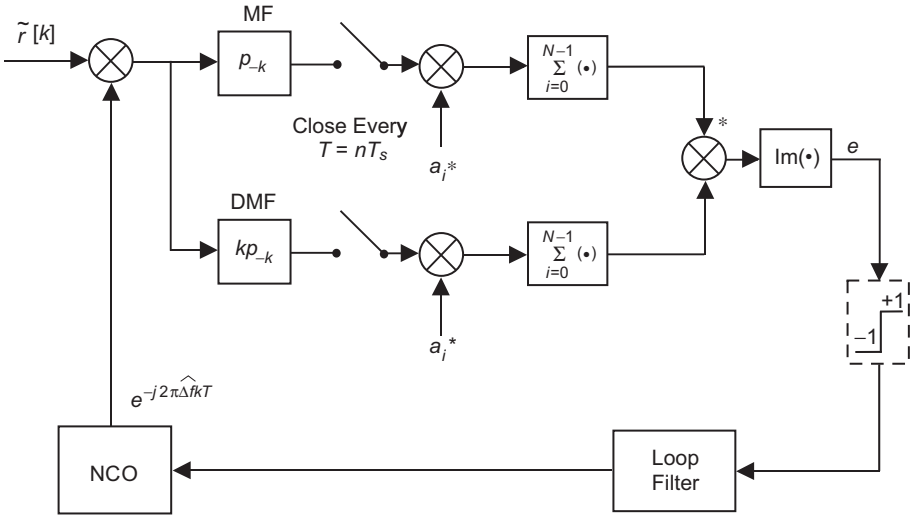


Fig. 4-13. All-digital closed-loop frequency estimator for suppressed carrier, known data.

4.3 Frequency Correction for Modulated Signals with Unknown Data

Consider again a data-modulated signal with no residual (suppressed) carrier. In this section, we assume perfect timing but no knowledge of the data (non-data-aided system). Again using the ML estimation, we derive the open- and closed-loop frequency estimators. In Section 4.2, we obtained the conditional pdf of the received observation given the frequency Δf and data sequence $\mathbf{a}$. We repeat the result here for clarity:

$$P(\tilde{\mathbf{r}}|\Delta f, \mathbf{a}) = C_8 I_0\left(\frac{2}{N_0}|Y|\right) \tag{4-45}$$

where

$$Y = \sum_{i=-\infty}^{\infty} a_i^* z_i(\Delta f) \tag{4-46}$$

and

$$z_i(\Delta f) = \int_{iT}^{(i+1)T} \tilde{r}(t)p(t - iT)e^{-j(2\pi\Delta f t)}dt \tag{4-47}$$

Now we have to average Eq. (4-46) over **a**. Unfortunately, implementation of this averaging is too complex. Instead, first we approximate the $I_0(x)$ function as

$$I_0\left(\frac{2}{N_0}|Y|\right) \cong 1 + \frac{1}{N_0^2}|Y|^2 \tag{4-48}$$

Now we need only to average $|Y|^2$ over the data sequence **a** as

$$E\{|Y|^2\} = E\left\{\left|\sum_{k=0}^{N-1} a_k^* z_k(\Delta f)\right|^2\right\}$$

$$= \sum_{k=0}^{N-1}\sum_{i=0}^{N-1} E\{a_k^* a_i\} z_k(\Delta f) z_i^*(\Delta f)$$

$$= C_a \sum_{k=0}^{N-1} |z_k(\Delta f)|^2 \tag{4-49}$$

where $C_a \triangleq E\{|a_k|^2\}$ and the a_k's are assumed to be zero mean and independent. Thus, estimating the frequency offset over the N data symbol interval can be obtained by maximizing the following metric:

$$\lambda(\Delta f) = \sum_{k=0}^{N-1} |z_k(\Delta f)|^2 \tag{4-50}$$

4.3.1 Open-Loop Frequency Estimation

For open-loop estimation, we have

$$\widehat{\Delta f} = \underset{\Delta f}{\operatorname{argmax}}\, \lambda(\Delta f) \tag{4-51}$$

However, this operation is equivalent to multiplying the received signal by $e^{-j(2\pi\Delta f t)}$, passing it through a matched filter with impulse response $p(-t)$, and sampling the result at $t = (k+1)T$, which produces the sequence of z_k's. Next, take the magnitude square of each z_k, perform summation, and then find

the maximum value by varying the frequency Δf between $-\Delta f_{max}$ and Δf_{max}, where Δf_{max} is the maximum expected frequency offset. The block diagram to perform these operations is shown in Fig. 4-14.

4.3.2 Closed-Loop Frequency Estimation

The error signal for closed-loop tracking can be obtained as

$$e = \frac{\partial}{\partial \Delta f}\lambda(\Delta f) \tag{4-52}$$

We can approximate the derivative of $\lambda(\Delta f)$ for small ε as in Eq. (4-17). Then, we can approximate the error signal as

$$e = \sum_{k=0}^{N-1} \{|z_k(\Delta f + \varepsilon)|^2 - |z_k(\Delta f - \varepsilon)|^2\} \tag{4-53}$$

The error signal detector for the closed-loop frequency correction is implemented using the above equations, and it is shown in Fig. 4-15.

Now again, rather than using the approximate derivative of $\lambda(\Delta f)$, we can take the derivative of $\lambda(\Delta f) = \sum_{k=0}^{N-1} |z_k(\Delta f)|^2$ and obtain the error signal as

$$e = \sum_{k=0}^{N-1} \mathrm{Im}\{z_k^*(\Delta f) u_k(\Delta f)\} \tag{4-54}$$

where

$$u_i(\Delta f) = \int_{iT}^{(i+1)T} \tilde{r}(t) tp(t - iT) e^{-j(2\pi \Delta f t)} dt \tag{4-55}$$

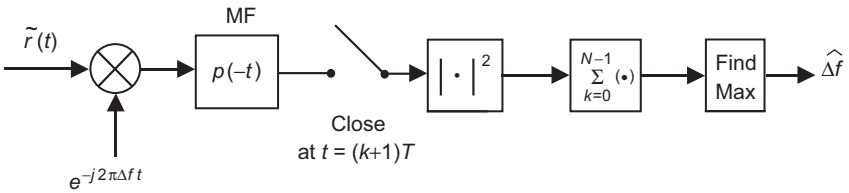


Fig. 4-14. Open-loop frequency estimation for suppressed carrier, unknown data.

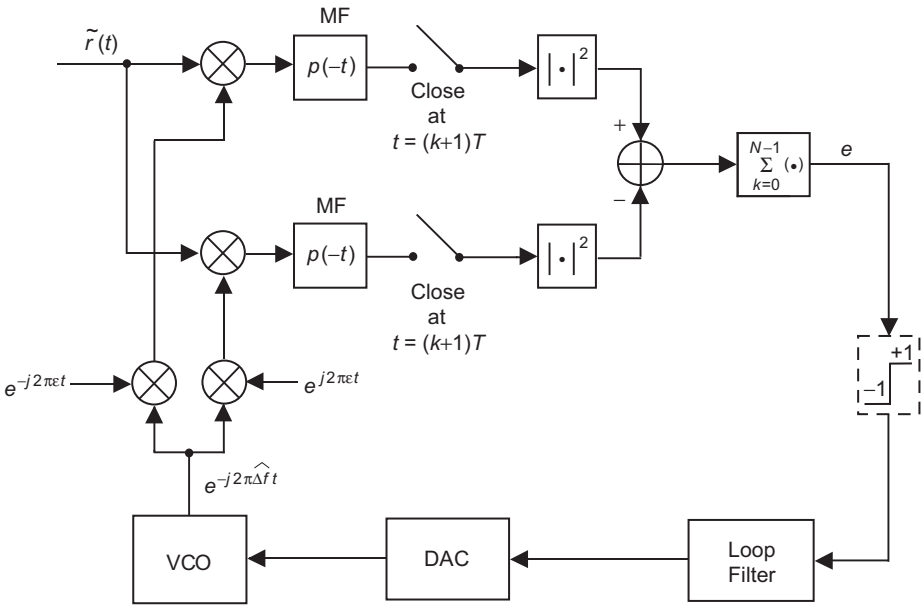


Fig. 4-15. Error signal detector and closed-loop block diagram for suppressed carrier, unknown data.

Note that the error signal in Eq. (4-54) also can be written as

$$e = \sum_{k=0}^{N-1} \{|z_k(\Delta f) - ju_k(\Delta f)|^2 - |z_k(\Delta f) + ju_k(\Delta f)|^2\} \tag{4-56}$$

The block diagram of the closed-loop frequency estimator using the error signal detector given by Eq. (4-54) is shown in Fig. 4-16. Similarly, the block diagram of the closed-loop frequency estimator using the error signal detector given by Eq. (4-56) is shown in Fig. 4-17.

The closed-loop frequency estimator block diagrams shown in this section contain mixed analog and digital circuits. An all-digital version of the closed-loop frequency estimator in Fig. 4-16 operating on the received samples $\tilde{r}[k]$ is shown in Fig. 4-18. All-digital versions of other closed-loop estimators can be obtained similarly.

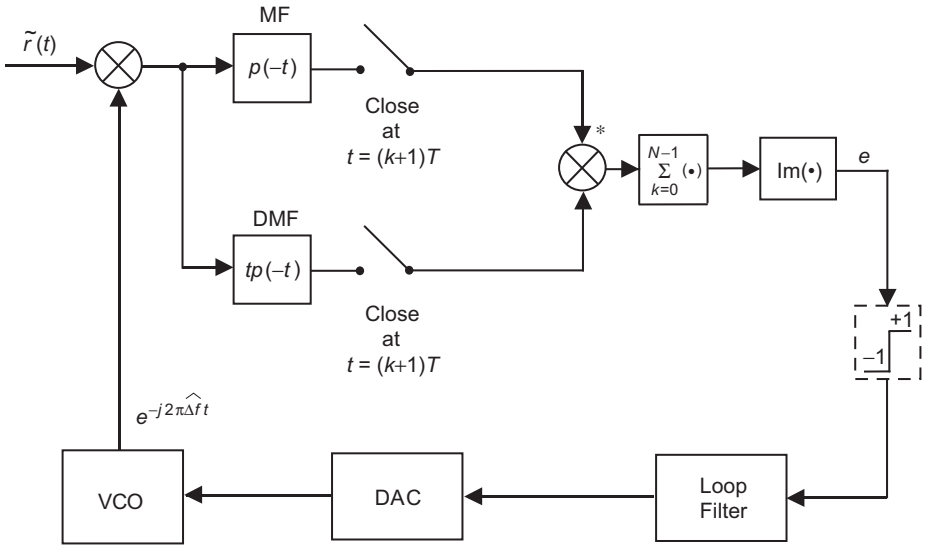


Fig. 4-16. Closed-loop estimator with error signal detector for suppressed carrier, unknown data, Eq. (4-54).

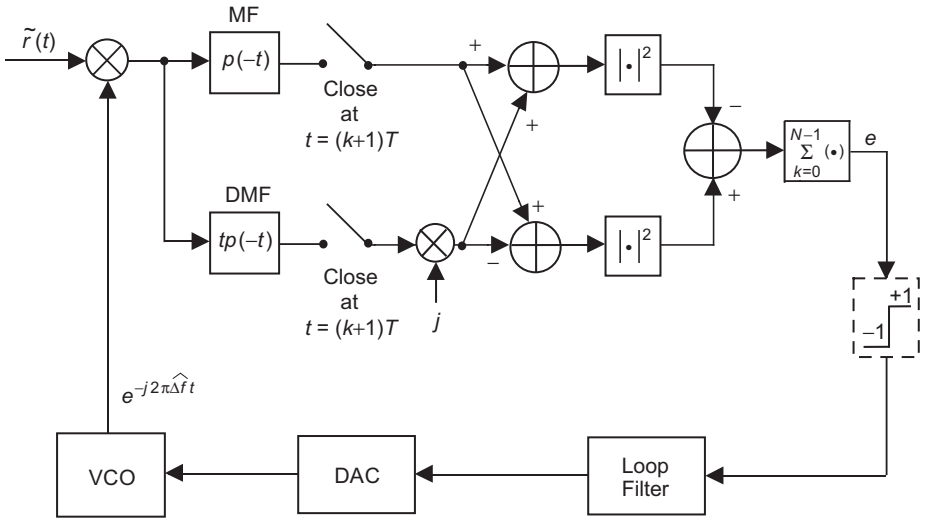


Fig. 4-17. Closed-loop estimator with error signal detector for suppressed carrier, unknown data, Eq. (4-56).

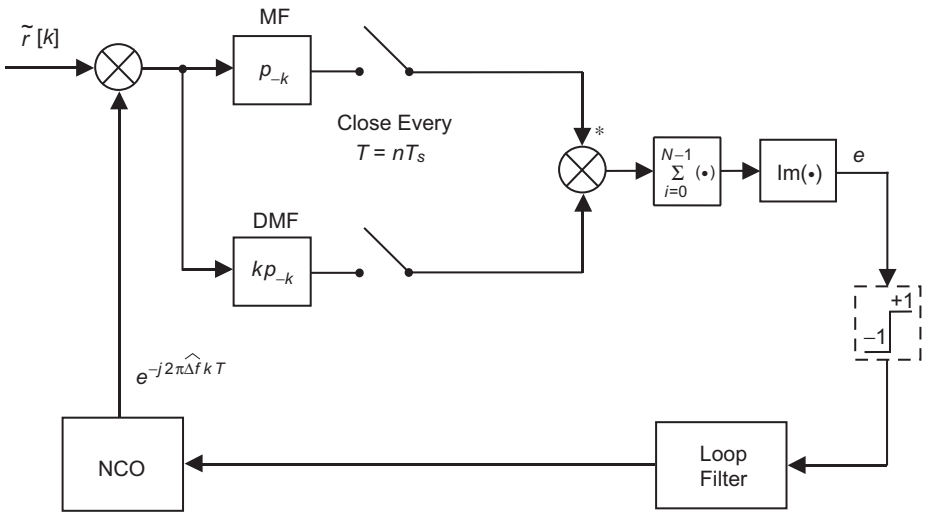


Fig. 4-18. All-digital closed-loop frequency estimator for suppressed carrier, unknown data.

References

[1] H. Meyr and G. Ascheid, *Synchronization in Digital Communications*, New York: John Wiley and Sons Inc., 1990.

[2] H. Meyr, M. Moeneclaey, and S. A. Fechtel, *Digital Communication Receivers*, New York: John Wiley and Sons Inc., 1998.

Chapter 5
Data Format and Pulse Shape Classification

Marvin K. Simon and Dariush Divsalar

In autonomous radio operation, aside from classifying the modulation type, e.g., deciding between binary phase-shift keying (BPSK) and quadrature phase-shift keying (QPSK), it is also desirable to have an algorithm for choosing the data format, e.g., non-return to zero (NRZ) versus Manchester encoding. We will see in our discussions of carrier synchronization in Chapter 8 that, in the absence of subcarriers, when NRZ is employed along with a residual carrier, the carrier-tracking loop takes a loss due to overlapping carrier and modulation spectra, whereas Manchester coding may use either suppressed or residual carrier without such a loss. With this consideration in mind, we shall consider two different scenarios. In one case, independent of the data format, the modulations are assumed to be fully suppressed carrier. In the other case, which is typical of the current Electra radio design, an NRZ data format is always used on a fully suppressed carrier modulation whereas a residual carrier modulation always employs Manchester-coded data. In the latter case, the data format classification algorithm and its performance clearly will be a function of the modulation index, i.e., the allocation of the power to the discrete and data-modulated signal components. Estimation of the modulation index was discussed in Chapter 3.

In this chapter, we derive the maximum-likelihood (ML)-based data format classification algorithms as well as reduced-complexity versions of them obtained by applying suitable approximations of the nonlinearities resulting from the ML formulation. As in previous classification problems of this type, we shall first assume that all other system parameters are known. Following this, we relax the assumption of known carrier phase and, as was done for the modulation classification discussion, we shall consider the noncoherent version of the ML

classifiers. Numerical performance evaluation will be obtained by computer simulations and, wherever possible, by theoretical analyses to verify the simulation results.

5.1 Coherent Classifiers of Data Format for BPSK

5.1.1 Maximum-Likelihood Coherent Classifier of Data Format for BPSK

We begin by considering suppressed-carrier BPSK modulation and a choice between NRZ and Manchester encoding. Thus, the received signal is given by Eqs. (1-3) and (1-6), or in passband by

$$r(t) = \sqrt{2P}\left(\sum_{n=-\infty}^{\infty} c_n p(t-nT)\right)\cos\omega_c t + n(t) \tag{5-1}$$

where P is the signal power,[1] $\{c_n\}$ is the sequence of binary independent, identically distributed (iid) data taking on values ± 1 with equal probability, $p(t)$ is the pulse shape (the item to be classified), ω_c is the radian carrier frequency, $1/T$ is the data (symbol) rate, and $n(t)$ is a bandpass additive white Gaussian noise (AWGN) source with single-sided power spectral density N_0 W/Hz. Based on the above AWGN model, then for an observation of K data intervals, the conditional-likelihood function (CLF) is given by

$$\begin{aligned}
&p\big(r(t)|\{c_n\}, p(t)\big) \\
&= \frac{1}{\sqrt{\pi N_0}}\exp\left(-\frac{1}{N_0}\int_0^{KT}\left[r(t) - \sqrt{2P}\left(\sum_{n=-\infty}^{\infty} c_n p(t-nT)\right)\cos\omega_c t\right]^2 dt\right) \\
&= C\exp\left(\frac{2\sqrt{2P}}{N_0}\sum_{k=0}^{K-1} c_k \int_{kT}^{(k+1)T} r(t)p(t-kT)\cos\omega_c t dt\right) \\
&= C\prod_{k=0}^{K-1}\exp\left(\frac{2\sqrt{2P}}{N_0} c_k \int_{kT}^{(k+1)T} r(t)p(t-kT)\cos\omega_c t dt\right)
\end{aligned} \tag{5-2}$$

[1] There is no need to distinguish between total and data power here since in the suppressed-carrier case all of the signal power is allocated to the data modulation. Thus, for simplicity of notation, we shall simply use P without a subscript to denote signal power.

where C is a constant that has no bearing on the classification. Averaging over the iid data sequence gives

$$p\big(r\,(t)\,|p\,(t)\big) = C\prod_{k=0}^{K-1}\cosh\left(\frac{2\sqrt{2P}}{N_0}\int_{kT}^{(k+1)T} r\,(t)\,p\,(t-kT)\cos\omega_c t dt\right) \tag{5-3}$$

Finally, taking the logarithm of Eq. (5-3), we obtain the log-likelihood function (LLF)

$$\Lambda \triangleq \ln p\big(r(t)|p(t)\big) = \sum_{k=0}^{K-1}\ln\cosh\left(\frac{2\sqrt{2P}}{N_0}\int_{kT}^{(k+1)T} r(t)p(t-kT)\cos\omega_c t dt\right) \tag{5-4}$$

where we have ignored the additive constant $\ln C$.

For NRZ data, $p(t)$ is a unit rectangular pulse of duration T, i.e.,

$$p_1(t) = \begin{cases} 1, & 0 \le t \le T \\ 0, & \text{otherwise} \end{cases} \tag{5-5}$$

For Manchester-encoded data, $p(t)$ is a unit square-wave pulse of duration T, i.e.,

$$p_2(t) = \begin{cases} 1, & 0 \le t \le T/2 \\ -1, & T/2 \le t \le T \end{cases} \tag{5-6}$$

Thus, defining the received observable

$$\begin{aligned} r_k(l) &\triangleq \int_{kT}^{(k+1)T} r(t)p_l(t-kT)\cos\omega_c t dt \\ &= \begin{cases} \int_{kT}^{(k+1)T} r(t)\cos\omega_c t dt; & l=1 \\ \int_{kT}^{(k+1/2)T} r(t)\cos\omega_c t dt - \int_{(k+1/2)T}^{(k+1)T} r(t)\cos\omega_c t dt; & l=2 \end{cases} \end{aligned} \tag{5-7}$$

then a classification choice between the two pulse shapes based on the LLF would be to choose Manchester if

$$\sum_{k=0}^{K-1} \ln\cosh\left(\frac{2\sqrt{2P}}{N_0} r_k(1)\right) < \sum_{k=0}^{K-1} \ln\cosh\left(\frac{2\sqrt{2P}}{N_0} r_k(2)\right) \tag{5-8}$$

Otherwise, choose NRZ.

5.1.2 Reduced-Complexity Data Format BPSK Classifiers

To simplify the form of the classification rule in Eq. (5-8), we replace the $\ln\cosh(\cdot)$ function by its small and large argument approximations. In particular,

$$\ln\cosh x \cong \begin{cases} x^2/2; & x \text{ small} \\ |x| - \ln 2; & x \text{ large} \end{cases} \tag{5-9}$$

Thus, for low signal-to-noise ratio (SNR), Eq. (5-8) simplifies to

$$\sum_{k=0}^{K-1} \left(\int_{kT}^{(k+1)T} r(t) \cos\omega_c t dt \right)^2 < \sum_{k=0}^{K-1} \left(\int_{kT}^{(k+1/2)T} r(t) \cos\omega_c t dt - \int_{(k+1/2)T}^{(k+1)T} r(t) \cos\omega_c t dt \right)^2 \tag{5-10}$$

or

$$\sum_{k=0}^{K-1} \int_{kT}^{(k+1/2)T} r(t) \cos\omega_c t dt \int_{(k+1/2)T}^{(k+1)T} r(\tau) \cos\omega_c \tau d\tau < 0 \tag{5-11}$$

For high SNR, Eq. (5-8) reduces to

$$\sum_{k=0}^{K-1} \left| \int_{kT}^{(k+1/2)T} r(t) \cos\omega_c t dt + \int_{(k+1/2)T}^{(k+1)T} r(t) \cos\omega_c t dt \right| < \sum_{k=0}^{K-1} \left| \int_{kT}^{(k+1/2)T} r(t) \cos\omega_c t dt - \int_{(k+1/2)T}^{(k+1)T} r(t) \cos\omega_c t dt \right| \tag{5-12}$$

Note that while the optimum classifier of Eq. (5-8) requires knowledge of SNR, the reduced-complexity classifiers of Eqs. (5-10) and (5-12) do not. Figure 5-1 is a block diagram of the implementation of the low and high SNR classifiers defined by Eqs. (5-11) and (5-12).

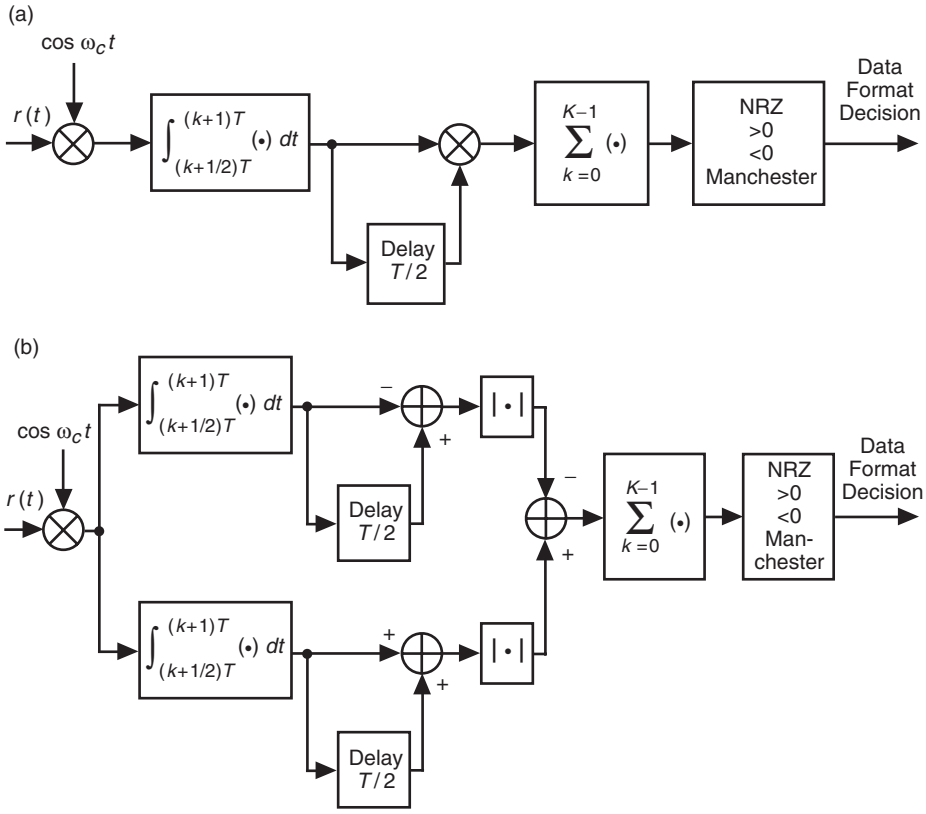


Fig. 5-1. Reduced-complexity coherent data format classifiers for BPSK modulation: (a) low SNR and (b) high SNR.

5.1.3 Probability of Misclassification for Coherent BPSK

5.1.3.1. Exact Evaluation. To illustrate the behavior of the misclassification probability, P_M, with SNR, we consider the low SNR case and evaluate first the probability of the event in Eq. (5-11) given that the transmitted data sequence was in fact NRZ encoded. In particular, we recognize that, given a particular data sequence of K bits,

$$X_{ck} = \int_{kT}^{(k+1/2)T} r(t)\cos\omega_c t dt$$

$$Y_{ck} = \int_{(k+1/2)T}^{(k+1)T} r(\tau)\cos\omega_c t d\tau$$

$k = 0, 1, \cdots, K-1$, are mutually independent and identically distributed (iid) Gaussian random variables (RVs). Thus, the LLF

$$D = \sum_{k=0}^{K-1} \int_{kT}^{(k+1/2)T} r(t) \cos \omega_c t dt \int_{(k+1/2)T}^{(k+1)T} r(\tau) \cos \omega_c \tau d\tau = \sum_{k=0}^{K-1} X_{ck} Y_{ck} \tag{5-13}$$

is a special case of a quadratic form of *real* Gaussian RVs and the probability of the event in Eq. (5-11), namely, $\Pr\{D < 0\}$ can be evaluated in closed form by applying the results in [1, Appendix B] and the additional simplification of these in [2, Appendix 9A]. To see this connection, we define the complex Gaussian RVs $X_k = X_{ck} + jX_{c,k+1}$, $Y_k = Y_{ck} + jY_{c,k+1}$. Then, $X_k Y_k^* + X_k^* Y_k = 2(X_{ck} Y_{ck} + X_{c,k+1} Y_{c,k+1})$. Assuming arbitrarily that K is even, then we can rewrite D of Eq. (5-13) as

$$D = \frac{1}{2} \sum_{k=0}^{K/2-1} (X_k Y_k^* + X_k^* Y_k) \tag{5-14}$$

Comparing Eq. (5-14) with [2, Eq. (B.1)], we see that the former is a special case of the latter, corresponding to $A = B = 0, C = 1/2$. Specifically, making use of the first and second moments of X_k and Y_k given by

$$\begin{aligned} \bar{X}_k &= \bar{Y}_k = (c_k + jc_{k+1})\sqrt{P/8}T \\ \mu_{xx} &= \frac{1}{2} E\left\{ \left| X_k - \bar{X}_k \right|^2 \right\} = N_0 T/8 \\ \mu_{yy} &= \frac{1}{2} E\left\{ \left| Y_{ck} - \bar{Y}_{ck} \right|^2 \right\} = N_0 T/8 \\ \mu_{xy} &= \frac{1}{2} E\left\{ \left(X_{ck} - \bar{X}_{ck} \right) \left(Y_{ck} - \bar{Y}_{ck} \right)^* \right\} = 0 \end{aligned} \tag{5-15}$$

then from [2, Eq. (9A.15)],

$$P_M(1) = \frac{1}{2} + \frac{1}{2^{K-1}} \sum_{k=1}^{K/2} \binom{K-1}{K/2-k} \left[Q_k(a,b) - Q_k(b,a) \right] \tag{5-16}$$

where $Q_k(a,b)$ is the kth-order Marcum Q-function and

$$\begin{aligned} a &= \sqrt{\frac{v\left(\xi_1 v - \xi_2\right)}{2}} \\ b &= \sqrt{\frac{v\left(\xi_1 v + \xi_2\right)}{2}} \end{aligned} \tag{5-17}$$

with

$$\begin{aligned} v &= \sqrt{\frac{1}{\mu_{xx}\mu_{yy}}} = \frac{8}{N_0 T} \\ \xi_1 &= \frac{1}{2}\sum_{k=0}^{K/2-1}\left(\left|\bar{X}_{ck}\right|^2 \mu_{yy} + \left|\bar{Y}_{ck}\right|^2 \mu_{xx}\right) = \frac{KPT^3 N_0}{64} \\ \xi_2 &= \sum_{k=0}^{K/2-1}\left|\bar{X}_{ck}\right|\left|\bar{Y}_{ck}\right| = \frac{KPT^2}{8} \end{aligned} \tag{5-18}$$

Substituting Eq. (5-18) into Eq. (5-17) gives

$$\begin{aligned} a &= 0 \\ b &= \sqrt{K\left(E_s/N_0\right)} \end{aligned} \tag{5-19}$$

where $E_s = PT$ denotes the signal energy. However,

$$\begin{aligned} Q_k(0,b) &= \sum_{n=0}^{k-1}\exp\left(-\frac{b^2}{2}\right)\frac{(b^2/2)^n}{n!} \\ Q_k(b,0) &= 1 \end{aligned} \tag{5-20}$$

Thus, using Eqs. (5-19) and (5-20) in Eq. (5-16) gives the desired result:

$$P_M(1) = \frac{1}{2} + \frac{1}{2^{K_b-1}}\sum_{k=1}^{K/2}\binom{K-1}{K/2-k}\left[\sum_{n=0}^{k-1}\exp\left(-\frac{KE_s}{2N_0}\right)\frac{(KE_s/2N_0)^n}{n!} - 1\right] \tag{5-21}$$

Noting that

$$\sum_{k=1}^{K/2} \binom{K-1}{K/2-k} = 2^{K-2} \tag{5-22}$$

then Eq. (5-21) further simplifies to

$$P_M(1) = \frac{1}{2^{K-1}} \sum_{k=1}^{K/2} \binom{K-1}{K/2-k} \sum_{n=0}^{k-1} \exp\left(-\frac{KE_s}{2N_0}\right) \frac{(KE_s/2N_0)^n}{n!} \tag{5-23}$$

To compute the probability of choosing NRZ when in fact Manchester is the true encoding, we need to evaluate $\Pr\{D \geq 0\} = 1 - \Pr\{D < 0\}$ when instead of Eq. (5-15) we have

$$\begin{aligned} \bar{X}_k &= (c_k + jc_{k+1}) \sqrt{\frac{P}{8}}\, T \\ \bar{Y}_k &= -(c_k + jc_{k+1}) \sqrt{\frac{P}{8}}\, T \end{aligned} \tag{5-24}$$

Since the impact of the negative mean for $\bar{Y}_k$ in Eq. (5-24) is to reverse the sign of ξ_2 in Eq. (5-18), then we immediately conclude that for this case the values of a and b in Eq. (5-19) merely switch roles, i.e.,

$$\begin{aligned} a &= \sqrt{K\left(\frac{E_s}{N_0}\right)} \\ b &= 0 \end{aligned} \tag{5-25}$$

Substituting these values in Eq. (5-16) now gives

$$P_M(2) =$$

$$1 - \left\{ \frac{1}{2} + \frac{1}{2^{K-1}} \sum_{k=1}^{K/2} \binom{K-1}{K/2-k} \left[1 - \sum_{n=0}^{k-1} \exp\left(-\frac{KE_s}{2N_0}\right) \frac{(KE_s/2N_0)^n}{n!} \right] \right\} \tag{5-26}$$

which again simplifies to

$$P_M(2) = \frac{1}{2^{K-1}} \sum_{k=1}^{K/2} \binom{K-1}{K/2-k} \sum_{n=0}^{k-1} \exp\left(-\frac{KE_s}{2N_0}\right) \frac{(KE_s/2N_0)^n}{n!} \qquad (5\text{-}27)$$

Since Eqs. (5-23) and (5-27) are identical, the average probability of mismatch, P_M, is then either of the two results.

Illustrated in Fig. 5-2 are numerical results for the misclassification probability obtained by computer simulation for the optimum and reduced-complexity data format classifiers as given by Eqs. (5-8), (5-11) and (5-12). Also illustrated are the numerical results obtained from the closed-form analytical solution given in Eq. (5-23) for the low-SNR reduced-complexity scheme. As can be seen, the agreement between theoretical and simulated results is exact. Furthermore, the difference in performance between the optimum and reduced-complexity classifiers is quite small over a large range of SNRs.

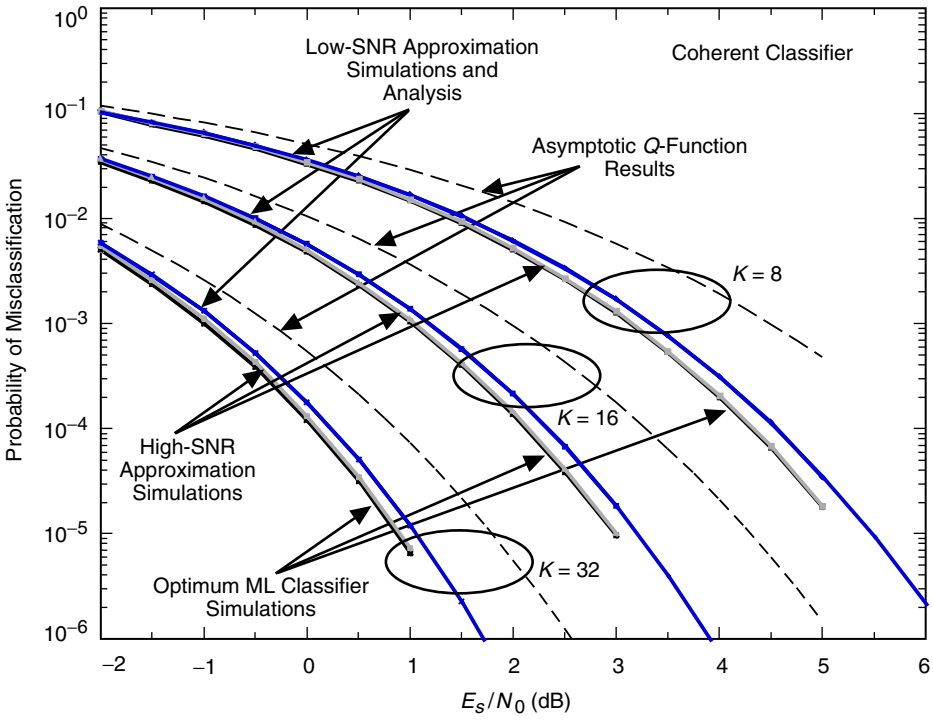


Fig. 5-2. A comparison of the performance of coherent data format classifiers for BPSK modulation.

5.1.3.2 Asymptotic Behavior. To evaluate the asymptotic (large K) behavior of the misclassification probability, we apply the central limit theorem to the quadratic form in Eq. (5-13). Specifically, in the limit of large K, D tends to a Gaussian RV with mean

$$\bar{D} = K\bar{X}_{ck}\bar{Y}_{ck} = \frac{KPT^2}{8} \tag{5-28}$$

and variance

$$\sigma_D^2 = K\text{var}\left\{X_{ck}Y_{ck}\right\} = K\left[\overline{X_{ck}^2}\,\overline{Y_{ck}^2} - \overline{X}_{ck}^2\,\overline{Y}_{ck}^2\right] \tag{5-29}$$

After some manipulation, it can be shown that Eq. (5-29) can be expressed as

$$\begin{aligned}\sigma_D^2 &= K\left[\text{var}\left\{X_{ck}\right\}\text{var}\left\{Y_{ck}\right\} + \text{var}\left\{X_{ck}\right\}\overline{Y_{ck}^2} + \text{var}\left\{Y_{ck}\right\}\overline{X_{ck}^2}\right] \\ &= K\left[\left(\frac{N_0T}{8}\right)^2 + 2\left(\frac{N_0T}{8}\right)\left(\frac{PT^2}{8}\right)\right] = K\left(\frac{N_0T}{8}\right)^2\left(1 + 2\frac{E_s}{N_0}\right)\end{aligned} \tag{5-30}$$

Thus, in view of the Gaussian assumption, $P_M = \Pr\left\{D < 0\right\}$ is obtained in the form of a Gaussian Q-function, namely,

$$P_M = Q\left(\frac{\bar{D}}{\sigma_D}\right) = Q\left(\sqrt{K\frac{\left(E_s/N_0\right)^2}{1 + 2E_s/N_0}}\right) \tag{5-31}$$

The asymptotic misclassification probability of Eq. (5-31) is superimposed on the results in Fig. 5-1.

5.2 Coherent Classifiers of Data Format for QPSK

5.2.1 Maximum-Likelihood Coherent Classifier of Data Format for QPSK

For QPSK modulation, the received signal is given by

$$r(t) = \sqrt{P}\left(\sum_{n=-\infty}^{\infty} c_n p(t-nT)\right)\cos\omega_c t + \sqrt{P}\left(\sum_{n=-\infty}^{\infty} b_n p(t-nT)\right)\sin\omega_c t + n(t) \tag{5-32}$$

where now $\{c_n\}$ and $\{b_n\}$ are the in-phase (I) and quadrature (Q) sequences of binary iid data taking on values ± 1 with equal probability. For simplicity, we have assumed that the I and Q baseband waveforms have the same data format. For an observation of K symbol intervals, each of duration T, the CLF is given by[2]

$$\begin{aligned}
&p\big(r(t)|\{c_n\},\{b_n\},p(t)\big) \\
&= \frac{1}{\sqrt{\pi N_0}}\exp\left\{-\frac{1}{N_0}\int_0^{KT}\left[r(t) - \sqrt{P}\left(\sum_{n=-\infty}^{\infty} c_n p(t-nT)\right)\cos\omega_c t \right.\right. \\
&\quad \left.\left. -\sqrt{P}\left(\sum_{n=-\infty}^{\infty} b_n p(t-nT)\right)\sin\omega_c t\right]^2 dt\right\} \\
&= C\exp\left(\frac{2\sqrt{P}}{N_0}\sum_{k=0}^{K-1} c_k \int_{kT}^{(k+1)T} r(t)\,p(t-kT)\cos\omega_c t dt\right) \\
&\quad \times \exp\left(\frac{2\sqrt{P}}{N_0}\sum_{k=0}^{K-1} b_k \int_{kT}^{(k+1)T} r(t)\,p(t-kT)\sin\omega_c t dt\right) \\
&= C\prod_{k=0}^{K-1}\exp\left(\frac{2\sqrt{P}}{N_0} c_k \int_{kT}^{(k+1)T} r(t)\,p(t-kT)\cos\omega_c t dt\right) \\
&\quad \times \exp\left(\frac{2\sqrt{P}}{N_0}\sum_{k=0}^{K-1} b_k \int_{kT}^{(k+1)T} r(t)\,p(t-kT)\sin\omega_c t dt\right)
\end{aligned} \tag{5-33}$$

[2] As in other chapters, we again assume a system with a fixed modulation bandwidth or, equivalently, a fixed symbol rate. Thus, under this assumption, T, which denotes the duration of a modulation symbol, is equal to two bit times for QPSK and is equal to a single bit time for BPSK.

Averaging over the iid data sequences and taking the logarithm gives the LLF

$$\Lambda \triangleq \ln p\big(r\left(t\right)|p\left(t\right)\big) = \sum_{k=0}^{K-1}\left[\ln\cosh\left(\frac{2\sqrt{P}}{N_0}\int_{kT}^{(k+1)T} r\left(t\right)p\left(t-kT\right)\cos\omega_c t dt\right)\right.$$

$$\left. + \ln\cosh\left(\frac{2\sqrt{P}}{N_0}\int_{kT}^{(k+1)T} r\left(t\right)p\left(t-kT\right)\sin\omega_c t dt\right)\right] \tag{5-34}$$

Analogous to Eq. (5-7), defining the received I and Q observables

$$\begin{aligned} r_{ck}\left(l\right) &\triangleq \int_{kT}^{(k+1)T} r\left(t\right)p_l\left(t-kT\right)\cos\omega_c t dt \\ r_{sk}\left(l\right) &\triangleq \int_{kT}^{(k+1)T} r\left(t\right)p_l\left(t-kT\right)\sin\omega_c t dt \end{aligned} \tag{5-35}$$

then the classification rule for choosing the data format is as follows: Choose Manchester encoding if

$$\sum_{k=0}^{K-1}\left[\ln\cosh\left(\frac{2\sqrt{P}}{N_0}r_{ck}\left(1\right)\right) + \ln\cosh\left(\frac{2\sqrt{P}}{N_0}r_{sk}\left(1\right)\right)\right] <$$

$$\sum_{k=0}^{K-1}\left[\ln\cosh\left(\frac{2\sqrt{P}}{N_0}r_{ck}\left(2\right)\right) + \ln\cosh\left(\frac{2\sqrt{P}}{N_0}r_{sk}\left(2\right)\right)\right] \tag{5-36}$$

Otherwise, choose NRZ.

5.2.2 Reduced-Complexity Data Format QPSK Classifiers

Here again we may simplify the form of the classification rule in Eq. (5-36) by using the nonlinearity approximations in Eq. (5-9). For example, for low SNR, the classification decision would be based on the inequality

$$\sum_{k=0}^{K-1}\left[\int_{kT}^{(k+1/2)T} r\left(t\right)\cos\omega_c t dt \int_{(k+1/2)T}^{(k+1)T} r\left(\tau\right)\cos\omega_c \tau d\tau\right.$$

$$\left.+\int_{kT}^{(k+1/2)T} r\left(t\right)\sin\omega_c t dt \int_{(k+1/2)T}^{(k+1)T} r\left(\tau\right)\sin\omega_c \tau d\tau\right] < 0 \qquad (5\text{-}37)$$

Figure 5-3 illustrates the implementation of the classifier defined above.

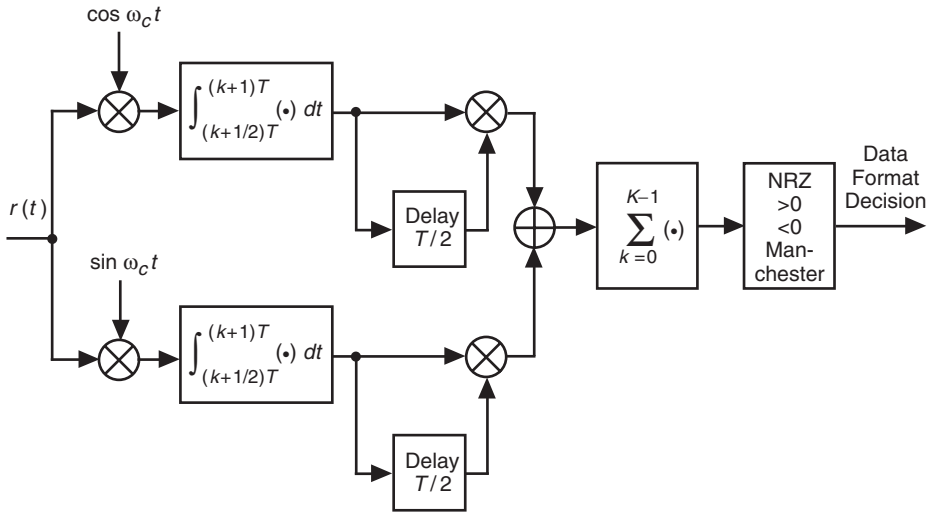


Fig. 5-3. Reduced-complexity coherent data format classifiers for QPSK modulation, for low SNR.

5.2.3 Probability of Misclassification for Coherent QPSK

Defining $X_{sk} = \int_{kT}^{(k+1/2)T} r\left(t\right)\sin\omega_c t dt, Y_{sk} = \int_{(k+1/2)T}^{(k+1)T} r\left(\tau\right)\sin\omega_c t d\tau; k = 0, 1, \cdots, K-1$, and assigning them to the complex Gaussian RVs $X_{k+K/2} = X_{sk} + jX_{s,k+1}, Y_{k+K/2} = Y_{sk} + jY_{s,k+1}$, then analogous to Eq. (5-14) we can write

$$D = \frac{1}{2}\sum_{k=0}^{K-1}\left(X_k Y_k^* + X_k^* Y_k\right) \qquad (5\text{-}38)$$

where the means of the observables are now given by

$$\bar{X}_k = \bar{Y}_k = (a_k + ja_{k+1})\sqrt{P/16}T;\ k = 0, 1, \cdots, K/2 - 1$$
$$\bar{X}_k = \bar{Y}_k = (b_k + jb_{k+1})\sqrt{P/16}T;\ k = K/2, K/2 + 1, \cdots, K - 1 \tag{5-39}$$

Since all the observables are again mutually iid Gaussian RVs, then the LLF in Eq. (5-38) is still a quadratic form of Gaussian RVs and the probability $\Pr\{D < 0\}$ can be evaluated in closed form in the same manner as before. Note that because of the assumption of a fixed modulation bandwidth the probabilities of misclassification for BPSK and QPSK are different, whereas had we assumed an equivalence between the information (bit) rates of the two modulations, these probabilities would have become equal.

Before moving on to a discussion of noncoherent data format classification schemes, it is of interest to ask whether there exists a universal classification rule that for a given symbol rate (modulation bandwidth) is appropriate (but not necessarily optimum) for M-ary phase-shift keying (M-PSK) independent of the value of M. This would allow determination of the data format prior to modulation classification, which for M-PSK constitutes determining the value of M. Before answering this question, we first point out that the low SNR ML classification rule of Eq. (5-37), which is explicitly derived for QPSK modulation, also would work for BPSK, albeit with a penalty in performance relative to the ML rule of Eq. (5-11) due to the presence now of a noise-only term in the quadrature channel. Having said this, it also can be demonstrated that the classification rule of Eq. (5-37), which can be viewed as the extension of the classification rule in Eq. (5-11) to complex observables, is also suitable for M-PSK ($M > 4$), and furthermore the misclassification performance of this scheme still would be given by Eq. (5-23) independent now of the value of M.

5.3 Noncoherent Classification of Data Format for BPSK

5.3.1 Maximum-Likelihood Noncoherent Classifier of Data Format for BPSK

Here we assume that the carrier has a time-invariant random phase, θ_c, that is unknown and uniformly distributed. Thus, the received signal of Eq. (5-1) is now modeled as

$$r(t) = \sqrt{2P}\left(\sum_{n=-\infty}^{\infty} c_n p\left(t - nT\right)\right)\cos\left(\omega_c t + \theta_c\right) + n(t) \tag{5-40}$$

and the corresponding CLF becomes

$$p\big(r(t)|\{c_n\},p(t),\theta_c\big) =$$

$$C\prod_{k=0}^{K-1}\exp\left(\frac{2\sqrt{2P}}{N_0}c_k\int_{kT}^{(k+1)T} r\left(t\right)p\left(t-kT\right)\cos\left(\omega_c t+\theta_c\right)dt\right) \quad (5\text{-}41)$$

At this point, we have the option of first averaging over the random carrier phase and then the data or vice versa. Considering the first option, we start by rewriting Eq. (5-41) as

$$p\big(r(t)|\{c_n\},p(t),\theta_c\big) =$$

$$C\exp\left(\frac{2\sqrt{2P}}{N_0}\sqrt{\left(\sum_{k=0}^{K-1}c_k r_{ck}\right)^2+\left(\sum_{k=0}^{K-1}c_k r_{sk}\right)^2}\cos\left(\theta_c+\eta\right)\right) \quad (5\text{-}42)$$

$$\eta=\tan^{-1}\frac{\sum\limits_{k=0}^{K-1}c_k r_{sk}}{\sum\limits_{k=0}^{K-1}c_k r_{ck}}$$

Averaging over the carrier phase results in (ignoring constants)

$$p\big(r(t)|\{c_n\},p(t)\big)=I_0\left(\frac{2\sqrt{2P}}{N_0}\sqrt{\left(\sum_{k=0}^{K-1}c_k r_{ck}\right)^2+\left(\sum_{k=0}^{K-1}c_k r_{sk}\right)^2}\right) \quad (5\text{-}43)$$

where $I_0(\cdot)$ is the zero-order modified Bessel function of the first kind. Unfortunately, the average over the data sequence cannot be obtained in closed form. Hence, the classification algorithm can be stated only as follows: Given that NRZ was transmitted, choose the Manchester format if

$$\underset{\mathbf{c}}{E}\left\{I_0\left(\frac{2\sqrt{2P}}{N_0}\sqrt{\left(\sum_{k=0}^{K-1}c_k r_{ck}\left(1\right)\right)^2+\left(\sum_{k=0}^{K-1}c_k r_{sk}\left(1\right)\right)^2}\right)\right\}<$$

$$\underset{\mathbf{c}}{E}\left\{I_0\left(\frac{2\sqrt{2P}}{N_0}\sqrt{\left(\sum_{k=0}^{K-1}c_k r_{ck}\left(2\right)\right)^2+\left(\sum_{k=0}^{K-1}c_k r_{sk}\left(2\right)\right)^2}\right)\right\} \quad (5\text{-}44)$$

where $\underset{\mathbf{c}}{E}\{\cdot\}$ denotes expectation over the data sequence $\mathbf{c} = (c_0, c_1, \cdots, c_{K-1})$. Otherwise, choose NRZ.

Consider now the second option, where we first average over the data sequence. Then,

$$
\begin{aligned}
& p\big(r(t)|p(t),\theta_c\big) \\
& \quad = C \prod_{k=0}^{K-1} \underset{c_k}{E} \left\{ \exp\left(\frac{2\sqrt{2P}}{N_0} c_k \int_{kT}^{(k+1)T} r\,(t)\, p\,(t-kT) \cos\left(\omega_c t + \theta_c\right) dt \right) \right\} \\
& \quad = C \exp\left[\ln\left(\prod_{k=0}^{K-1} \underset{c_k}{E} \left\{ \exp\left(\frac{2\sqrt{2P}}{N_0} c_k \right.\right.\right. \\
& \qquad \left.\left.\left. \times \int_{kT}^{(k+1)T} r(t) p(t-kT) \cos\left(\omega_c t + \theta_c\right) dt \right) \right\} \right) \right] \\
& \quad = C \exp\left[\sum_{k=0}^{K-1} \ln\left(\underset{c_k}{E} \left\{ \exp\left(\frac{2\sqrt{2P}}{N_0} c_k \right.\right.\right. \\
& \qquad \left.\left.\left. \times \int_{kT}^{(k+1)T} r(t) p(t-kT) \cos\left(\omega_c t + \theta_c\right) dt \right) \right\} \right) \right] \\
& \quad = C \exp\left[\sum_{k=0}^{K-1} \ln\cosh\left(\frac{2\sqrt{2P}}{N_0} \int_{kT}^{(k+1)T} r(t) p(t-kT) \cos(\omega_c t + \theta_c) dt \right) \right]
\end{aligned}
\tag{5-45}
$$

Thus, a classification between NRZ and Manchester encoding would be based on a comparison of

$$
LR = \frac{\underset{\theta_c}{E} \left\{ \exp\left[\sum_{k=0}^{K-1} \ln\cosh\left(\frac{2\sqrt{2P}}{N_0} \int_{kT}^{(k+1)T} r\,(t)\, p_1\,(t-kT) \cos\left(\omega_c t + \theta_c\right) dt \right) \right] \right\}}{\underset{\theta_c}{E} \left\{ \exp\left[\sum_{k=0}^{K-1} \ln\cosh\left(\frac{2\sqrt{2P}}{N_0} \int_{kT}^{(k+1)T} r\,(t)\, p_2\,(t-kT) \cos\left(\omega_c t + \theta_c\right) dt \right) \right] \right\}}
\tag{5-46}
$$

to unity.

To simplify matters, before averaging over the carrier phase, one must employ the approximations to the nonlinearities given in Eq. (5-9). In particular, for low SNR, we have

$$
\begin{aligned}
&p\big(r(t)|p(t)\big) \\
&= \underset{\theta_c}{E}\left\{\exp\left[\frac{1}{2}\sum_{k=0}^{K-1}\left(\frac{2\sqrt{2P}}{N_0}\int_{kT}^{(k+1)T} r\,(t)\,p\,(t-kT)\cos\left(\omega_c t+\theta_c\right)dt\right)^2\right]\right\} \\
&= \underset{\theta_c}{E}\left\{\exp\left[\frac{4P}{N_0^2}\sum_{k=0}^{K-1}\left(r_{ck}\cos\theta_c - r_{sk}\sin\theta_c\right)^2\right]\right\} \\
&= \underset{\theta_c}{E}\left\{\exp\left[\frac{4P}{N_0^2}\sum_{k=0}^{K-1}\left(r_{ck}^2+r_{sk}^2\right)\cos^2\left(\theta_c+\eta_k\right)\right]\right\} \\
&= \exp\left[\frac{2P}{N_0^2}\sum_{k=0}^{K-1}\left(r_{ck}^2+r_{sk}^2\right)\right] \\
&\quad\times \underset{\theta_c}{E}\left\{\exp\left[\left(\frac{2P}{N_0^2}\sum_{k=0}^{K-1}\left(r_{ck}^2+r_{sk}^2\right)\cos\left(2\left(\theta_c+\eta_k\right)\right)\right)\right]\right\} \\
&= \exp\left[\frac{2P}{N_0^2}\sum_{k=0}^{K-1}\left(r_{ck}^2+r_{sk}^2\right)\right] \\
&\quad\times \underset{\theta_c}{E}\left\{\exp\left[\left(\frac{2P}{N_0^2}\left(\cos 2\theta_c\sum_{k=0}^{K-1}\left(r_{ck}^2+r_{sk}^2\right)\cos 2\eta_k\right.\right.\right.\right. \\
&\quad\left.\left.\left.\left.-\ \sin 2\theta_c\sum_{k=0}^{K-1}\left(r_{ck}^2+r_{sk}^2\right)\sin 2\eta_k\right)\right)\right]\right\} \\
&= \exp\left[\frac{2P}{N_0^2}\sum_{k=0}^{K-1}\left(r_{ck}^2+r_{sk}^2\right)\right] \\
&\quad\times I_0\left(\frac{2P}{N_0^2}\sqrt{\left(\sum_{k=0}^{K-1}\left(r_{ck}^2+r_{sk}^2\right)\cos 2\eta_k\right)^2+\left(\sum_{k=0}^{K-1}\left(r_{ck}^2+r_{sk}^2\right)\sin 2\eta_k\right)^2}\right)
\end{aligned}
\tag{5-47}
$$

where

$$\eta_k = \tan^{-1} \frac{r_{sk}}{r_{ck}} \tag{5-48}$$

Thus, since

$$\begin{aligned} \cos 2\eta_k &= \frac{r_{ck}^2 - r_{sk}^2}{r_{ck}^2 + r_{sk}^2} \\ \sin 2\eta_k &= \frac{2 r_{ck} r_{sk}}{r_{ck}^2 + r_{sk}^2} \end{aligned} \tag{5-49}$$

we finally have

$$\begin{aligned} p\big(r(t)|p(t)\big) = &\exp\left[\frac{2P}{N_0^2}\sum_{k=0}^{K-1}\left(r_{ck}^2 + r_{sk}^2\right)\right] \\ &\times I_0\left(\frac{2P}{N_0^2}\sqrt{\left(\sum_{k=0}^{K-1}\left(r_{ck}^2 - r_{sk}^2\right)\right)^2 + 4\left(\sum_{k=0}^{K-1} r_{ck} r_{sk}\right)^2}\right) \\ = &\exp\left[\frac{2P}{N_0^2}\sum_{k=0}^{K-1}\left(r_{ck}^2 + r_{sk}^2\right)\right] I_0\left(\frac{2P}{N_0^2}\left|\sum_{k=0}^{K-1}\tilde{r}_k^2\right|\right) \end{aligned} \tag{5-50}$$

where

$$\tilde{r}_k \triangleq r_{ck} + j r_{sk} = \int_{kT}^{(k+1)T} r(t)\, p(t - kT)\, e^{j\omega_c t} dt \tag{5-51}$$

Finally then, the classification decision rule analogous to Eq. (5-44) is: Given that NRZ data were transmitted, decide on Manchester coding if

$$\begin{aligned} &\exp\left[\frac{2P}{N_0^2}\sum_{k=0}^{K-1}\left|\tilde{r}_k(1)\right|^2\right] I_0\left(\frac{2P}{N_0^2}\left|\sum_{k=0}^{K-1}\tilde{r}_k^2(1)\right|\right) < \\ &\qquad \exp\left[\frac{2P}{N_0^2}\sum_{k=0}^{K-1}\left|\tilde{r}_k(2)\right|^2\right] I_0\left(\frac{2P}{N_0^2}\left|\sum_{k=0}^{K-1}\tilde{r}_k^2(2)\right|\right) \end{aligned} \tag{5-52}$$

Equivalently, normalizing the observables to

$$\tilde{r}'_k \triangleq \frac{1}{T}\int_{kT}^{(k+1)T} \frac{r(t)}{\sqrt{2P}} p(t-kT)\, e^{j\omega_c t} dt \tag{5-53}$$

then Eq. (5-52) becomes

$$\exp\left[\left(\frac{2E_s}{N_0}\right)^2 \sum_{k=0}^{K-1} |\tilde{r}'_k(1)|^2\right] I_0\left(\left(\frac{2E_s}{N_0}\right)^2 \left|\sum_{k=0}^{K-1} \tilde{r}'^2_k(1)\right|\right) <$$

$$\exp\left[\left(\frac{2E_s}{N_0}\right)^2 \sum_{k=0}^{K-1} |\tilde{r}'_k(2)|^2\right] I_0\left(\left(\frac{2E_s}{N_0}\right)^2 \left|\sum_{k=0}^{K-1} \tilde{r}'^2_k(2)\right|\right) \tag{5-54}$$

Since we have already assumed low SNR in arriving at Eq. (5-54), we can further approximate the nonlinearities in that equation by their values for small arguments. Retaining only linear terms, we arrive at the simplification

$$\sum_{k=0}^{K-1} |\tilde{r}'_k(1)|^2 < \sum_{k=0}^{K-1} |\tilde{r}'_k(2)|^2 \tag{5-55}$$

or, equivalently,

$$\sum_{k=0}^{K-1} |\tilde{r}_k(1)|^2 < \sum_{k=0}^{K-1} |\tilde{r}_k(2)|^2 \tag{5-56}$$

which again does not require knowledge of SNR. On the other hand, if we retain second-order terms, then Eq. (5-54) simplifies to

$$\sum_{k=0}^{K-1} |\tilde{r}'_k(1)|^2 + \frac{1}{4}\left(\frac{2E_s}{N_0}\right)^2 \left[2\left(\sum_{k=0}^{K-1} |\tilde{r}'_k(1)|^2\right)^2 + \left|\sum_{k=0}^{K-1} \tilde{r}'^2_k(1)\right|^2\right] <$$

$$\sum_{k=0}^{K-1} |\tilde{r}'_k(2)|^2 + \frac{1}{4}\left(\frac{2E_s}{N_0}\right)^2 \left[2\left(\sum_{k=0}^{K-1} |\tilde{r}'_k(2)|^2\right)^2 + \left|\sum_{k=0}^{K-1} \tilde{r}'^2_k(2)\right|^2\right] \tag{5-57}$$

which is SNR-dependent.

Expanding Eq. (5-56) in the form of Eq. (5-10), we obtain

$$\sum_{k=0}^{K-1}\left(\int_{kT}^{(k+1)T} r\left(t\right)\cos\omega_c t dt\right)^2+\left(\int_{kT}^{(k+1)T} r\left(t\right)\sin\omega_c t dt\right)^2 <$$

$$\sum_{k=0}^{K-1}\left(\int_{kT}^{(k+1/2)T} r\left(t\right)\cos\omega_c t dt-\int_{(k+1/2)T}^{(k+1)T} r\left(t\right)\cos\omega_c t dt\right)^2$$

$$+\sum_{k=0}^{K-1}\left(\int_{kT}^{(k+1/2)T} r\left(t\right)\sin\omega_c t dt-\int_{(k+1/2)T}^{(k+1)T} r\left(t\right)\sin\omega_c t dt\right)^2$$

or

$$\sum_{k=0}^{K-1}\int_{kT}^{(k+1/2)T} r\left(t\right)\cos\omega_c t dt\int_{(k+1/2)T}^{(k+1)T} r\left(\tau\right)\cos\omega_c \tau d\tau$$

$$+\sum_{k=0}^{K-1}\int_{kT}^{(k+1/2)T} r\left(t\right)\sin\omega_c t dt\int_{(k+1/2)T}^{(k+1)T} r\left(\tau\right)\sin\omega_c \tau d\tau < 0$$

$$=\mathrm{Re}\left\{\sum_{k=0}^{K-1}\int_{kT}^{(k+1/2)T} r\left(t\right)e^{j\omega_c t}dt\int_{(k+1/2)T}^{(k+1)T} r\left(\tau\right)e^{-j\omega_c t}d\tau\right\}<0 \quad (5\text{-}58)$$

which is the analogous result to Eq. (5-11) for the coherent case.

For high SNR, even after applying the approximations to the nonlinearities given in Eq. (5-9), it is still difficult to average over the random carrier phase. Instead, we take note of the resemblance between Eqs. (5-58) and (5-59) for the low SNR case and propose an ad hoc complex equivalent to Eq. (5-12) for the noncoherent high SNR case, namely,

$$\sum_{k=0}^{K-1}\left|\int_{kT}^{(k+1/2)T} r\left(t\right)e^{j\omega_c t}dt+\int_{(k+1/2)T}^{(k+1)T} r\left(t\right)e^{j\omega_c t}dt\right| <$$

$$\sum_{k=0}^{K-1}\left|\int_{kT}^{(k+1/2)T} r\left(t\right)e^{j\omega_c t}dt-\int_{(k+1/2)T}^{(k+1)T} r\left(t\right)e^{j\omega_c t}dt\right| \quad (5\text{-}59)$$

Figure 5-4 is a block diagram of the implementation of the low and high SNR classifiers defined by Eqs. (5-58) and (5-59).

5.3.2 Probability of Misclassification for Noncoherent BPSK

To compute the probability of misclassification, we note that Eq. (5-58) is still made up of a sum of products of mutually independent real Gaussian RVs and thus can still be written in the form of Eq. (5-14) with twice as many terms, i.e.,

$$D = \frac{1}{2}\sum_{k=0}^{K-1}(X_k Y_k^* + X_k^* Y_k) \tag{5-60}$$

where now the complex Gaussian RVs are defined as $X_k = X_{ck} + jX_{sk}$, $Y_k = Y_{ck} + jY_{sk}$. The means of the terms are given by

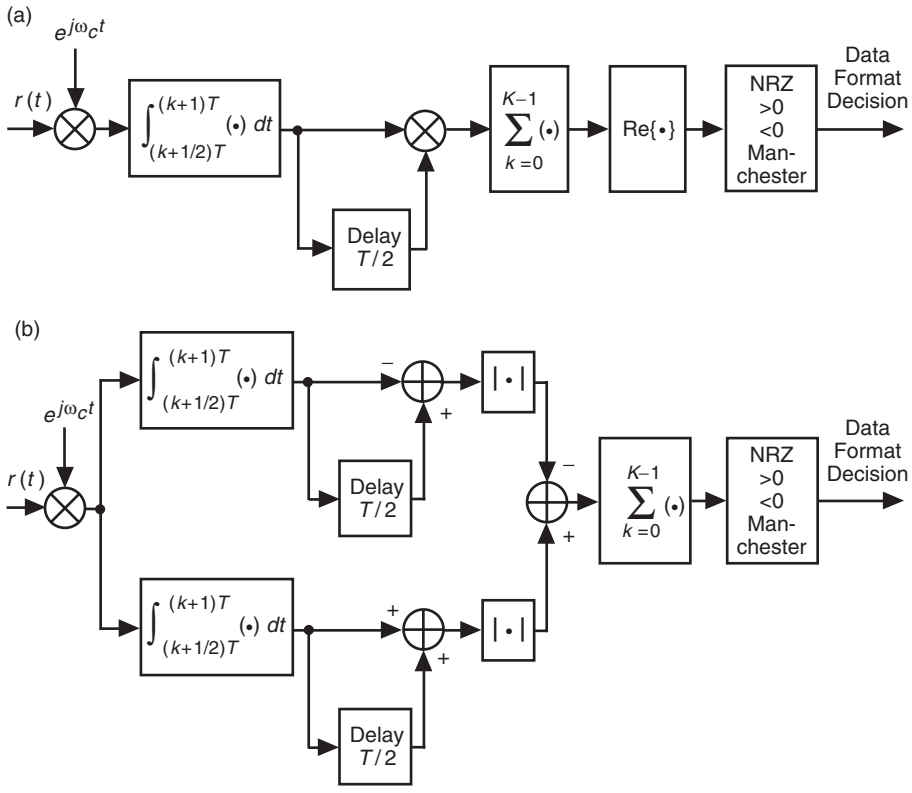


Fig. 5-4. Reduced-complexity noncoherent data format classifiers for BPSK modulation: (a) low SNR and (b) high SNR.

$$\bar{X}_k = \bar{Y}_k = c_k \left(\cos\theta_c - j \sin\theta_c\right) \sqrt{P/8}T \tag{5-61}$$

whereas the variances and cross-correlations are the same as in Eq. (5-15). Thus, since the magnitude of the means in Eq. (5-61) is reduced by a factor of $\sqrt{2}$ relative to that of the means in Eq. (5-15), we conclude that the probability of misclassification is obtained from Eq. (5-23) by replacing E_s/N_0 with $E_s/(2N_0)$ and K with $2K$, resulting in

$$P_M = \frac{1}{2^{2K-1}} \sum_{k=1}^{K} \binom{2K-1}{K-k} \sum_{n=0}^{k-1} \exp\left(-\frac{KE_s}{2N_0}\right) \frac{(KE_s/2N_0)^n}{n!} \tag{5-62}$$

Furthermore, the asymptotic behavior of Eq. (5-62) for large K can be determined from Eq. (5-31) by making the same replacements as above, resulting in

$$P_M = Q\left(\sqrt{\frac{K\left(E_s/N_0\right)^2}{2 + 2E_s/N_0}}\right) \tag{5-63}$$

which for sufficiently large E_s/N_0 approaches Eq. (5-31) for the coherent case.

Figure 5-5 illustrates numerical results for the misclassification probability obtained by computer simulation for the low SNR and high SNR reduced-complexity data format classifiers as specified by Eqs. (5-58) and (5-59), respectively, as well as the optimum classifier described by Eq. (5-46). Also illustrated are the numerical results obtained from the closed-form analytical solution given in Eq. (5-62) for the low SNR reduced-complexity scheme (which are in exact agreement with the simulation results) and the asymptotic results obtained from Eq. (5-63). As in the coherent case, the difference in performance between the low and high SNR reduced-complexity classifiers is again quite small over a large range of SNRs. Furthermore, we see here again that the performances of the approximate but simpler classification algorithms are in close proximity to that of the optimum one. Finally, comparison between the corresponding coherent and noncoherent classifiers is illustrated in Fig. 5-6 and reveals a penalty of approximately 1 dB or less depending on the SNR.

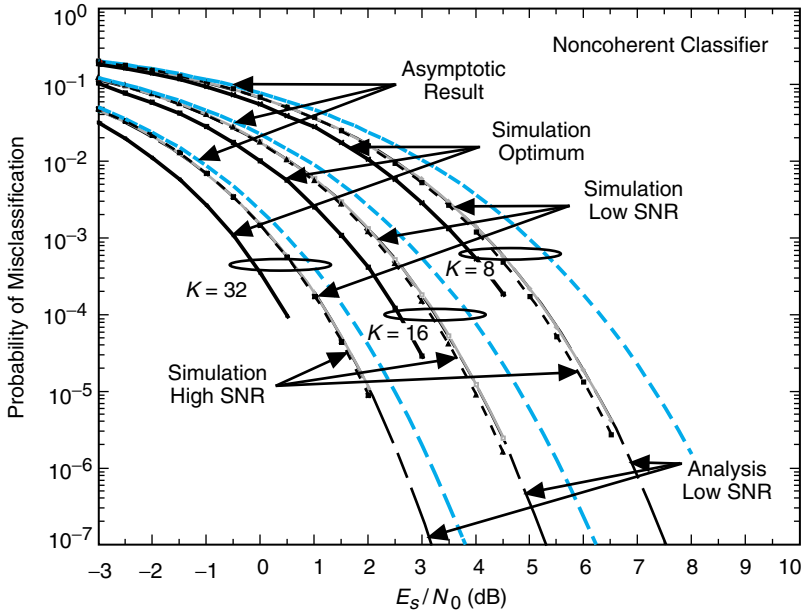


Fig. 5-5. A comparison of the performance of noncoherent data format classifiers for BPSK modulation.

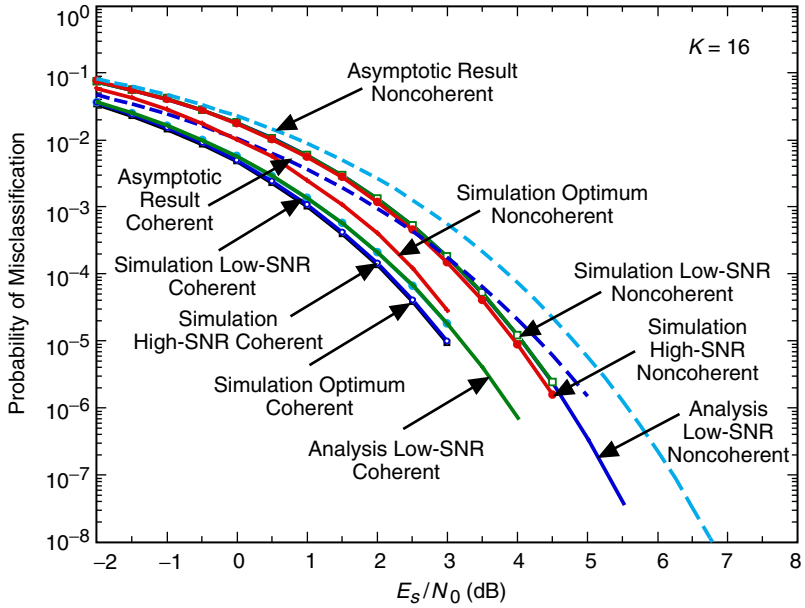


Fig. 5-6. A comparison of the performance of coherent and noncoherent data format classifiers for BPSK modulation: suppressed-carrier case.

5.4 Maximum-Likelihood Noncoherent Classifier of Data Format for QPSK

Following the same approach as in Section 5.2.1, the LLF for the noncoherent QPSK case is easily shown to be

$$\Lambda \triangleq \ln p\big(r\left(t\right)|p\left(t\right)\big) \tag{5-64}$$

$$= \underset{\theta_c}{E}\left\{\exp\left(\sum_{k=0}^{K-1}\left[\ln\cosh\left(\frac{2\sqrt{P}}{N_0}\int_{kT}^{(k+1)T} r\left(t\right)p\left(t-kT\right)\cos\left(\omega_c t+\theta_c\right)dt\right)\right.\right.$$

$$\left.\left.+\ln\cosh\left(\frac{2\sqrt{P}}{N_0}\int_{kT}^{(k+1)T} r\left(t\right)p\left(t-kT\right)\sin\left(\omega_c t+\theta_c\right)dt\right)\right]\right)\right\} \tag{5-65}$$

Making the same small argument approximations to the nonlinearities, we get

$$p\big(r(t)|p(t)\big) = \underset{\theta_c}{E}\left\{\exp\left[\frac{1}{2}\sum_{k=0}^{K-1}\left(\frac{2\sqrt{P}}{N_0}\int_{kT}^{(k+1)T} r\left(t\right)p\left(t-kT\right)\cos\left(\omega_c t+\theta_c\right)dt\right)^2\right.\right.$$

$$\left.\left.+\frac{1}{2}\sum_{k=0}^{K-1}\left(\frac{2\sqrt{P}}{N_0}\int_{kT}^{(k+1)T} r\left(t\right)p\left(t-kT\right)\sin\left(\omega_c t+\theta_c\right)dt\right)^2\right]\right\}$$

$$= \underset{\theta_c}{E}\left\{\exp\left[\frac{2P}{N_0^2}\sum_{k=0}^{K-1}\left(r_{ck}\cos\theta_c - r_{sk}\sin\theta_c\right)^2\right.\right.$$

$$\left.\left.+\ \frac{2P}{N_0^2}\sum_{k=0}^{K-1}\left(r_{ck}\sin\theta_c + r_{sk}\cos\theta_c\right)^2\right]\right\}$$

$$= \underset{\theta_c}{E}\left\{\exp\left[\frac{2P}{N_0^2}\sum_{k=0}^{K-1}\left(r_{ck}^2+r_{sk}^2\right)\cos^2\left(\theta_c+\eta_k\right)\right.\right.$$

$$\left.\left.+\ \frac{2P}{N_0^2}\sum_{k=0}^{K-1}\left(r_{ck}^2+r_{sk}^2\right)\sin^2\left(\theta_c+\eta_k\right)\right]\right\}$$

$$= \exp\left[\frac{2P}{N_0^2}\sum_{k=0}^{K-1}\left(r_{ck}^2+r_{sk}^2\right)\right] \tag{5-66}$$

Comparing Eq. (5-66) with Eq. (5-50), we note the absence of the Bessel function factor. However, in arriving at Eq. (5-56), which was based on retaining only linear terms, we approximated this factor by unity. Thus, applying the same small argument approximation of the exponential as before, we again arrive at a classification based on Eq. (5-56). Finally then, as in the coherent case, we conclude that the performance of the noncoherent classifier of data format for QPSK is identical to that for BPSK.

5.5 Maximum-Likelihood Coherent Classifier of Data Format for BPSK with Residual and Suppressed Carriers

When NRZ is transmitted, the received signal takes the form of Eq. (5-1) with $p(t) = p_1(t)$ and $P = P_t$, where P_t now denotes the total transmitted power. On the other hand, when Manchester-coded data are transmitted, the received signal has the form

$$r(t) = \sqrt{2P_c}\sin\omega_c t + \sqrt{2P_d}\left(\sum_{n=-\infty}^{\infty} c_n p_2(t-nT)\right)\cos\omega_c t + n(t) \tag{5-67}$$

where $P_c = P_t\cos^2\beta$ and $P_d = P_t\sin^2\beta$ are, respectively, the powers allocated to the discrete and data-modulated carriers with β the phase modulation index. Then, analogous to Eq. (5-2), it is straightforward to show that

$$p\big(r(t)|\{c_n\}, p_2(t)\big) = C\prod_{k=0}^{K-1}\exp\left(\frac{2\sqrt{2P_c}}{N_0}\int_{kT}^{(k+1)T} r(t)\sin\omega_c t dt\right) \times \exp\left(\frac{2\sqrt{2P_d}}{N_0}c_k\int_{kT}^{(k+1)T} r(t)\,p_2(t-kT)\cos\omega_c t dt\right) \tag{5-68}$$

Averaging over the iid data sequence and taking the logarithm gives

$$\ln p\big(r(t)\,|p(t)\big) = \sum_{k=0}^{K-1}\frac{2\sqrt{2P_c}}{N_0}\int_{kT}^{(k+1)T} r(t)\sin\omega_c t dt + \sum_{k=0}^{K-1}\ln\cosh\left(\frac{2\sqrt{2P_d}}{N_0}\int_{kT}^{(k+1)T} r(t)\,p_2(t-kT)\cos\omega_c t dt\right) \tag{5-69}$$

Finally then, we obtain the classification rule:

Choose Manchester coding if

$$\sum_{k=0}^{K-1} \ln\cosh\left(\frac{2\sqrt{2P_t}}{N_0}\int_{kT}^{(k+1)T} r\left(t\right)p_1\left(t-kT\right)\cos\omega_c t dt\right) <$$

$$\sum_{k=0}^{K-1}\frac{2\sqrt{2P_c}}{N_0}\int_{kT}^{(k+1)T} r\left(t\right)\sin\omega_c t dt$$

$$+\sum_{k=0}^{K-1} \ln\cosh\left(\frac{2\sqrt{2P_d}}{N_0}\int_{kT}^{(k+1)T} r\left(t\right)p_2\left(t-kT\right)\cos\omega_c t dt\right) \quad (5\text{-}70)$$

Otherwise, choose NRZ.

To obtain the reduced-complexity version of Eq. (5-70), we once again use the nonlinearity approximations in Eq. (5-9). For the low SNR case, we get after some manipulation

$$D \triangleq \sum_{k=0}^{K-1}\left[\frac{2(P_t-P_d)}{N_0^2}\left(X_{ck}^2+Y_{ck}^2\right)+\frac{4(P_t+P_d)}{N_0^2}X_{ck}Y_{ck}-\frac{\sqrt{2P_c}}{N_0}\left(X_{sk}+Y_{sk}\right)\right]$$

$$< 0 \quad (5\text{-}71)$$

where for convenience of notation as before we have defined

$$\begin{aligned} X_{ck} &= \int_{kT_b}^{(k+1/2)T} r\left(t\right)\cos\omega_c t dt,\; Y_{ck} = \int_{(k+1/2)T}^{(k+1)T} r\left(\tau\right)\cos\omega_c t d\tau \\ X_{sk} &= \int_{kT}^{(k+1/2)T} r\left(t\right)\sin\omega_c t dt,\; Y_{sk} = \int_{(k+1/2)T}^{(k+1)T} r\left(\tau\right)\sin\omega_c t d\tau \end{aligned} \quad (5\text{-}72)$$

$k = 0, 1, \cdots, K-1$. Alternatively, in terms of the modulation index, SNR, and normalized observables

$$\begin{aligned} X'_{ck} &\triangleq \frac{X_{ck}}{T\sqrt{2P_t}},\ Y'_{ck} \triangleq \frac{Y_{ck}}{T\sqrt{2P_t}} \\ X'_{sk} &\triangleq \frac{X_{sk}}{T\sqrt{2P_t}},\ Y'_{sk} \triangleq \frac{Y_{sk}}{T\sqrt{2P_t}} \end{aligned} \tag{5-73}$$

Eq. (5-71) becomes

$$\begin{aligned} D \triangleq \sum_{k=0}^{K-1} \Bigg[& 2\frac{E_t}{N_0}\cos^2\beta\left(X_{ck}^{'2} + Y_{ck}^{'2}\right) + 4\frac{E_t}{N_0}\left(1+\sin^2\beta\right)X'_{ck}Y'_{ck} \\ & - (\cos\beta)\left(X'_{sk} + Y'_{sk}\right)\Bigg] < 0 \end{aligned} \tag{5-74}$$

where $E_t/N_0 \triangleq P_tT/N_0$. Although the first two terms in the summation satisfy the type of quadratic form considered in [1, Appendix B], unfortunately, the last term, which does not contain second-order Gaussian RVs, prevents analytically evaluating the misclassification probability in the same manner that was done previously in Section 5.1.3. Nevertheless it is still possible to analytically evaluate the asymptotic (large K) performance in the same manner as before. Here, however, because of the lack of symmetry of the two hypotheses, one must individually evaluate the two misclassification probabilities (probability of choosing Manchester when NRZ is transmitted and vice versa) and then average the resulting expressions.

Considering first the case where NRZ data are transmitted, i.e., the received signal takes the form of Eq. (5-1), then after considerable manipulation, it can be shown that

$$\begin{aligned} \bar{D} &= \frac{K}{4}\left(\cos^2\beta + \frac{2E_t}{N_0}\right) \\ \sigma_D^2 &= \frac{K}{8}\frac{N_0}{E_t}\left[\cos^2\beta + \frac{E_t}{N_0}\left(1+\sin^4\beta\right) + 4\left(\frac{E_t}{N_0}\right)^2\right] \end{aligned} \tag{5-75}$$

Thus, making the same Gaussian assumption on D, the probability of misclassification for this case is given by

$$\begin{aligned} P_{M1} = \Pr\{D<0\} &= Q\left(\frac{\bar{D}}{\sigma_D}\right) \\ &= Q\left(\sqrt{\frac{K\frac{E_t}{N_0}\left(\cos^2\beta + \frac{2E_t}{N_0}\right)^2}{2\left[\cos^2\beta + \frac{E_t}{N_0}\left(1+\sin^4\beta\right) + 4\left(\frac{E_t}{N_0}\right)^2\right]}}\right) \end{aligned} \tag{5-76}$$

For the case where Manchester-coded data are transmitted, i.e., the received signal takes the form of Eq. (5-67), then again, after considerable manipulation, it can be shown that

$$\begin{aligned} \bar{D} &= -\frac{K}{4}\left(\cos^2\beta + \frac{2E_t}{N_0}\sin^4\beta\right) \\ \sigma_D^2 &= \frac{K}{8}\frac{N_0}{E_t}\left[\cos^2\beta + 2\frac{E_t}{N_0}\sin^2\beta + 4\left(\frac{E_t}{N_0}\right)^2\sin^6\beta\right] \end{aligned} \tag{5-77}$$

whereupon the probability of misclassification becomes

$$\begin{aligned} P_{M2} &= \Pr\{D > 0\} = Q\left(-\frac{\bar{D}}{\sigma_D}\right) \\ &= Q\left(\sqrt{\frac{K\frac{E_t}{N_0}\left(\cos^2\beta + \frac{2E_t}{N_0}\sin^4\beta\right)^2}{2\left[\cos^2\beta + 2\frac{E_t}{N_0}\sin^2\beta + 4\left(\frac{E_t}{N_0}\right)^2\sin^6\beta\right]}}\right) \end{aligned} \tag{5-78}$$

Finally, assuming the equiprobable data format hypothesis, the asymptotic average probability of misclassification is the average of Eq. (5-76) and Eq. (5-78), namely,

$$P_M = \frac{1}{2}\left(P_{M1} + P_{M2}\right) \tag{5-79}$$

Note that, for $\beta = 90$ deg, $E_t = E_s$ and Eq. (5-79) reduces to Eq. (5-31) as it should.

For high SNR, using the approximation

$$\ln\cosh x \cong |x| - \ln 2 \tag{5-80}$$

we obtain, analogous to Eq. (5-71),

$$D \triangleq \sum_{k=0}^{K-1}\left[\sqrt{P_t}\,|X_{ck} + Y_{ck}| - \sqrt{P_c}\,(X_{sk} + Y_{sk}) - \sqrt{P_d}\,|X_{ck} - Y_{ck}|\right] < 0 \tag{5-81}$$

or in terms of the modulation index and the normalized observables,

$$D \triangleq \sum_{k=0}^{K-1} \left[|X'_{ck} + Y'_{ck}| - (X'_{sk} + Y'_{sk}) \cos\beta - |X'_{ck} - Y'_{ck}| \sin\beta \right] < 0 \qquad (5\text{-}82)$$

Figure 5-7 is an illustration of the average (over the two hypotheses) misclassification probability for the various coherent classification algorithms, where the results are all obtained by computer simulation. We observe that, over a very wide range of SNRs, the performance of the high SNR approximation classifier is virtually a perfect match to that of the optimum classifier, but its implementation is somewhat simpler. On the other hand, while the performance of the low SNR classifier converges to that of the optimum classifier at low SNR as it should, at high SNR it results in considerable degradation. The reasoning behind this relative difference in behavior between the approximate and optimum classifiers can be explained as follows: Whereas at low SNR the maximum difference between $\ln\cosh x$ and its high SNR approximation $|x| - \ln 2$ occurs at $x = 0$ and is equal to $\ln 2$, at high SNR the difference between $\ln\cosh x$ and its low SNR approximation $x^2/2$ grows without bound, i.e., the difference between a linear and a square law behavior. Thus, using the high SNR approximation of $\ln\cosh x$ over the entire range of SNR is a much better fit than using the low SNR approximation over the same SNR range. Illustrated in Fig. 5-8 is a comparison of the performances of the coherent classifiers for the residual- and suppressed-carrier cases, the latter being obtained from the discussion in Section 5.1.1. We observe that for the optimum and high SNR approximation classifiers the two are quite similar in performance although the suppressed-carrier one is a bit inferior. This implies that a discrete carrier component is slightly influential in improving data format classification for coherent communications.

5.6 Maximum-Likelihood Noncoherent Classifier of Data Format for BPSK with Residual and Suppressed Carriers

As in Section 5.3.1, we again assume that the carrier has a random phase, θ_c, that is unknown and uniformly distributed. Then when NRZ is transmitted, the received signal takes the form of Eq. (5-40) with $p(t) = p_1(t)$ and $P = P_t$. On the other hand, when Manchester-coded data are transmitted, the received signal has the form

$$r(t) = \sqrt{2P_c}\sin(\omega_c t + \theta_c) + \sqrt{2P_d}\left(\sum_{n=-\infty}^{\infty} c_n p_2(t - nT)\right)\cos(\omega_c t + \theta_c) + n(t) \qquad (5\text{-}83)$$

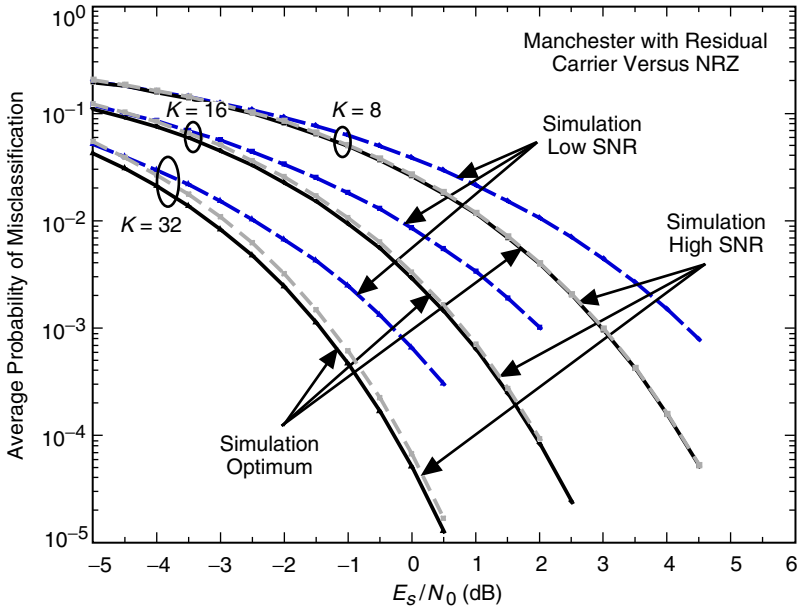


Fig. 5-7. Misclassification probability for residual carrier coherent classifier: β = 60 deg.

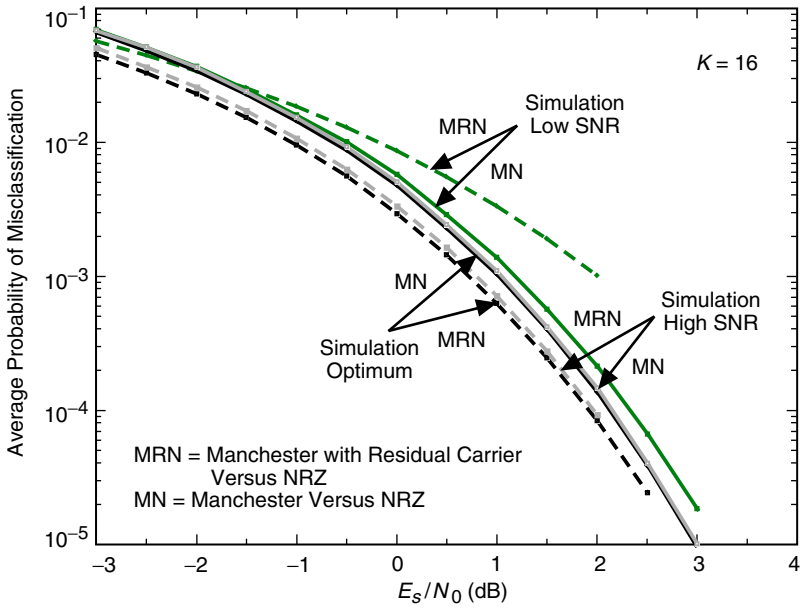


Fig. 5-8. Comparison of misclassification probability for suppressed and residual carrier coherent classifiers.

Without going to great length, following the same procedure as in Section 5.3.1, it is straightforward to show that, analogous to Eq. (5-46), the likelihood ratio for choosing between NRZ and residual-carrier Manchester-coded data is given by

$$LR = \frac{\underset{\theta_c}{E}\left\{ \exp\left[\sum_{k=0}^{K-1} \ln\cosh\left(\frac{2\sqrt{2P_t}}{N_0} \int_{kT}^{(k+1)T} r(t) p_1(t-kT)\cos(\omega_c t+\theta_c)dt \right) \right] \right\}}{\underset{\theta_c}{E}\left\{ \exp\left[\sum_{k=0}^{K-1} \ln\cosh\left(\frac{2\sqrt{2P_d}}{N_0} \int_{kT}^{(k+1)T} r(t) p_2(t-kT)\cos(\omega_c t+\theta_c)dt \right) \right.\right.}$$

$$\left.\left. + \frac{2\sqrt{2P_c}}{N_0} \sum_{k=0}^{K-1} \int_{kT}^{(k+1)T} r(t)\sin(\omega_c t+\theta_c)\, dt \right] \right\} \tag{5-84}$$

To obtain a low SNR classifier, we approximate the nonlinearities in Eq. (5-84) by their small argument values which results after considerable simplification in a test analogous to Eq. (5-55) given by the following: Choose Manchester if

$$\sum_{k=0}^{K-1} \left|\tilde{r}'_k(1)\right|^2 < \left(\sin^2\beta\right) \sum_{k=0}^{K-1} \left|\tilde{r}'_k(2)\right|^2 + \left(\cos^2\beta\right) \left| \sum_{k=0}^{K-1} \tilde{r}'_k(1) \right|^2 \tag{5-85}$$

where as before the real and imaginary components of $\tilde{r}_k(l) = \left(T_b\sqrt{2P_t}\right)\tilde{r}'_k(l)$; $l = 1, 2$ are defined in Eq. (5-35). Alternatively, in terms of integrals, Eq. (5-85) becomes

$$\sum_{k=0}^{K-1} \left| \int_{kT}^{(k+1/2)T} r(t)\, e^{j\omega_c t} dt + \int_{(k+1/2)T}^{(k+1)T} r(t)\, e^{j\omega_c t} dt \right|^2 <$$

$$\left(\sin^2\beta\right) \sum_{k=0}^{K-1} \left| \int_{kT}^{(k+1/2)T} r(t)\, e^{j\omega_c t} dt - \int_{(k+1/2)T}^{(k+1)T} r(t)\, e^{j\omega_c t} dt \right|^2$$

$$+ \left(\cos^2\beta\right) \left| \sum_{k=0}^{K-1} \int_{kT}^{(k+1/2)T} r(t)\, e^{j\omega_c t} dt + \int_{(k+1/2)T}^{(k+1)T} r(t)\, e^{j\omega_c t} dt \right|^2 \tag{5-86}$$

For the high SNR case, by analogy with Eq. (5-86), we propose the ad hoc test

$$\sum_{k=0}^{K-1}\left|\int_{kT}^{(k+1/2)T} r(t)\,e^{j\omega_c t}dt+\int_{(k+1/2)T}^{(k+1)T} r(t)\,e^{j\omega_c t}dt\right|<$$

$$(\sin\beta)\sum_{k=0}^{K-1}\left|\int_{kT}^{(k+1/2)T} r(t)\,e^{j\omega_c t}dt-\int_{(k+1/2)T}^{(k+1)T} r(t)\,e^{j\omega_c t}dt\right|$$

$$+(\cos\beta)\left|\sum_{k=0}^{K-1}\int_{kT}^{(k+1/2)T} r(t)\,e^{j\omega_c t}dt+\int_{(k+1/2)T}^{(k+1)T} r(t)\,e^{j\omega_c t}dt\right| \qquad (5\text{-}87)$$

which is consistent with the ad hoc test in Eq. (5-59) when $\beta = 90$ deg.

Analogous to Fig. 5-7, Fig. 5-9 is an illustration of the average misclassification probability for the various classification noncoherent algorithms, where the results are all obtained by computer simulation. We again observe that, over a very wide range of SNRs, the performance of the high SNR approximation

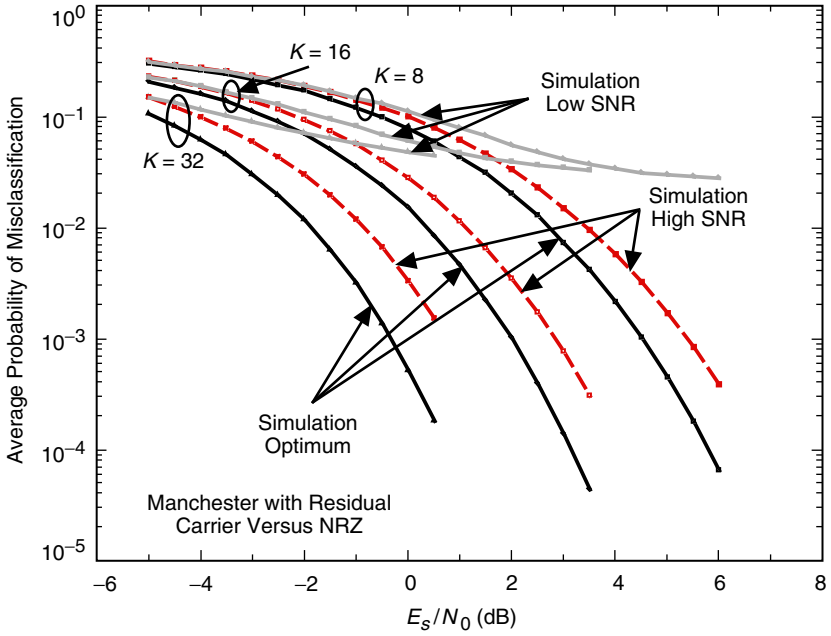


Fig. 5-9. Misclassification probability for residual carrier noncoherent classifier: β = 60 deg.

classifier is virtually a perfect match to that of the optimum classifier, but its implementation is somewhat simpler. On the other hand, while the performance of the low SNR classifier converges to that of the optimum classifier at low SNR as it should, at high SNR it results in considerable degradation. Illustrated in Fig. 5-10 is a comparison of the performances of the coherent classifiers for the residual- and suppressed-carrier noncoherent classifier cases, the latter being obtained from the discussion in Section 5.3.1 of the chapter. We observe that, as in the coherent comparison illustrated in Fig. 5-8, for the optimum and high SNR approximation classifiers the two are again quite similar in performance, although now the suppressed-carrier one is a bit superior. Finally, analogous to Fig. 5-6, a comparison of the corresponding coherent and noncoherent classifiers for the residual-carrier case is illustrated in Fig. 5-11, and for the optimum metric reveals a penalty of approximately 1.25 dB or less depending on the SNR.

5.7 Maximum-Likelihood Pulse Shape Classification

The solution to the problem of making an ML decision on the pulse shape of a modulation from a variety of different possibilities in principle follows the identical procedure discussed in the previous sections for data format classification, except for the fact that we no longer restrict ourselves to a digital pulse waveform. For example, suppose that we are transmitting a binary modulation, where the pulse shape is known to be one of L possibilities, namely, $p_l(t),\ l = 1, 2, \cdots, L$. Then, using the LLF of Eq. (5-4) and defining $r_k(l)$ as in Eq. (5-7) (without the special cases of NRZ and Manchester), then the ML coherent classifier of pulse shape would be to choose $p_{l*}(t)$ corresponding to

$$l* = \underset{l}{\operatorname{argmax}} \sum_{k=0}^{K-1} \ln\cosh\left(\frac{2\sqrt{2P}}{N_0}\int_{kT}^{(k+1)T} r(t)\, p_l(t - kT)\cos\omega_c t dt\right) \quad (5\text{-}88)$$

where the notation $\underset{l}{\operatorname{argmax}}\, f(l)$ denotes the value of l that maximizes the function $f(l)$. Once again by making small and large argument approximations of the $\ln\cosh(\cdot)$ nonlinearity, one can obtain reduced-complexity classifiers in the same manner as was used in Section 5.1.2 for the special case of data format classification. Other examples involving higher-order modulations follow along the same lines as those just discussed.

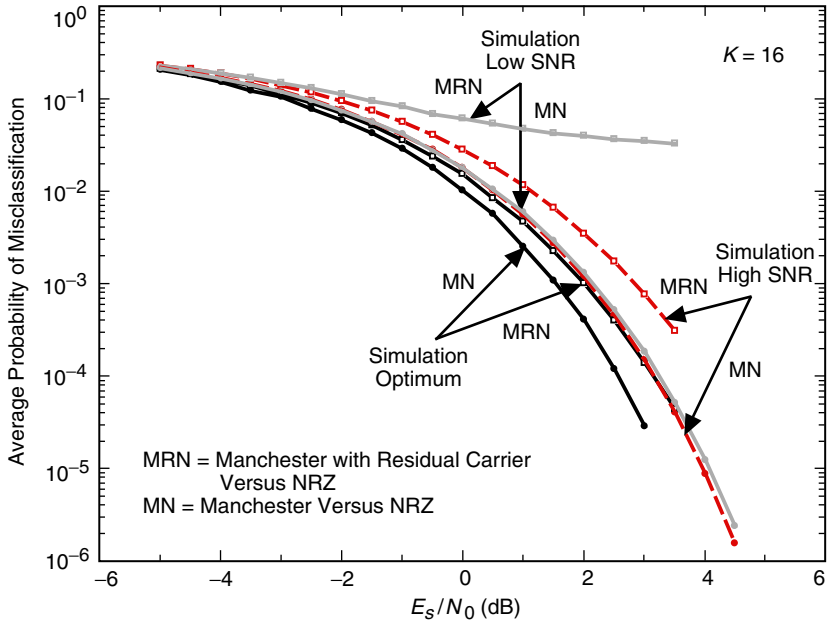


Fig. 5-10. Comparison of misclassification probability for suppressed and residual carrier noncoherent classifiers.

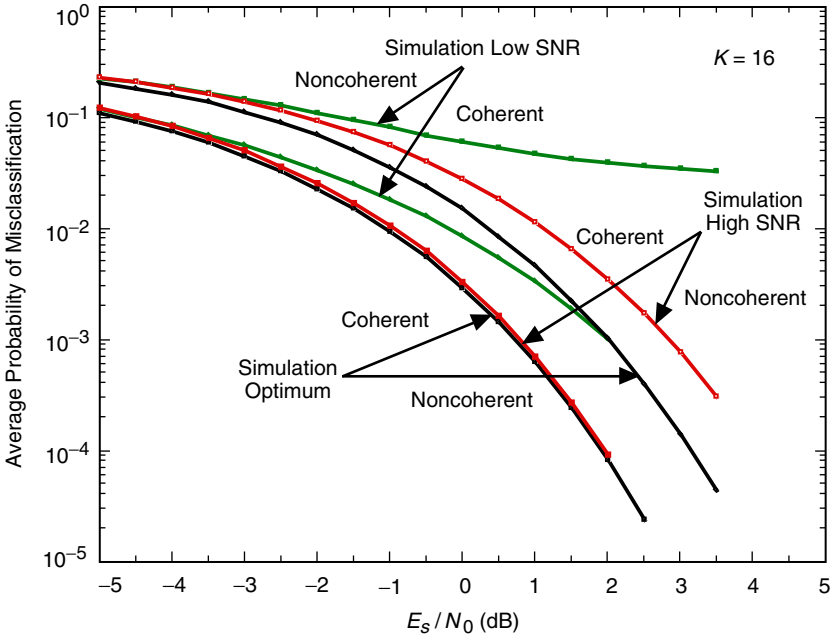


Fig. 5-11. A comparison of performance of coherent and noncoherent data format classifiers for BPSK modulation: residual carrier case.

References

[1] J. G. Proakis, *Digital Communications*, fourth ed., New York: McGraw Hill, Inc., 2001.

[2] M. K. Simon and M.-S. Alouini, *Digital Communication over Fading Channels*, second ed., New York: John Wiley & Sons, 2005.

Chapter 6
Signal-to-Noise Ratio Estimation

Marvin K. Simon and Samuel Dolinar

Of the many measures that characterize the performance of a communication receiver, signal-to-noise ratio (SNR) is perhaps the most fundamental in that many of the other measures directly depend on its knowledge for their evaluation. In the design of receivers for autonomous operation, it is desirable that the estimation of SNR take place with as little known information as possible regarding other system parameters such as carrier phase and frequency, order of the modulation, data symbol stream, data format, etc. While the maximum-likelihood (ML) approach to the problem will result in the highest quality estimator, as is typically the case with this approach, it results in a structure that is quite complex unless the receiver is provided with some knowledge of the data symbols typically obtained from data estimates made at the receiver (which themselves depend on knowledge of the SNR). SNR estimators of this type have been referred to in the literature as *in-service* estimators, and the evaluation of their performance has been considered in [1]. Since our interest here is in SNR estimation for autonomous operation, the focus of our attention will be on estimators that perform their function without any data symbol knowledge and, despite their ad hoc nature, maintain a high level of quality and robustness with respect to other system parameter variations.

One such ad hoc SNR estimator that has received considerable attention in the past is the so-called *split-symbol moments estimator (SSME)* [2–5] that forms its SNR estimation statistic from the sum and difference of information extracted from the first and second halves of each received data symbol. Implicit in this estimation approach, as is also the case for the in-service estimators, is that the data rate and symbol timing are known or can be estimated. (Later on in the chapter we shall discuss how the SNR estimation procedure can be modified

when symbol timing is unknown.) In the initial investigations, the performance of the SSME was investigated only for binary phase-shift keying (BPSK) modulations with and without carrier frequency uncertainty and as such was based on real sample values of the channel output. In fact, it was stated in [1, p. 1683], in reference to the SSME, that "none of these methods is easily extended to higher orders of modulations." More recently, it has been shown [6] that such is not the case. Specifically, the traditional SSME structure, when extended to the complex symbol domain, is readily applicable to the class of M-phase-shift keying (M-PSK) ($M \geq 2$) modulations, and furthermore its performance is independent of the value of M! Even more generally, the complex symbol version of the SSME structure can also be used to provide SNR estimation for two-dimensional signal sets such as quadrature amplitude modulation (QAM) although the focus of the chapter will be on the M-PSK application.

We begin the chapter by defining the signal model and formation of the SSME estimator. Following this, we develop exact as well as highly accurate approximate expressions for its mean and variance for a variety of different scenarios related to the degree of knowledge assumed for the carrier frequency uncertainty and to what extent it is compensated for in obtaining the SNR estimate. With regard to the observables from which the SNR estimate was formed, two different models will be considered. In one case, we consider the availability of a plurality of uniformly spaced independent[1] samples of the received signal in each half-symbol, whereas in the second case only one sample of information from each half-symbol, e.g., the output of half-symbol matched filters, is assumed available—hence, two samples per symbol. Furthermore, we consider the wideband case wherein the symbol pulse shape is assumed to be rectangular, and thus the matched filters are in fact integrate-and-dump (I&D) filters. Finally, we discuss in detail a method for reconfiguring the conventional SSME to improve its performance for SNRs above a particular critical value. The reconfiguration, initially disclosed in [7], consists of partitioning the symbol interval into a larger (but even) number of subdivisions than the two that characterize the conventional SSME where the optimum number of subdivisions depends on the SNR region in which the true SNR lies. It will also be shown that these SNR regions can be significantly widened with very little loss in performance. Most important is the fact that, with this reconfiguration, the SNR estimator tracks the Cramer–Rao bound (with a fixed separation from it) on the variance of the estimator over the entire range of SNR values.

---

[1] Clearly the independence assumption on the samples is dependent on the sampling rate in relation to the bandwidth of the signal.

6.1 Signal Model and Formation of the Estimator

6.1.1 Sampled Version

A block diagram of the SSME structure in complex baseband form is illustrated in Fig. 6-1. Corresponding to the kth transmitted M-PSK symbol $d_k = e^{j\phi_k}$ in the interval $(k-1)\,T \leq t \leq kT$, the lth complex baseband received sample is given by[2]

$$y_{lk} = \frac{m}{N_s} d_k e^{j(\omega l T_s + \phi)} + n_{lk}, l = 0, 1, \cdots, N_s - 1, k = 1, 2, \cdots, N \tag{6-1}$$

where ϕ and ω are the carrier phase and frequency uncertainties (offsets), N_s is the number of uniform samples per symbol and is assumed to be an even integer, $1/T_s$ is the sampling rate, N is the number of symbols in the observation, n_{lk} is a sample of a zero-mean additive white Gaussian noise (AWGN) process with variance σ^2/N_s in each (real and imaginary) part, and m reflects the signal amplitude. It is also convenient to denote the duration of a symbol by $T = N_s T_s$. Based on the above, the true symbol SNR is given by

$$R = \frac{m^2}{2\sigma^2} \tag{6-2}$$

The received samples of Eq. (6-1) are first accumulated separately over the first and second halves of the kth symbol interval, resulting in the sums

$$\begin{aligned} Y_{\alpha_k} &= \sum_{l=0}^{N_s/2-1} y_{lk} e^{-j\theta_{lk}} = \sum_{l=0}^{N_s/2-1} \left(\frac{m}{N_s} d_k e^{j([l/N_s]\delta + \phi)} + n_{lk} \right) e^{-j\theta_{lk}} \\ Y_{\beta_k} &= \sum_{l=N_s/2}^{N_s-1} y_{lk} e^{-j\theta_{lk}} = \sum_{l=N_s/2}^{N_s-1} \left(\frac{m}{N_s} d_k e^{j([l/N_s]\delta + \phi)} + n_{lk} \right) e^{-j\theta_{lk}} \end{aligned} \tag{6-3}$$

where $e^{-j\theta_{lk}}$ is a phase compensation that accounts for the possible adjustment of the lkth sample for phase variations across a given symbol due to the frequency offset and $\delta \triangleq \omega T$ is the normalized (to the symbol time) frequency offset. Next, the half-symbol sums in Eq. (6-3) are summed and differenced to produce

[2] For convenience, we assume that ϕ includes the accumulated phase due to the frequency offset up until the beginning of the kth symbol interval.

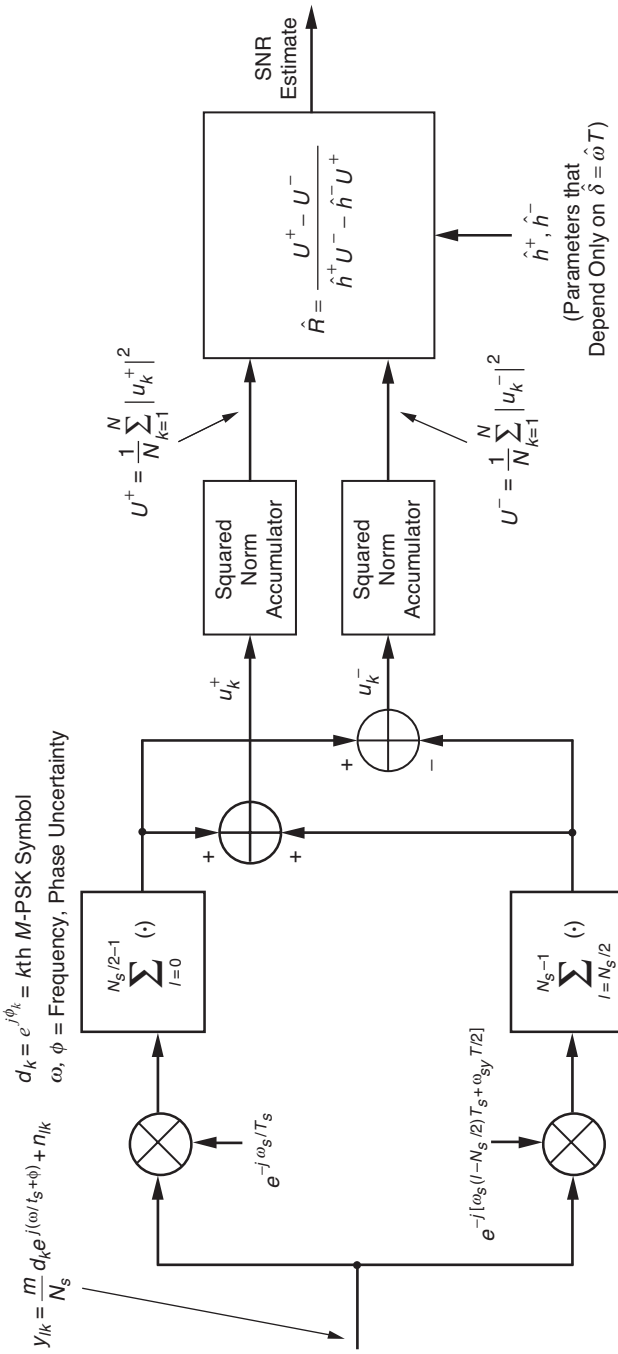


Fig. 6-1. Split-symbol SNR estimator for *M*-PSK modulation (sampled version).

$$u_k^{\pm} = Y_{\alpha_k} \pm Y_{\beta_k} \triangleq s_k^{\pm} + n_k^{\pm}, k = 1, 2, \cdots, N \tag{6-4}$$

where $s_k^{\pm}$ and $n_k^{\pm}$ respectively represent the signal and noise components of these half-symbol sums and differences and can be written in the form

$$\begin{aligned} s_k^{\pm} &= \frac{m}{N_s} e^{j(\phi+\phi_k)} \left[\sum_{l=0}^{N_s/2-1} e^{j([l/N_s]\delta-\theta_{lk})} \pm \sum_{l=N_s/2}^{N_s-1} e^{j([l/N_s]\delta-\theta_{lk})} \right] \\ n_k^{\pm} &= \sum_{l=0}^{N_s/2-1} n_{lk} e^{-j\theta_{lk}} \pm \sum_{l=N_s/2}^{N_s-1} n_{lk} e^{-j\theta_{lk}} \end{aligned} \tag{6-5}$$

Finally, we average the squared norms of the half-symbol sums and differences over the N-symbol duration of the observation, producing

$$U^{\pm} = \frac{1}{N} \sum_{k=1}^{N} \left| u_k^{\pm} \right|^2 \tag{6-6}$$

Note that U^+ is a statistical measure of signal-plus-noise power where U^- is a statistical measure of noise power. Also, depending on the amount of information available for the frequency uncertainty ω and the method by which it is compensated for (if at all), the SNR estimator will take on a variety of forms (to be discussed shortly), all of which, however, will depend on the received complex samples only via the averages U^+ and U^-.

Making the key observation that the observables U^+ and U^- are independent random variables (RVs) and denoting the normalized squared norm of their sum and difference signal components by

$$h^{\pm} \triangleq \frac{\left| s_k^{\pm} \right|^2}{m^2} \tag{6-7}$$

then it is straightforward to show that their means and variances are given by

$$\begin{aligned} E\left\{U^{\pm}\right\} &= 2\sigma^2 + \left| s_k^{\pm} \right|^2 = 2\sigma^2 \left(1 + h^{\pm} R\right) \\ \text{var}\left\{U^{\pm}\right\} &= \frac{4}{N}\sigma^2 \left(\left| s_k^{\pm} \right|^2 + \sigma^2 \right) = \frac{4}{N}\sigma^4 \left(1 + 2h^{\pm} R\right) \end{aligned} \tag{6-8}$$

Note that while the parameters $h^{\pm}$ depend on whether or not phase compensation is used and also on the frequency uncertainty, they are independent of the random carrier phase ϕ and the particular data symbol phase ϕ_k. As such, the $h^{\pm}$ are independent of the order M of the M-PSK modulation and, thus, so are the first and second moments of $U^{\pm}$ in Eq. (6-8).

Solving for the true SNR R from the first relation in Eq. (6-8) gives

$$R = \frac{E\{U^+\} - E\{U^-\}}{h^+ E\{U^-\} - h^- E\{U^+\}} \tag{6-9}$$

and the general form of the ad hoc SSME $\hat{R}$ is obtained by substituting the sample values $U^{\pm}$ for their expected values and the estimates $\hat{h}^{\pm}$ for their true values, namely,

$$\hat{R} = \frac{U^+ - U^-}{\hat{h}^+ U^- - \hat{h}^- U^+} \triangleq g\left(U^+, U^-\right) \tag{6-10}$$

For the case of real data symbols, i.e., BPSK, the estimator in Eq. (6-10) is exactly identical to the SSME considered in [2–5].

Note that in the absence of frequency uncertainty, i.e., $\delta = 0$, and thus of course no phase compensation, i.e., $\theta_{lk} = 0$, we have from Eq. (6-5) that $h^+ = 1$ and $h^- = 0$, in which case Eq. (6-9) simplifies to

$$R = \frac{E\{U^+\} - E\{U^-\}}{E\{U^-\}} \tag{6-11}$$

which appears reasonable in terms of the power interpretations of U^+ and U^- given above. Likewise, in this case we would have $\hat{h}^+ = 1$ and $\hat{h}^- = 0$, and the ad hoc SNR estimator would simplify to

$$\hat{R} = \frac{U^+ - U^-}{U^-} \tag{6-12}$$

6.1.2 I&D Version

A block diagram of the complex baseband SSME for this version is obtained from Fig. 6-1 by replacing the half-symbol accumulators by half-symbol I&Ds and is illustrated in Fig. 6-2. Corresponding to the kth transmitted M-PSK

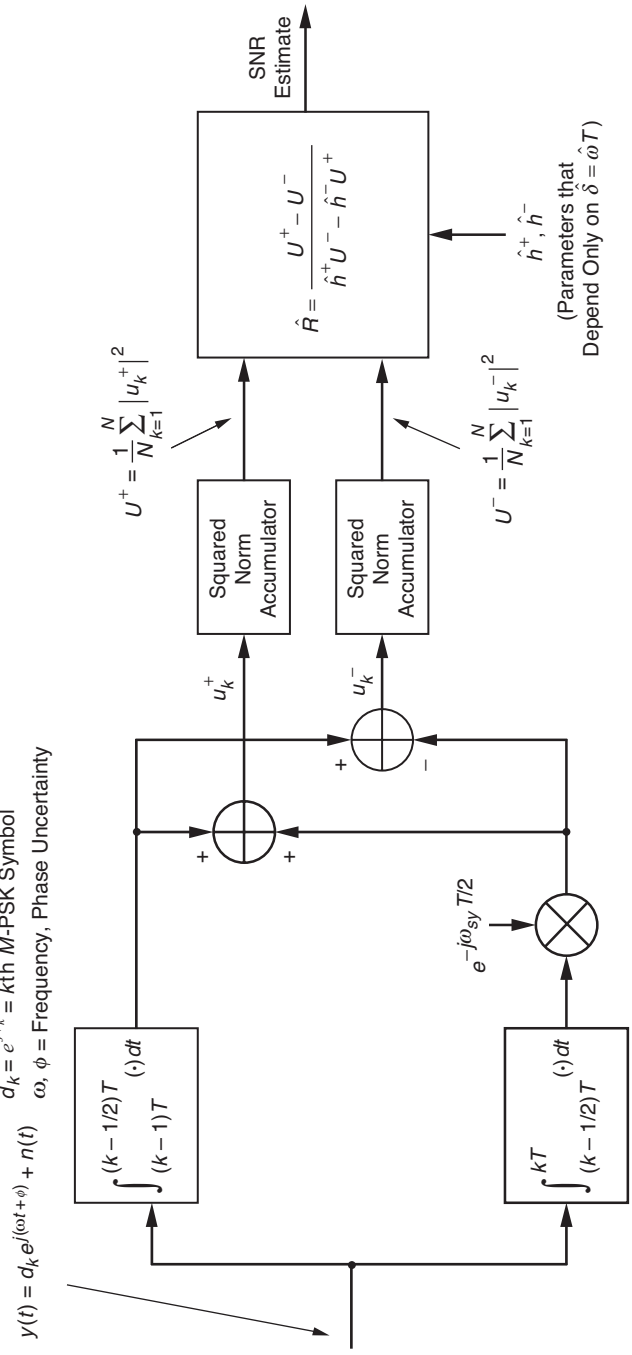


Fig. 6-2. Split-symbol SNR estimator for *M*-PSK modulation (I&D version).

symbol, the complex baseband received signal that is input to the first and second half-symbol I&Ds is given by

$$y(t) = md_k e^{j(\omega t+\phi)} + n(t), \quad (k-1)T \le t < kT \tag{6-13}$$

where $n(t)$ is the zero-mean AWGN process. The outputs of these same I&Ds are given by

$$\begin{aligned} Y_{\alpha k} &= md_k \frac{1}{T}\int_{(k-1)T}^{(k-1/2)T} e^{j(\omega t+\phi)}dt + \frac{1}{T}\int_{(k-1)T}^{(k-1/2)T} n(t)\,dt \\ &= (md_k/2)\, e^{j\phi} e^{j\omega(k-3/4)T}\mathrm{sinc}(\delta/4) + n_{\alpha k} \\ Y_{\beta k} &= \left(md_k \frac{1}{T}\int_{(k-1/2)T}^{kT} e^{j(\omega t+\phi)}dt + \frac{1}{T}\int_{(k-1/2)T}^{kT} n(t)\,dt \right) e^{-j\theta_k} \\ &= \left((md_k/2)\, e^{j\phi} e^{j\omega(k-3/4)T} e^{j\omega T/2}\mathrm{sinc}(\delta/4) + n_{\beta k} \right) e^{-j\theta_k} \end{aligned} \tag{6-14}$$

where $\mathrm{sinc}\, x \triangleq \sin x/x$, $n_{\alpha k}$, and $n_{\beta k}$ are complex Gaussian noise variables with zero mean and variance $\sigma^2/2$ for each real and imaginary component and $e^{-j\theta_k}$ is once again a phase compensation that accounts for the possible adjustment of the kth second-half sample for phase variations across a given symbol due to the frequency offset. As before, forming the half-symbol sums and differences produces

$$\begin{aligned} u_k^{\pm} = Y_{\alpha k} \pm Y_{\beta k} &= (md_k/2)\, e^{j\phi} e^{j\omega(k-3/4)T}\mathrm{sinc}(\delta/4)\left[1 \pm e^{j([\delta/2]-\theta_k)}\right] \\ &\quad + n_{\alpha k} \pm n_{\beta k} e^{-j\theta_k} \triangleq s_k^{\pm} + n_k^{\pm} \end{aligned} \tag{6-15}$$

If once again, as in Eq. (6-6), we average the squared norms of the half-symbol sums and differences over the N-symbol duration of the observation, then following the same series of steps as in Eqs. (6-7) through (6-9), we arrive at the ad hoc SNR estimator in Eq. (6-10).

6.2 Methods of Phase Compensation

For the sampled version of the SSME, we observe from Eq. (6-6) together with Eqs. (6-3) and (6-4) that the split-symbol observables $U^{\pm}$ are defined in terms of phase compensation factors $\left\{e^{-j\theta_{lk}}\right\}$ applied to the received samples $\{y_{lk}\}$ to compensate for phase variations across a given symbol due to the frequency offset ω. To perform this compensation, one requires some form of knowledge about this offset. In this regard, we shall assume that an estimate $\hat{\omega}$ of ω is externally provided. In principle, there are two ways in which this estimate can be used to provide the necessary compensation. The best-performing but most complex method adjusts the phases sample by sample, using a sample-by-sample compensation frequency $\omega_s = \hat{\omega}$. The alternative and less complex method does not compensate every sample but rather only once per symbol by adjusting the relative phase of the two half-symbols using a half-symbol compensation frequency $\omega_{sy} = \hat{\omega}$. Of course, the least complex form of phase compensation would be none at all even though the estimate $\hat{\omega}$ is available. In all three cases, the phase adjustment θ_{lk} can be written in the generic form

$$\theta_{lk} = \begin{cases} \omega_s l T_s, & 0 \le l \le N_s/2 - 1 \\ \omega_s \left(l - N_s/2\right) T_s + \omega_{sy} T/2, & N_s/2 \le l \le N_s - 1 \end{cases} \tag{6-16}$$

where

$$\begin{aligned} \omega_s = \omega_{sy} = \hat{\omega} &\text{ for sample-by-sample phase compensation} \\ \omega_s = 0, \omega_{sy} = \hat{\omega} &\text{ for half-symbol phase compensation} \\ \omega_s = \omega_{sy} = 0 &\text{ for no phase compensation} \end{aligned} \tag{6-17}$$

For the I&D version of the SSME, we only have the half-symbol phase compensation option available and thus $\theta_k = \omega_{sy} T/2 = \hat{\omega} T/2$. Of course, even though the estimate $\hat{\omega}$ is available, we again might still choose not to use it to compensate for the phase due to the frequency uncertainty. In this case, we would simply set $\theta_k = 0$ in Eqs. (6-14) and (6-15).

Besides being used for phase compensation of the samples or half-symbols that enter into the expressions for computing $U^{\pm}$, the frequency estimate also enters into play in determining the estimates $\hat{h}^{\pm}$ that are computed from $h^{\pm}$ by replacing ω with its estimate $\hat{\omega}$. Thus, the performance of the SSME will depend on the accuracy of the frequency estimate $\hat{\omega}$ with or without phase compensation. In the most general scenario, we shall consider a taxonomy of cases for analysis that, for the sampled version of the SSME, are illustrated by the tree diagram in Fig. 6-3. In this diagram, we start at the square node in the middle and proceed outward to any of the eight leaf nodes representing interesting com-

binations of $\omega, \hat{\omega}, \omega_{sy}$, and ω_s. The relative performance and complexity of each case is given qualitatively in Table 6-1, where the former is rated from worst (*) to best (****) and the latter from simplest (x) to most complex (xxxx). In the I&D version, a few of the tree branches of Fig. 6-3, namely, 2c and 3c, do not apply.

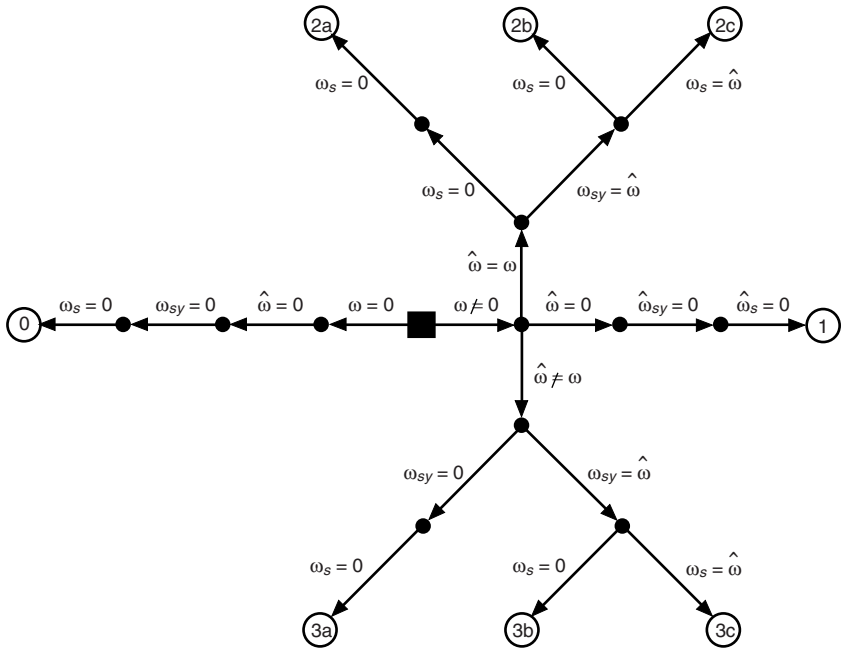


Fig. 6-3. A taxonomy of interesting cases for analysis.

Table 6-1. Qualitative relative performance and complexity of the various estimators.

Case number	Frequency offset	Frequency estimate	Phase compensation	Performance	Complexity
0	0	Perfect	None	****	x
1	≠0	0	None	*	x
2a	≠0	Perfect	None	**	xx
2b	≠0	Perfect	Half-symbol	***	xxx
2c	≠0	Perfect	Sample-by-sample	****	xxxx
3a	≠0	Imperfect	None	* to **	xx
3b	≠0	Imperfect	Half-symbol	* to ***	xxx
3c	≠0	Imperfect	Sample-by-sample	* to ****	xxxx

6.3 Evaluation of $h^{\pm}$

For the sampled version SSME, we first insert the expression for the phase compensation in Eq. (6-16) into Eq. (6-5), which after simplification becomes

$$s_k^{\pm} = \frac{m}{N_s} e^{j(\phi+\phi_k)} \left(\frac{1 - e^{j(\delta-\omega_s T)/2}}{1 - e^{j(\delta-\omega_s T)/N_s}} \right) \left(1 \pm e^{j(\delta-\omega_{sy} T)/2} \right) \tag{6-18}$$

Then taking the squared norm of Eq. (6-18) and normalizing by m^2 gives, in accordance with Eq. (6-7),

$$h^{\pm} = W_{N_s}(\delta_s) \frac{1 \pm W_0(\delta_{sy})}{2} \tag{6-19}$$

where

$$\begin{aligned} \delta_s &= \delta - \omega_s T \\ \delta_{sy} &= \delta - \omega_{sy} T \end{aligned} \tag{6-20}$$

and

$$\begin{aligned} W_0(\delta) &= \cos(\delta/2) \\ W_{N_s}(\delta) &= \frac{\operatorname{sinc}^2(\delta/4)}{\operatorname{sinc}^2(\delta/2N_s)} \end{aligned} \tag{6-21}$$

are windowing functions. Note that $W_0(\delta)$ has zeros at odd multiples of π, and $W_{N_s}(\delta)$ has zeros at all multiples of 4π except for multiples of $2N_s\pi$.

For the I&D version, $h^{\pm}$ is still given by Eq. (6-19) but with $W_{N_s}(\delta_s)$ replaced by

$$W(\delta) = \operatorname{sinc}^2(\delta/4) \tag{6-22}$$

which is tantamount to taking the limit of $W_{N_s}(\delta)$ as N_s approaches infinity.

6.4 Mean and Variance of the SNR Estimator

In this section, we evaluate the mean and variance of $\hat{R}$ for a variety of special cases related to (1) the absence or presence of carrier frequency uncertainty ω and likewise for its estimation, (2) whether or not its estimate $\hat{\omega}$ is used for phase compensation, and (3) the degree to which $\hat{\omega}$ matches ω. In all cases involving frequency estimation, we treat $\hat{\omega}$ as a nonrandom parameter that is externally provided.

6.4.1 Exact Moment Evaluations

Since from Eq. (6-6) U^+ and U^- are sums of squared norms of complex Gaussian RVs, then they themselves are chi-square distributed, each with $2N$ degrees of freedom. Furthermore, since U^+ and U^- are independent, then the moments of their ratio can be computed from the product of the positive moments of U^+ and the positive moments of $1/U^-$ (or equivalently the negative moments of U^-), i.e.,

$$E\left\{\left(\frac{U^+}{U^-}\right)^k\right\} = E\left\{\left(U^+\right)^k\right\} E\left\{\left(U^-\right)^{-k}\right\} \tag{6-23}$$

Based on the availability of closed-form expressions for these positive and negative moments for both central and non-central chi-square RVs [8], we shall see shortly that it is possible to make use of these expressions to evaluate the first two moments of the SSME either in closed form or as an infinite series whose terms are expressible in terms of tabulated functions. In each case considered, the method for doing so will be indicated but the explicit details for carrying it out will be omitted for the sake of brevity, and only the final results will be presented.

•Case 0: No Frequency Uncertainty
$\left(\omega = \hat{\omega} = \omega_{sy} = 0 \Rightarrow \delta = \hat{\delta} = \delta_{sy} = \hat{\delta}_{sy} = 0\right)$

Since in this case $W(0) = W_{N_s}(0) = W_0(0) = 1$, then we have from Eq. (6-19) that $h^+ = \hat{h}^+ = 1, h^- = \hat{h}^- = 0$ and $\hat{R} = \left(U^+ - U^-\right)/U^-$, which was previously arrived at in Eq. (6-12). Since $\hat{R} + 1 = U^+/U^-$ is the ratio of a non-central to a central chi-square RV, each with $2N$ degrees of freedom, then the mean and variance of $\hat{R}$ can be readily evaluated as

$$E\left\{\hat{R}\right\} = \frac{N}{N-1}R + \frac{1}{N-1}$$
$$\text{var}\left\{\hat{R}\right\} = \frac{1}{N-2}\left(\frac{N}{N-1}\right)^2\left[(1+2R)\left(\frac{2N-1}{N}\right) + R^2\right] \tag{6-24}$$

Since N is known, the bias of the estimator is easily removed in this case by defining a *bias-removed estimator* $\hat{R}_0 = [(N-1)/N]R - 1/N$ whose mean and variance now become

$$E\left\{\hat{R}_0\right\} = R$$
$$\text{var}\left\{\hat{R}_0\right\} = \frac{1}{N-2}\left[(1+2R)\left(\frac{2N-1}{N}\right) + R^2\right] \tag{6-25}$$

•Case 1: Frequency Uncertainty, No Frequency Estimation (and thus No Phase Compensation)
$\left(\omega \neq 0, \hat{\omega} = \omega_{sy} = \omega_s = 0 \Rightarrow \delta \neq 0, \hat{\delta} = 0, \delta_{sy} = \delta_s = \delta, \hat{\delta}_{sy} = \hat{\delta}_s = 0\right)$

For this case, $h^{\pm} = W_{N_s}(\delta)[1 \pm W_0(\delta)]/2$ for the sampled version or $h^{\pm} = W(\delta)[1 \pm W_0(\delta)]/2$ for the I&D version, $\hat{h}^+ = 1, \hat{h}^- = 0$, and again $\hat{R} = (U^+ - U^-)/U^-$. Since h^- is non-zero, then $\hat{R} + 1 = U^+/U^-$ is now the ratio of two non-central chi-square RVs, each with $2N$ degrees of freedom. Using [8, Eq. (2.47)] to evaluate the first and second positive moments of U^+ and the first and second negative moments of U^-, then using these in Eq. (6-23) allows one, after some degree of effort and manipulation, to obtain the mean and variance of $\hat{R} + 1$, from which the mean and variance of $\hat{R}$ can be evaluated as

$$E\left\{\hat{R}\right\} = \frac{N}{N-1}(1 + h^+R)\,{}_1F_1\left(1; N; -Nh^-R\right) - 1$$
$$\text{var}\left\{\hat{R}\right\} = \left(\frac{N}{N-1}\right)^2\left\{\left(\frac{N-1}{N-2}\right)\left[\frac{(1+2h^+R)}{N} + \left(1+h^+R\right)^2\right]\right. \tag{6-26}$$
$$\left. \times\ {}_1F_1\left(2; N; -Nh^-R\right) - \left(1+h^+R\right)^2\left[{}_1F_1\left(1; N; -Nh^-R\right)\right]^2\right\}$$

where ${}_1F_1\left(a;b;z\right)$ is the confluent hypergeometric function [9]. Since ω and thus $h^{\pm}$ are now unknown, the bias of the estimator cannot be removed in this case. Furthermore, since ${}_1F_1\left(a;b;0\right)=1$, then when $h^{+}=1$ and $h^{-}=0$, Eq. (6-26) immediately reduces to Eq. (6-24) as it should.

•Case 2a: Frequency Uncertainty, Perfect Frequency Estimation, No Phase Compensation
$\left(\omega\neq 0,\hat{\omega}=\omega,\omega_{sy}=\omega_s=0\Rightarrow\delta=\hat{\delta}\neq 0,\delta_{sy}=\hat{\delta}_{sy}=\delta_s=\hat{\delta}_s=\delta\right)$

For this case, $h^{\pm}=\hat{h}^{\pm}=W_{N_s}\left(\delta\right)\left[1\pm W_0\left(\delta\right)\right]/2$ for the sampled version or $h^{\pm}=\hat{h}^{\pm}=W\left(\delta\right)\left[1\pm W_0\left(\delta\right)\right]/2$ for the I&D version, and $\hat{R}$ is given by the generic form of Eq. (6-10). Obtaining an exact compact closed-form expression in this case is much more difficult since $h^{\pm}$ and $\hat{h}^{\pm}$ are now *all* non-zero. However, it is nevertheless possible to obtain an expression in the form of an infinite series. In particular, defining $\hat{\xi}\triangleq\hat{h}^{-}/\hat{h}^{+}=\tan^2\left(\hat{\delta}_{sy}/4\right)$ (for this case, $\hat{\xi}=\tan^2\left[\delta/4\right]$) and $\Lambda=U^{+}/U^{-}$, then after considerable effort and manipulation, the mean and variance of $\hat{R}$ can be evaluated in terms of the moments of Λ as

$$\begin{aligned}E\left\{\hat{R}\right\}&=-1+\left(1-\hat{\xi}\right)\sum_{n=1}^{\infty}\hat{\xi}^{n-1}E\left\{\Lambda^n\right\}\\ \mathrm{var}\left\{\hat{R}\right\}&=\left(1-\hat{\xi}\right)^2\left[\sum_{n=2}^{\infty}\left(n-1\right)\hat{\xi}^{n-2}E\left\{\Lambda^n\right\}-\left(\sum_{n=1}^{\infty}\hat{\xi}^{n-1}E\left\{\Lambda^n\right\}\right)^2\right]\end{aligned}\tag{6-27}$$

where

$$E\left\{\Lambda^n\right\}=\frac{\Gamma\left(N+n\right)\Gamma\left(N-n\right)}{\Gamma^2\left(N\right)}{}_1F_1\left(-n;N;-Nh^{+}R\right){}_1F_1\left(n;N;-Nh^{-}R\right)\tag{6-28}$$

For small frequency error, i.e., $\hat{\xi}$ small, Eq. (6-27) can be simply approximated by

$$\begin{aligned}E\left\{\hat{R}\right\}&=-1+E\left\{\Lambda\right\}+\hat{\xi}\left[E\left\{\Lambda^2\right\}-E\left\{\Lambda\right\}\right]\\ \mathrm{var}\left\{\hat{R}\right\}&=\left(1-2\hat{\xi}\right)\times\mathrm{var}\left\{\Lambda\right\}+2\hat{\xi}\left[E\left\{\Lambda^3\right\}-E\left\{\Lambda\right\}E\left\{\Lambda^2\right\}\right]\end{aligned}\tag{6-29}$$

Although not obvious from Eq. (6-27), it can be shown that the mean of the SNR estimator can be written in the form $E\{\hat{R}\} = R + O(1/N)$ and thus, for this case, the estimator is *asymptotically* (large N) unbiased.

•Case 2b: Frequency Uncertainty, Perfect Frequency Estimation, Half-Symbol Phase Compensation
$\left(\omega \neq 0, \hat{\omega} = \omega, \omega_{sy} = \omega, \omega_s = 0 \Rightarrow \delta = \hat{\delta} \neq 0, \delta_{sy} = \hat{\delta}_{sy} = 0, \delta_s = \hat{\delta}_s = \delta\right)$

Here we have $h^+ = \hat{h}^+ = W_{N_s}(\delta)$ for the sampled version or $h^+ = \hat{h}^+ = W(\delta)$ for the I&D version, $h^- = \hat{h}^- = 0$, and thus $\hat{R} = [(U^+ - U^-)/U^-]/\hat{h}^+$. Recognizing then that $\hat{h}^+\hat{R} + 1 = U^+/U^-$, the moments of $\hat{h}^+\hat{R}$ can be directly obtained from the moments of $\hat{R}$ of Case 0 by replacing R with h^+R. Thus,

$$E\left\{\hat{R}\right\} = \frac{1}{\hat{h}^+}\left[\frac{N}{N-1}h^+R + \frac{1}{N-1}\right] \tag{6-30}$$

$$\text{var}\left\{\hat{R}\right\} = \frac{1}{\left(\hat{h}^+\right)^2}\frac{1}{N-2}\left(\frac{N}{N-1}\right)^2\left[\left(1+2h^+R\right)\left(\frac{2N-1}{N}\right) + \left(h^+R\right)^2\right]$$

where for this case, as noted above, we can further set $h^+ = \hat{h}^+$. Once this is done in Eq. (6-30), then since $\hat{h}^+$ is known, we can once again completely remove the bias from the estimator by defining the bias-removed estimator $\hat{R}_0 = \left[(N-1)/N\right]R - 1/\left(N\hat{h}^+\right)$, whose mean is given by $E\{\hat{R}_0\} = R$ and whose variance is obtained from $\text{var}\{\hat{R}\}$ of Eq. (6-30) by multiplying it by $\left[(N-1)/N\right]^2$.

•Case 2c: Frequency Uncertainty, Perfect Frequency Estimation, Sample-by-Sample Phase Compensation
$\left(\omega \neq 0, \hat{\omega} = \omega, \omega_{sy} = \omega_s = \omega \Rightarrow \delta = \hat{\delta} \neq 0, \delta_{sy} = \hat{\delta}_{sy} = \delta_s = \hat{\delta}_s = 0\right)$

This case applies only to the sample-by-sample version of the SSME. In particular, we have $h^+ = \hat{h}^+ = 1, h^- = \hat{h}^- = 0$, and thus $\hat{R} = (U^+ - U^-)/U^-$, which is identical to the SSME of Case 0. Thus, the moments of $\hat{R}$ are given by Eq. (6-24).

•Case 3a: Frequency Uncertainty, Imperfect Frequency Estimation, No Phase Compensation

$\left(\omega \neq 0, \hat{\omega} \neq \omega, \omega_{sy} = \omega_s = 0 \Rightarrow \delta, \hat{\delta} \neq 0, \delta_{sy} = \delta_s = \delta, \hat{\delta}_{sy} = \hat{\delta}_s = \hat{\delta}\right)$

Here, $h^{\pm} = W_{N_s}(\delta)\left[1 \pm W_0(\delta)\right]/2, \hat{h}^{\pm} = W_{N_s}\left(\hat{\delta}\right)\left[1 \pm W_0\left(\hat{\delta}\right)\right]/2$ for the sampled version or $h^{\pm} = W(\delta)\left[1 \pm W_0(\delta)\right]/2, \hat{h}^{\pm} = W\left(\hat{\delta}\right)\left[1 \pm W_0\left(\hat{\delta}\right)\right]/2$ for the I&D version, and $\hat{R}$ is given by the generic form of Eq. (6-10). The method used to obtain the moments of the SNR estimator is analogous to that used for Case 2a. In particular, noting that for this case $\hat{\xi} = \tan^2\left(\hat{\delta}/4\right)$, the results are obtained from Eq. (6-27) by multiplying $E\{\hat{R}\}$ by $1/\hat{h}^+$ and $\mathrm{var}\{\hat{R}\}$ by $\left(1/\hat{h}^+\right)^2$.

•Case 3b: Frequency Uncertainty, Imperfect Frequency Estimation, Half-Symbol Phase Compensation

$\left(\omega \neq 0, \hat{\omega} \neq \omega, \omega_{sy} = \hat{\omega}, \omega_s = 0 \Rightarrow \delta, \hat{\delta} \neq 0, \delta_{sy} = \delta - \hat{\delta}, \hat{\delta}_{sy} = 0,\ \delta_s = \delta, \hat{\delta}_s = \hat{\delta}\right)$

Here $h^{\pm} = W_{N_s}(\delta)\left[1 \pm W_0\left(\delta - \hat{\delta}\right)\right]/2, \hat{h}^+ = W_{N_s}\left(\hat{\delta}\right)$ for the sampled version or $h^{\pm} = W\left(\delta\right)\left[1 \pm W_0\left(\delta - \hat{\delta}\right)\right]/2, \hat{h}^+ = W\left(\hat{\delta}\right)$ for the I&D version, $\hat{h}^- = 0$, and once again $\hat{R} = \left[\left(U^+ - U^-\right)/U^-\right]/\hat{h}^+$. Hence, by analogy with Case 1, the mean and variance of the SNR estimator can be obtained from a scaled version of Eq. (6-26).

•Case 3c: Frequency Uncertainty, Imperfect Frequency Estimation, Sample-by-Sample Phase Compensation

$\left(\omega \neq 0, \hat{\omega} \neq \omega, \omega_{sy} = \omega_s = \hat{\omega} \Rightarrow \delta, \hat{\delta} \neq 0, \delta_{sy} = \delta_s = \delta - \hat{\delta}, \hat{\delta}_{sy} = \hat{\delta}_s = 0\right)$

This case applies only to the sample-by-sample version of the SSME. In particular, we have $h^{\pm} = W_{N_s}\left(\delta - \hat{\delta}\right)\left[1 \pm W_0\left(\delta - \hat{\delta}\right)\right]/2, \hat{h}^+ = 1, \hat{h}^- = 0$, and thus $\hat{R} = \left(U^+ - U^-\right)/U^-$, which is the form given in Eq. (6-12) and resembles Case 1. Thus, the moments of $\hat{R}$ are given by Eq. (6-26), using now the values of h^+ and h^- as are appropriate to this case.

6.4.2 Asymptotic Moment Evaluations

Despite having exact results, in many instances it is advantageous to have asymptotic results, particularly if their analytical form is less complex and as such lends insight into the their behavior in terms of the various system parameters. In this section, we provide approximate expressions for the mean and variance of the SSME by employing a Taylor series expansion of $g\left(U^+, U^-\right)$ in

Eq. (6-10), assuming that this function is smooth in the vicinity of the point $\left(E\{U^+\}, E\{U^-\}\right)$. With this in mind, the mean and variance of the estimate $\hat{R}$ are approximated by [10, p. 212]

$$E\{\hat{R}\} = g\left(E\left\{U^+\right\}, E\left\{U^-\right\}\right) + \frac{1}{2}\left(\operatorname{var}\left\{U^+\right\}\frac{\partial^2 g}{\partial\left(U^+\right)^2} + \operatorname{var}\left\{U^-\right\}\frac{\partial^2 g}{\partial\left(U^-\right)^2}\right) + O\left(\frac{1}{N^2}\right) \quad (6\text{-}31)$$

$$\operatorname{var}\{\hat{R}\} = \left(\frac{\partial g}{\partial U^+}\right)^2 \operatorname{var}\{U^+\} + \left(\frac{\partial g}{\partial U^-}\right)^2 \operatorname{var}\{U^-\} + O\left(\frac{1}{N^2}\right)$$

In Eq. (6-31), all of the partial derivatives are evaluated at $\left(E\{U^+\}, E\{U^-\}\right)$. Ordinarily, there would be another term in these Taylor series expansions involving $\partial^2 g/\partial U^+ \partial U^-$ and $\operatorname{cov}\{U^+, U^-\}$. However, in our case, this term is absent in view of the independence of U^+ and U^-.

In Appendix 6-A, we derive explicit expressions for $E\{\hat{R}\}$ and $\operatorname{var}\{\hat{R}\}$ based on the evaluations of the partial derivatives required in Eq. (6-31). The results of these evaluations are given below:

$$E\left\{\hat{R}\right\} = \frac{\left(h^+ - h^-\right)R}{\hat{h}^+ - \hat{h}^- + \left(\hat{h}^+ h^- - \hat{h}^- h^+\right)R} + \frac{1}{N}\frac{\left(\hat{h}^+ - \hat{h}^-\right)\left(\hat{h}^+ + \hat{h}^-\right)}{\left[\hat{h}^+ - \hat{h}^- + \left(\hat{h}^+ h^- - \hat{h}^- h^+\right)R\right]^3} \times \left\{1 + \left(h^+ + h^- + \frac{\hat{h}^+ h^- + \hat{h}^- h^+}{\hat{h}^+ + \hat{h}^-}\right)R + 2h^+ h^- R^2\right\} + O\left(\frac{1}{N^2}\right) \quad (6\text{-}32)$$

and

$$\operatorname{var}\left\{\hat{R}\right\} = \frac{1}{N}\frac{\left(\hat{h}^{+}-\hat{h}^{-}\right)^{2}}{\left[\hat{h}^{+}-\hat{h}^{-}+\left(\hat{h}^{+}h^{-}-\hat{h}^{-}h^{+}\right)R\right]^{4}}$$

$$\times\left\{2+4\left(h^{+}+h^{-}\right)R+\left[\left(h^{+}+h^{-}\right)^{2}+6h^{+}h^{-}\right]R^{2}+4h^{+}h^{-}\left(h^{+}+h^{-}\right)R^{3}\right\}$$

$$+O\left(\frac{1}{N^{2}}\right) \tag{6-33}$$

It is now a simple matter to substitute in the various expressions for $h^{\pm}$ and $\hat{h}^{\pm}$ corresponding to the special cases treated in Section 6.1 to arrive at asymptotic closed-form expressions for the mean and variance of $\hat{R}$ for each of these cases. The results of these substitutions lead to the following simplifications:

•Case 0: No Frequency Uncertainty

$$\begin{aligned} E\{\hat{R}\} &= R+\frac{1}{N}(1+R)+O\left(\frac{1}{N^{2}}\right) \\ \operatorname{var}\left\{\hat{R}\right\} &= \frac{1}{N}\left(2+4R+R^{2}\right)+O\left(\frac{1}{N^{2}}\right) \end{aligned} \tag{6-34}$$

•Case 1: Frequency Uncertainty, No Frequency Estimation (and thus No Phase Compensation)

$$E\left\{\hat{R}\right\} = \frac{\left(h^{+}-h^{-}\right)R}{1+h^{-}R}$$

$$+\frac{1}{N}\frac{1}{\left(1+h^{-}R\right)^{3}}\left\{1+\left(h^{+}+2h^{-}\right)R+2h^{+}h^{-}R^{2}\right\}+O\left(\frac{1}{N^{2}}\right) \tag{6-35}$$

$$\operatorname{var}\left\{\hat{R}\right\} = \frac{1}{N}\frac{1}{\left(1+h^{-}R\right)^{4}}\left\{2+4\left(h^{+}+h^{-}\right)R+\left[\left(h^{+}+h^{-}\right)^{2}+6h^{+}h^{-}\right]R^{2}\right.$$

$$\left.+4h^{+}h^{-}\left(h^{+}+h^{-}\right)R^{3}\right\}+O\left(\frac{1}{N^{2}}\right)$$

where $h^{\pm} = W_{N_s}(\delta)\left[1 \pm W_0(\delta)\right]/2$ for the sampled version or $h^{\pm} = W(\delta)\left[1 \pm W_0(\delta)\right]/2$ for the I&D version.

•Case 2a: Frequency Uncertainty, Perfect Frequency Estimation, No Phase Compensation

$$E\left\{\hat{R}\right\} =$$

$$R + \frac{1}{N}\frac{(h^+ + h^-)}{(h^+ - h^-)^2}\left\{1 + \left(h^+ + h^- + \frac{2h^+h^-}{h^+ + h^-}\right)R + 2h^+h^-R^2\right\} + O\left(\frac{1}{N^2}\right) \tag{6-36}$$

$$\text{var}\left\{\hat{R}\right\} = \frac{1}{N}\frac{1}{(h^+ - h^-)^2}\left\{2 + 4\left(h^+ + h^-\right)R + \left[\left(h^+ + h^-\right)^2 + 6h^+h^-\right]R^2\right.$$

$$\left. + 4h^+h^-\left(h^+ + h^-\right)R^3\right\} + O\left(\frac{1}{N^2}\right)$$

where $h^{\pm} = W_{N_s}(\delta)\left[1 \pm W_0(\delta)\right]/2$ for the sampled version or $h^{\pm} = W(\delta)\left[1 \pm W_0(\delta)\right]/2$ for the I&D version.

•Case 2b: Frequency Uncertainty, Perfect Frequency Estimation, Half-Symbol Phase Compensation

$$E\left\{\hat{R}\right\} = R + \frac{1}{N}\frac{1}{h^+}\left(1 + h^+R\right) + O\left(\frac{1}{N^2}\right)$$

$$\text{var}\left\{\hat{R}\right\} = \frac{1}{N}\frac{1}{(h^+)^2}\left[2 + 4h^+R + \left(h^+\right)^2R^2\right] + O\left(\frac{1}{N^2}\right) \tag{6-37}$$

where $h^+ = W_{N_s}(\delta)$ for the sampled version or $h^+ = W(\delta)$ for the I&D version.

•Case 2c: Frequency Uncertainty, Perfect Frequency Estimation, Sample-by-Sample Phase Compensation

As was true for the exact results, the asymptotic mean and variance are again the same as for Case 0.

•Case 3a: Frequency Uncertainty, Imperfect Frequency Estimation, No Phase Compensation

No simplification of the results occurs here, and thus one merely applies Eqs. (6-32) and (6-33), where $h^{\pm} = W_{N_s}(\delta)\left[1 \pm W_0(\delta)\right]/2, \hat{h}^{\pm} = W_{N_s}(\hat{\delta})\left[1 \pm W_0\left(\hat{\delta}\right)\right]/2$ for the sampled version or $h^{\pm} = W(\delta)\left[1 \pm W_0(\delta)\right]/2, \hat{h}^{\pm} = W\left(\hat{\delta}\right)\left[1 \pm W_0\left(\hat{\delta}\right)\right]/2$ for the I&D version.

•Case 3b: Frequency Uncertainty, Imperfect Frequency Estimation, Half-Symbol Phase Compensation

$$E\left\{\hat{R}\right\} = \frac{\left(h^{+} - h^{-}\right)R}{\hat{h}^{+}\left(1 + h^{-}R\right)} + \frac{1}{N}\frac{1}{\hat{h}^{+}}\frac{1}{\left(1 + h^{-}R\right)^{3}}$$
$$\times\left\{1 + \left(h^{+} + 2h^{-}\right)R + 2h^{+}h^{-}R^{2}\right\} + O\left(\frac{1}{N^{2}}\right) \tag{6-38}$$

$$\mathrm{var}\left\{\hat{R}\right\} = \frac{1}{N}\frac{1}{\left(\hat{h}^{+}\right)^{2}}\frac{1}{\left(1 + h^{-}R\right)^{4}}$$
$$\times\left\{2 + 4\left(h^{+} + h^{-}\right)R + \left[\left(h^{+} + h^{-}\right)^{2} + 6h^{+}h^{-}\right]R^{2}\right.$$
$$\left.+4h^{+}h^{-}\left(h^{+} + h^{-}\right)R^{3}\right\} + O\left(\frac{1}{N^{2}}\right)$$

where $h^{\pm} = W_{N_s}(\delta)\left[1 \pm W_0\left(\delta - \hat{\delta}\right)\right]/2, \hat{h}^{+} = W_{N_s}\left(\hat{\delta}\right)$ for the sampled version or $h^{\pm} = W(\delta)\left[1 \pm W_0\left(\delta - \hat{\delta}\right)\right]/2, \hat{h}^{+} = W\left(\hat{\delta}\right)$ for the I&D version.

•Case 3c: Frequency Uncertainty, Imperfect Frequency Estimation, Sample-by-Sample Phase Compensation

$$E\{\hat{R}\} = \frac{(h^+ - h^-)R}{1 + h^- R} + \frac{1}{N}\frac{1}{(1 + h^- R)^3} \times \{1 + (h^+ + 2h^-)R + 2h^+ h^- R^2\} + O(\frac{1}{N^2}) \tag{6-39}$$

$$\text{var}\{\hat{R}\} = \frac{1}{N}\frac{1}{(1 + h^- R)^4}\left\{2 + 4(h^+ + h^-)R + \left[(h^+ + h^-)^2 + 6h^+ h^-\right]R^2 + 4h^+ h^-(h^+ + h^-)R^3\right\} + O\left(\frac{1}{N^2}\right)$$

where $h^{\pm} = W_{N_s}(\delta - \hat{\delta})\left[1 \pm W_0(\delta - \hat{\delta})\right]$.

6.4.2.1. Numerical Results and Comparisons. To compare the performances of the estimator corresponding to the various cases just discussed, we first define a parameter $\hat{N} = N\text{var}\{\hat{R}\}/R^2$ (or in the cases where a bias-removed estimator is possible, $\hat{N}_0 = N\text{var}\{\hat{R}_0\}/R^2$), which measures the number of symbols that are needed to achieve a fractional mean-squared estimation error of 100 percent using that estimator. Then, if one wishes to achieve a smaller fractional mean-squared estimation error, say $\text{var}\{\hat{R}\}/R^2 = \varepsilon^2$ (or $\text{var}\{\hat{R}_0\}/R^2 = \varepsilon^2$), then the required number of symbols to achieve this level of performance would simply be $N_{req}(\varepsilon^2) = \hat{N}/\varepsilon^2$ (or $N_{req}(\varepsilon^2) = \hat{N}_0/\varepsilon^2$). As an example, consider the bias-removed SNR estimator for Case 2b for which $\hat{N}_0$ can be determined from Eq. (6-30) as

$$\hat{N}_0 = \frac{\left(1 - \frac{1}{2N}\right)\left(\frac{2}{(h^+ R)^2} + \frac{4}{h^+ R}\right) + 1}{1 - \frac{2}{N}} \tag{6-40}$$

Clearly, the above interpretation of the meaning of $\hat{N}_0$ is a bit circular in that $\hat{N}_0$ of Eq. (6-40) depends on N. However, this dependence is mild for reasonable values of N. Thus, to a good approximation one can replace $\hat{N}_0$ by its limiting value $\hat{N}_0^*$ corresponding to $N = \infty$, in which case the required number of symbols

to achieve a fractional mean-squared estimation of ε^2 would approximately be given by

$$N_{req}(\varepsilon^2) \cong \frac{\hat{N}_0^*}{\varepsilon^2}$$
$$\hat{N}_0^* = \frac{2}{(h^+R)^2} + \frac{4}{h^+R} + 1 \tag{6-41}$$

Alternatively, for this case one uses the exact expression for the fractional mean-squared estimation error to solve directly for $N_{req}(\varepsilon^2)$. In particular, dividing Eq. (6-30) (multiplied by $\left[(N-1)/N\right]^2$) by R^2 and equating the result to ε^2 results in a quadratic equation in N whose solution can be exactly expressed as

$$N_{req}\left(\varepsilon^2\right) = \left(1 + \frac{\hat{N}_0^*}{2\varepsilon^2}\right)\left[1 + \sqrt{1 - \frac{\left(\hat{N}_0^* - 1\right)2\varepsilon^2}{\left(\hat{N}_0^* + 2\varepsilon^2\right)^2}}\right] \tag{6-42}$$

Since the value of the negative term in the square root is less than $2\varepsilon^2/\hat{N}_0^*$, an approximate (for small ε^2) upper bound on Eq, (6-42) is given by

$$N_{req}\left(\varepsilon^2\right) < \left(1 + \frac{\hat{N}_0^*}{2\varepsilon^2}\right)\left[1 + \sqrt{1 - \frac{2\varepsilon^2}{\hat{N}_0^*}}\right] \cong \left(1 + \frac{\hat{N}_0^*}{2\varepsilon^2}\right)\left(2 - \frac{\varepsilon^2}{\hat{N}_0^*}\right) \cong \frac{\hat{N}_0^*}{\varepsilon^2} + \frac{3}{2} \tag{6-43}$$

Thus, we see that the exact number of requisite symbols is not more than two extra symbols beyond the number that would be obtained from the approximate number in Eq. (6-41).

Figure 6-4 is a plot of $\hat{N}_0$ versus R in decibels with N as a parameter for the biased-removed estimator of Case 0, where the results are determined from Eq. (6-25). We observe that a value of $N = 50$ is virtually sufficient to approach the asymptotic value $\hat{N}_0^* = 1 + 2(1 + 2R)/R^2$. For the biased-removed estimator of Case 2b, a plot of $\hat{N}_0$ versus h^+R in decibels would be identical to Fig. 6-4, in accordance with Eq. (6-30) and the comments below this equation. Thus, the degradation in performance when frequency uncertainty is present but is perfectly estimated and fully compensated for is reflected in a horizontal shift of the curves in Fig. 6-4 to the right by an amount equal to h^+. Equivalently, a

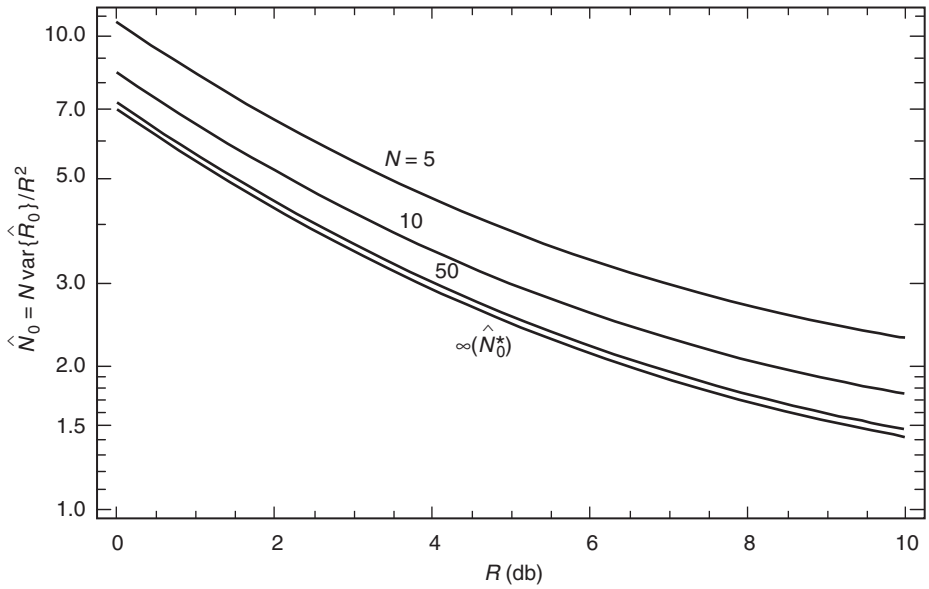


Fig. 6-4. Case 0: No frequency offset, perfect frequency estimate, no phase compensation.

larger number of symbols is now required to achieve the same SNR estimation accuracy as for the case of no frequency uncertainty.

For Case 3b where the frequency uncertainty estimate is imperfect but is still used for compensation, the asymptotic (N large) behavior is obtained from Eq. (6-33). In particular, for $N \to \infty$,

$$E\{\hat{R}\} = \frac{(h^+ - h^-)R}{\hat{h}^+(1 + h^- R)}$$

$$\hat{N}^* = \lim_{N\to\infty} \frac{N\text{var}(\hat{R})}{R^2} = \frac{1}{\left(\hat{h}^+\right)^2 (1 + h^- R)^4} \left\{ \frac{2}{R^2} + \frac{4(h^+ + h^-)}{R} \right. \quad (6\text{-}44)$$

$$\left. + (h^+ + h^-)^2 + 6h^+h^- + 4h^+h^-(h^+ + h^-)R \right\}$$

Figures 6-5 and 6-6 are plots of $E\{\hat{R}\}$ versus R in decibels for fixed $\delta/(2\pi) = fT$ and fractional frequency estimation error $(\delta - \hat{\delta})/\delta$ as a parameter varying between -10 percent and 10 percent. We observe that for a relative frequency

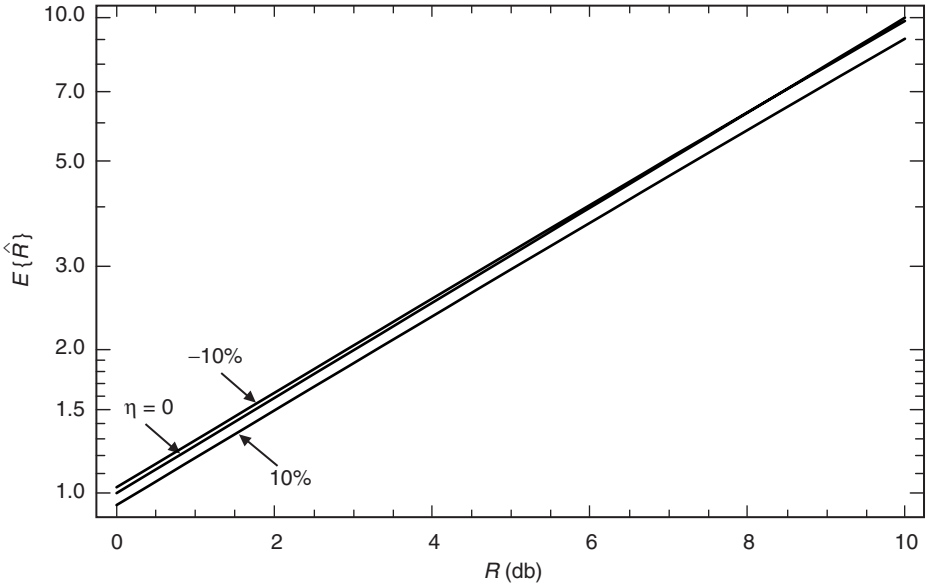


Fig. 6-5. Mean of estimator versus SNR in decibels; Case 3b; frequency offset, imperfect frequency estimate, half-symbol phase compensation; relative frequency uncertainty $\delta/(2\pi) = fT = 0.5$, fractional frequency error $\eta = 0, -10\%, 10\%$.

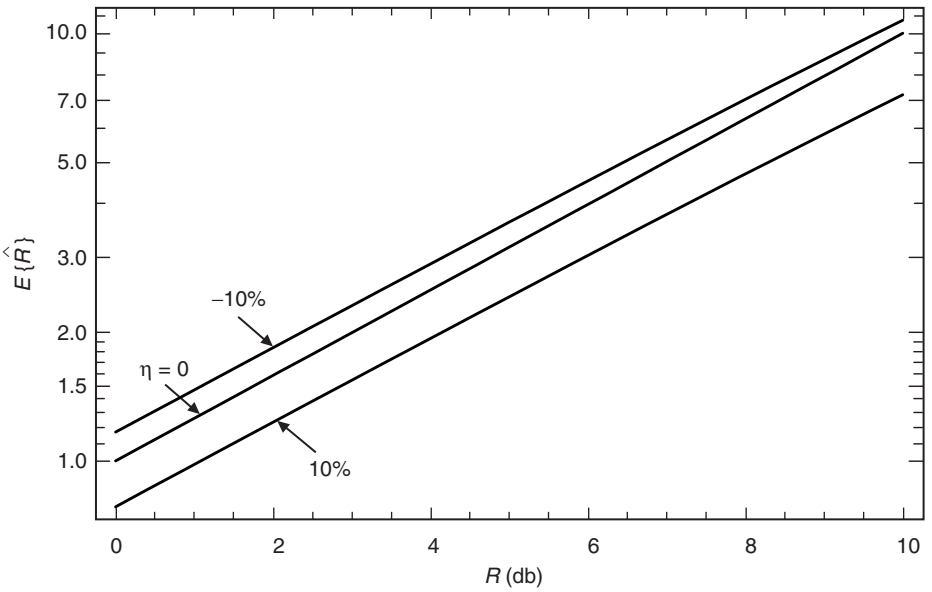


Fig. 6-6. Mean of estimator versus SNR in decibels; Case 3b; frequency offset, imperfect frequency estimate, half-symbol phase compensation; relative frequency uncertainty $\delta/(2\pi) = fT = 1.0$, fractional frequency error $\eta = 0, -10\%, 10\%$.

uncertainty of half a cycle ($\delta/(2\pi) = 0.5$), the amount of bias is quite small over the range of frequency estimation errors considered. When the relative frequency error increases to a full cycle ($\delta/(2\pi) = 1.0$), then the sensitivity of the bias to frequency estimation error becomes more pronounced. Also, it can be observed that, for a fixed frequency offset, the bias is not a symmetric function of the frequency estimation error. Figure 6-7 is a plot of $\hat{N}^*$ versus R in decibels for a fixed fractional estimation error $\eta = 5$ percent and $\delta/(2\pi) = fT$ as a parameter varying between 0.5 and 0.9. These curves are analogous to the ones in Fig. 6-4 with the purpose of demonstrating the sensitivity of the number of symbols required for a given level of mean-squared error performance to frequency uncertainty and estimation error.

6.5 SNR Estimation in the Presence of Symbol Timing Error

Until now we have considered the performance of the SSME estimator assuming that the symbol timing was either known or could be perfectly estimated. In this section, we extend the previous results corresponding to the I&D version[3]

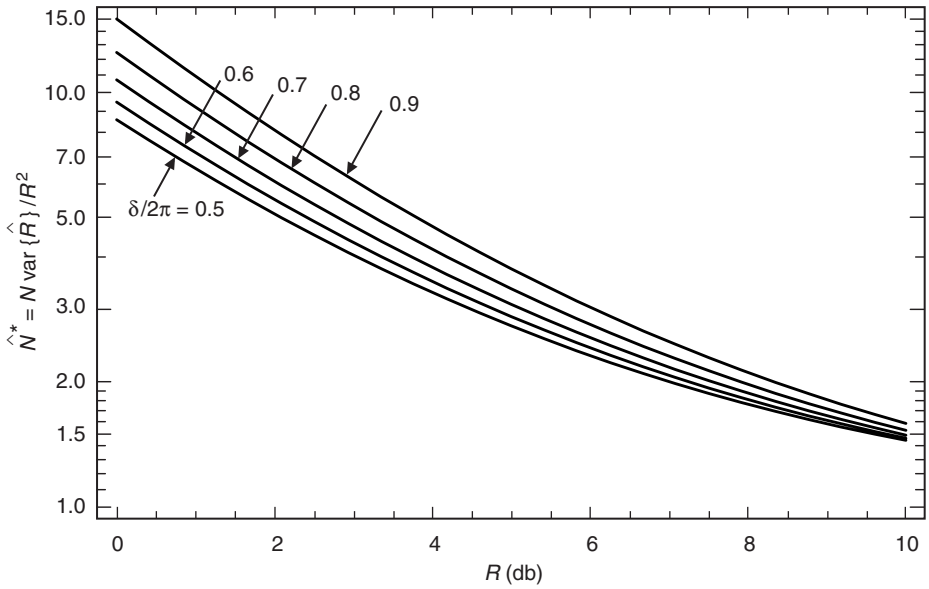


Fig. 6-7. Case 3b: frequency offset, imperfect frequency estimate, half-symbol phase compensation; fractional frequency uncertainty error $\eta = 5\%$.

[3] Extension of the results to the sampled version is straightforward and is omitted for the sake of brevity.

of the SSME to the case where symbol timing is imperfect but the carrier frequency is known[4]—carrier phase is still assumed unknown. Clearly, although in any realistic system implementation both frequency uncertainty and symbol timing error will exist simultaneously, treating them as separate entities gives us a means of obtaining analytical results for their individual behavior and the sensitivity of system performance to each. The true degradation in the performance of the SNR estimator in their joint presence must be determined from simulation results.

6.5.1 Signal Model and Formation of the Estimator

Corresponding to the complex baseband received signal in the kth interval $(k-1)T \leq t < kT$, as described by Eq. (6-13), in the presence of a symbol timing error εT (for the moment we assume $0 \leq \varepsilon \leq 1/2$), the outputs of the first and second half-symbol I&Ds are given by

$$Y_{\alpha k} = m d_k \frac{1}{T} \int_{(k-1+\varepsilon)T}^{(k-1/2+\varepsilon)T} e^{j\phi} dt + \frac{1}{T} \int_{(k-1+\varepsilon)T}^{(k-1/2+\varepsilon)T} n(t)\, dt = \frac{m d_k}{2} e^{j\phi} + n_{\alpha k}$$

$$\begin{aligned} Y_{\beta k} &= m d_k \frac{1}{T} \int_{(k-1/2+\varepsilon)T}^{kT} e^{j\phi} dt + m d_{k+1} \frac{1}{T} \int_{kT}^{(k+\varepsilon)T} e^{j(\omega t+\phi)} dt \\ &\quad + \frac{1}{T} \int_{(k-1/2+\varepsilon)T}^{(k+\varepsilon)T} n(t)\, dt \\ &= m e^{j\phi} \left[d_k \left(\frac{1}{2} - \varepsilon \right) + d_{k+1} \varepsilon \right] + n_{\beta k} \end{aligned} \tag{6-45}$$

Independent of the symbol timing offset εT, the complex noise variables $n_{\alpha k}$ and $n_{\beta k}$ are still zero-mean Gaussian with variance $\sigma^2/2$ for each real and imaginary component. From the observables in Eq. (6-45), we again form the sum and difference variables

$$u_k^{\pm} = Y_{\alpha k} \pm Y_{\beta k} = m e^{j\phi} \left[d_k \left(\frac{1}{2} \pm \left(\frac{1}{2} - \varepsilon \right) \right) \pm d_{k+1} \varepsilon \right] + n_{\alpha k} \pm n_{\beta k} \triangleq s_k^{\pm} + n_k^{\pm} \tag{6-46}$$

[4] Later on, we shall consider one particular system model that allows for frequency uncertainty with perfect compensation and that falls in the same mathematical framework.

It is straightforward to show that the normalized squared norm of the signal components can be evaluated as

$$h_k^+ \triangleq \frac{\left|s_k^+\right|^2}{m^2} = 1 - 4\varepsilon\,(1-\varepsilon)\sin^2\left(\frac{\Delta\phi_{k+1}}{2}\right)$$
$$h_k^- \triangleq \frac{\left|s_k^-\right|^2}{m^2} = 4\varepsilon^2\sin^2\left(\frac{\Delta\phi_{k+1}}{2}\right) \tag{6-47}$$

where $\Delta\phi_{k+1} = \phi_{k+1} - \phi_k$ denotes the transition in the data symbol phase in going from the kth symbol interval to the k+1st. Note that, as in the previous publication on the subject [6], the parameters $h_k^{\pm}$ do not depend on the random carrier phase ϕ; however, unlike these previous investigations, they do now depend on the data via the transitions in the symbol phase sequence. Furthermore, it is not obvious at this point (this will become clear shortly) to what extent $h_k^{\pm}$ is independent of the order M of the M-PSK modulation. For $-1/2 \le \varepsilon \le 0$, the analogous relations to Eq. (6-47) are

$$h_k^+ = 1 + 4\varepsilon\,(1+\varepsilon)\sin^2\left(\frac{\Delta\phi_k}{2}\right)$$
$$h_k^- = 4\varepsilon^2\sin^2\left(\frac{\Delta\phi_k}{2}\right) \tag{6-48}$$

Next, we calculate N-symbol averages of the squared norms of the half-symbol sums to produce $U^{\pm}$ as in Eq. (6-6). Once again we make the key observation (as previously proved) that, now conditioned on the data symbol transition sequence, the observables U^+ and U^- are independent RVs. Defining, as before, the true SNR by $R = m^2/2\sigma^2$, then after averaging over the uniform distribution of the data symbol transitions around the circle defining the M-PSK constellation, it is straightforward (although a bit laborious) to show that their means and variances are, analogous to Eq. (6-8), given by

$$E\left\{U^{\pm}\right\} = 2\sigma^2\left(1 + \bar{h}^{\pm}R\right)$$
$$\operatorname{var}\left\{U^{\pm}\right\} = \frac{4}{N}\sigma^4\left(1 + 2\bar{h}^{\pm}R + \operatorname{var}\left\{h^{\pm}\right\}R^2\right) \tag{6-49}$$

where the overbar on $h^{\pm}$ denotes the above-mentioned statistical averaging over the data symbol transition sequence and the subscript k has been dropped since the statistical averages do not depend on k. Making use of the relations

$$\begin{aligned}
\overline{\sin^2\left(\frac{\Delta\phi_k}{2}\right)} &= \frac{1}{M}\sum_{k=0}^{M-1}\sin^2\frac{k\pi}{M} = \frac{1}{2} \\
\overline{\sin^4\left(\frac{\Delta\phi_k}{2}\right)} &= \frac{1}{M}\sum_{k=0}^{M-1}\sin^4\frac{k\pi}{M} = \begin{cases} \frac{1}{2}, & M=2 \\ \frac{3}{8}, & M>2 \end{cases}
\end{aligned} \tag{6-50}$$

we have from Eqs. (6-47) and (6-48) that

$$\begin{aligned}
\bar{h}^{+} &= 1 - 2|\varepsilon|(1-|\varepsilon|) \\
\bar{h}^{-} &= 2|\varepsilon|^2
\end{aligned} \tag{6-51}$$

and

$$\begin{aligned}
\operatorname{var}\left\{h^{+}\right\} &= \begin{cases} 4|\varepsilon|^2(1-|\varepsilon|)^2, & M=2 \\ 2|\varepsilon|^2(1-|\varepsilon|)^2, & M>2 \end{cases} \\
\operatorname{var}\left\{h^{-}\right\} &= \begin{cases} 4|\varepsilon|^4, & M=2 \\ 2|\varepsilon|^4, & M>2 \end{cases}
\end{aligned} \tag{6-52}$$

Finally, since from the first relation in Eq. (6-49) R is expressible as

$$R = \frac{E\{U^{+}\} - E\{U^{-}\}}{\bar{h}^{+}E\{U^{-}\} - \bar{h}^{-}E\{U^{+}\}} \tag{6-53}$$

then, as in the perfect symbol timing case, the general form of the ad hoc SSME $\hat{R}$ is obtained by substituting the sample values $U^{\pm}$ for their expected values and the estimates $\hat{\bar{h}}^{\pm}$ for their true values, namely,

$$\hat{R} = \frac{U^{+} - U^{-}}{\hat{\bar{h}}^{+}U^{-} - \hat{\bar{h}}^{-}U^{+}} \tag{6-54}$$

where $\hat{\bar{h}}^{\pm}$ are obtained from $\bar{h}^{\pm}$, defined in Eq. (6-51) by substituting the symbol timing estimate $\hat{\varepsilon}$ for ε. Actually, in view of Eq. (6-51), it is necessary to have only an estimate of the magnitude of ε. A method for obtaining such an estimate based on the same statistics used to form the SNR estimator will be discussed elsewhere in the text.

6.5.2 Mean and Variance of the SNR Estimator

In this section, we evaluate the mean and variance of $\hat{R}$ using the same techniques as in previous sections of the chapter. Since, for $|\hat{\varepsilon}| > 0$, $\hat{\bar{h}}^{\pm}$ are both non-zero, obtaining an exact compact closed-form expression is difficult. Nevertheless, it is possible to obtain a closed-form expression in the form of an infinite series. In particular, defining $\hat{\xi} \triangleq \hat{\bar{h}}^{-}/\hat{\bar{h}}^{+}$ and $\Lambda \triangleq U^{+}/U^{-}$, we can express $\hat{R}$ of Eq. (6-54) in the form

$$\hat{R} = \frac{1}{\hat{\bar{h}}^{+}} \frac{\Lambda - 1}{1 - \hat{\xi}\Lambda} = \frac{1}{\hat{\bar{h}}^{+}} (\Lambda - 1) \sum_{n=0}^{\infty} \left(\hat{\xi}\Lambda\right)^{n} = \frac{1}{\hat{\bar{h}}^{+}} \left[-1 + \left(1 - \hat{\xi}\right) \sum_{n=1}^{\infty} \hat{\xi}^{n-1} \Lambda^{n}\right] \tag{6-55}$$

Thus, the mean of $\hat{R}$ is expressed in terms of the moments of Λ by

$$E\left\{\hat{R}\right\} = \frac{1}{\hat{\bar{h}}^{+}} \left[-1 + \left(1 - \hat{\xi}\right) \sum_{n=1}^{\infty} \hat{\xi}^{n-1} E\left\{\Lambda^{n}\right\}\right] \tag{6-56}$$

Similarly, the variance of $\hat{R}$ can be evaluated in terms of the moments of Λ as

$$\operatorname{var}\left\{\hat{R}\right\} = \frac{\left(1 - \hat{\xi}\right)^{2}}{\left(\hat{\bar{h}}^{+}\right)^{2}} \left[\sum_{n=2}^{\infty} (n-1)\, \hat{\xi}^{n-2} E\left\{\Lambda^{n}\right\} - \left(\sum_{n=1}^{\infty} \hat{\xi}^{n-1} E\left\{\Lambda^{n}\right\}\right)^{2}\right] \tag{6-57}$$

An expression for the moments of Λ in terms of $h^{\pm}$ can be obtained from Eq. (6-23) and [8, Eq. (2.47)] as

$$E\left\{\Lambda^{n}\right\} = \frac{\Gamma(N+n)\,\Gamma(N-n)}{\Gamma^{2}(N)} {}_1F_1\left(-n; N; -Nh^{+}R\right) {}_1F_1\left(n; N; -Nh^{-}R\right) \tag{6-58}$$

Since, in accordance with Eqs. (6-47) and (6-48), $h^{\pm}$ are now functions of the data phase symbol transitions $\Delta\phi_{k+1}$, we must further average Eq. (6-58) over the uniformly distributed statistics of this RV in the same manner as we did previously in arriving at $\bar{h}^{\pm}$. The difference here is that $h^{\pm}$ are embedded as arguments of the hypergeometric function, and thus the average cannot be obtained in closed form. Nevertheless, the appropriate modification of Eq. (6-58) now becomes

$$E\{\Lambda^n\} = \frac{\Gamma(N+n)\,\Gamma(N-n)}{\Gamma^2(N)} \overline{{}_1F_1(-n;N;-Nh^+R)\,{}_1F_1(n;N;-Nh^-R)}^{\Delta\phi} \tag{6-59}$$

where, for M-PSK, $\Delta\phi$ takes on values $2k\pi/M, k = 0, 1, 2, \cdots, M-1$, each with probability $1/M$.

For small symbol timing offset, i.e., $\hat{\xi}$ small, Eqs. (6-56) and (6-57) can be simply approximated by

$$E\left\{\hat{R}\right\} = \left(\frac{1}{\hat{\bar{h}}^+}\right)\left[-1 + E\{\Lambda\} + \hat{\xi}\left(E\left\{\Lambda^2\right\} - E\{\Lambda\}\right)\right] \tag{6-60}$$

$$\mathrm{var}\left\{\hat{R}\right\} = \left(\frac{1}{\hat{\bar{h}}^+}\right)^2\left[\left(1-2\hat{\xi}\right) \times \mathrm{var}\{\Lambda\} + 2\hat{\xi}\left(E\left\{\Lambda^3\right\} - E\{\Lambda\}\,E\left\{\Lambda^2\right\}\right)\right]$$

and thus only the first few moments of Λ need be evaluated.

6.6 A Generalization of the SSME Offering Improved Performance

In this section, we consider a modification of the SSME structure that provides improved performance in the sense of lowering the variance of the SNR estimator. To simplify matters, we begin the discussion by considering the ideal case of no frequency uncertainty. Also, for the sake of brevity, we investigate only the I&D version. Suffice it to say that the generalization is readily applied to the sampled version in an obvious manner.

To motivate the search for an SSME structure with improved performance, we define a measure of "quality" of the SNR estimator by its own SNR, namely, $Q = \left(E\{\hat{R}\}\right)^2/\mathrm{var}\{\hat{R}\}$. For large N and large R, we have from Eq. (6-24) that $E\{\hat{R}\} = R, \mathrm{var}\{\hat{R}\} = R^2/N$, and thus $Q = N$. Thus, we observe that for fixed

observation time, the quality of the conventional SSME does not continue to improve as the true SNR, R, increases, but instead saturates to a fixed value. With this in mind, we seek to modify the SSME such that for a fixed observation time the quality of the estimator continues to improve with increasing SNR.

Suppose now that instead of subdividing each data symbol interval T into two halves, we subdivide it into $2L$ subintervals of equal length $T/(2L)$ and use the integrations of the complex-valued received signal plus noise in successive pairs of these intervals to form the SNR estimator. In effect, we are estimating the symbol SNR of a data sequence at L times the actual data rate. This data sequence is obtained by repeating each original data symbol L times to form L consecutive shorter symbols, and thus it is reasonable to refer to L as an *oversampling factor*. For a given total observation time (equivalently, a given total number of original symbols N), there are LN short symbols corresponding to the higher data rate, and their symbol SNR is $r = R/L$. Since the SSME is completely independent of the data sequence, the new estimator, denoted by $\hat{r}_L$, is just an SSME of the SNR $r = R/L$ of the short symbols, based on observing LN short symbols, each split into half. Thus, the mean and variance of $\hat{r}_L$ are computed by simply replacing N by LN and R by R/L in Eq. (6-24), which is rewritten here for convenience as

$$\begin{aligned} E\left\{\hat{R}\right\} &= R + \frac{R+1}{N-1} \\ \operatorname{var}\left\{\hat{R}\right\} &= \frac{1}{N-2}\left(\frac{N}{N-1}\right)^2\left[(2+4R)\left(\frac{N-1/2}{N}\right)+R^2\right] \end{aligned} \tag{6-61}$$

Since, however, we desire an estimate of R, not $r = R/L$, we define $\hat{R}_L = L\hat{r}_L$ and write the corresponding expressions for the mean and variance of $\hat{R}_L$:

$$\begin{aligned} E\left\{\hat{R}_L\right\} &= L\left[\frac{R}{L} + \frac{R/L+1}{LN-1}\right] = R + \frac{R+L}{LN-1} \\ \operatorname{var}\left\{\hat{R}_L\right\} &= \frac{L^2}{LN-2}\left(\frac{LN}{LN-1}\right)^2\left[\left(2+\frac{4R}{L}\right)\left(\frac{LN-1/2}{LN}\right)+\left(\frac{R}{L}\right)^2\right] \end{aligned} \tag{6-62}$$

With this notation, the original SSME is simply $\hat{R} = \hat{R}_1$, and the performance expressions in Eq. (6-62) are valid for any positive integer $L \in \{1, 2, 3, \cdots\}$. For large N, i.e., $N \gg 1$, the mean and variance in Eq. (6-62) simplify within $O(1/N^2)$ to

$$E\left\{\hat{R}_L\right\} = R + \frac{R+L}{LN}$$
$$\text{var}\left\{\hat{R}_L\right\} = \frac{L}{N}\left(2 + \frac{4R}{L} + \frac{R^2}{L^2}\right) \tag{6-63}$$

For the remainder of this section, we base our analytic derivations on the asymptotic expressions in Eq. (6-63).

For small enough R, we can ignore the R and R^2 terms in the variance expression, and the smallest estimator variance is achieved for $L = 1$. In this case, $\hat{R} = \hat{R}_1$ outperforms (has smaller variance than) $\hat{R}_L$ for $L > 1$, approaching a $10\log_{10} L$ dB advantage as $R \to 0$. However, at large enough R for any fixed L, the reverse situation takes place. In particular, retaining only the R^2 term in Eq. (6-63) for sufficiently large R/L, we see that $\hat{R}_L$ offers a $10\log_{10} L$ dB advantage over $\hat{R}$ in this limit. This implies that for small values of R, a half-symbol SSME (i.e., $L = 1$) is the preferred implementation, whereas beyond a certain critical value of R (to be determined shortly) there is an advantage to using values of $L > 1$. In general, for any given R, there is an optimum integer $L = L_*(R)$ that minimizes the variance in Eq. (6-63). We denote the corresponding optimum estimator by $\hat{R}_*$. We show below that, unlike the case of the estimator $\hat{R}_L$ defined for a fixed L, the optimized estimator $\hat{R}_*$ requires proportionally more subdivisions of the true symbol interval as R gets large. As a result, the R^2/L^2 term in Eq. (6-63) does not totally dominate the variance for $R \gg L$, and the amount of improvement at high SNR differs from the $10\log_{10} L$ dB improvement calculated for an arbitrary fixed choice of L and $R \gg L$.

For the moment we ignore the fact that L must be an integer, and minimize the variance expression in Eq. (6-63) over continuously varying *real-valued* L. We define an optimum real-valued $L = L_\bullet(R)$, obtained by differentiating the variance expression of Eq. (6-63) with respect to L and equating the result to zero, as

$$L_\bullet(R) = \frac{R}{\sqrt{2}} \tag{6-64}$$

and a corresponding *fictitious* SNR estimator $\hat{R}_\bullet$ that "achieves" the minimum variance calculated by substituting Eq. (6-64) into the asymptotic variance expression of Eq. (6-63),

$$\text{var}\left\{\hat{R}_\bullet\right\} = \frac{R}{N}\left(4 + 2\sqrt{2}\right) \tag{6-65}$$

The minimum variance shown in Eq. (6-65) can be achieved only by a realizable estimator for values of R that yield an integer $L_\bullet(R)$ as defined by Eq. (6-64). Nevertheless, it serves as a convenient benchmark for comparisons with results corresponding to the optimized realistic implementation $\hat{R}_*$. For example, from Eqs. (6-63) and (6-65) we see that the ratio of the asymptotic variance achieved by any given realizable estimator $\hat{R}_L$ to that achieved by the fictitious estimator $\hat{R}_\bullet$ is a simple function of the short symbol SNR r, not of R and L separately. In particular,

$$\frac{\text{var}\left\{\hat{R}_L\right\}}{\text{var}\left\{\hat{R}_\bullet\right\}} = \frac{2/r + 4 + r}{4 + 2\sqrt{2}} \tag{6-66}$$

The numerator of Eq. (6-66) is a convex $\cup$ function of r, possessing a unique minimum at $r = \sqrt{2}$, at which point the ratio in Eq. (6-66) evaluates to unity. This result is not surprising since, from Eq. (6-64), $r = \sqrt{2}$ is the optimality condition defining the fictitious estimator $\hat{R}_\bullet$. For $r > \sqrt{2}$ or $r < \sqrt{2}$, the ratio in Eq. (6-66) for any fixed value of L grows without bound.

Before going on, let us examine how allowing L to vary with R in an optimum fashion in accordance with Eq. (6-64) has achieved the improvement in "quality" we previously set out to obtain. In particular, since for large N and large R we have $E\{R_\bullet\} = R$ and from Eq. (6-65) $\text{var}\{\hat{R}_\bullet\} = (R/N)\left(4 + 2\sqrt{2}\right)$, then it immediately follows that $Q = \left(E\{\hat{R}\}\right)^2/\text{var}\{\hat{R}\} = NR/(4 + 2\sqrt{2})$, which demonstrates that, for a fixed observation time, the quality of the estimator now increases linearly with true SNR.

We return now to the realistic situation where L must be an integer, but can vary with R or r. Since the variance expression in Eq. (6-63) is convex $\cup$ in L, we can determine whether $\hat{R}_L$ is optimum for a given R by simply comparing its performance to that of its nearest neighbors, $\hat{R}_{L-1}$ and $\hat{R}_{L+1}$. We find that $\hat{R}_L$ is optimum over a continuous range $R \in \left[R_L^-, R_L^+\right]$, where $R_1^- = 0$, $R_{L+1}^- = R_L^+$, and the upper boundary point is determined by equating the variance expressions in Eq. (6-63) for $\hat{R}_L$ and $\hat{R}_{L+1}$:

$$R_L^+ = \sqrt{2L\,(L+1)} \tag{6-67}$$

Thus, the optimum integer $L_*(R)$ is evaluated as

$$L_*(R) = L, \quad \text{if} \quad \sqrt{2L\,(L-1)} \le R \le \sqrt{2L\,(L+1)} \tag{6-68}$$

In particular, we see that $\hat{R}_1$ is optimum in the region $0 \leq R \leq 2$, implying no improvement over the original SSME for these values of R. For values of R in the region $2 \leq R < 2\sqrt{3}$, one should use $\hat{R}_2$ (i.e., an estimator based on pairs of quarter-symbol integrations), and in general one should use $\hat{R}_L$ when $\sqrt{2L\,(L-1)} \leq R \leq \sqrt{2L\,(L+1)}$. For R in this interval, the improvement factor $I(R)$ (reduction in variance) achieved by the new optimized estimator relative to the conventional half-symbol SSME $\hat{R} = \hat{R}_1$ is calculated as

$$I(R) = \frac{\text{var}\left\{\hat{R}\right\}}{\text{var}\left\{\hat{R}_*\right\}} = \frac{2 + 4R + R^2}{L\left(2 + \dfrac{4R}{L} + \dfrac{R^2}{L^2}\right)}, \quad \sqrt{2L\,(L-1)} \leq R \leq \sqrt{2L\,(L+1)} \tag{6-69}$$

We have already seen that $I(R) = 1$ for R ranging from 0 to 2, whereupon it becomes better to use $\hat{R}_2$, allowing $I(R)$ to increase monotonically to a value of $\left(7+4\sqrt{3}\right)/\left(5+4\sqrt{3}\right) = 1.168$ (equivalent to 0.674 dB) at $R = 2\sqrt{3}$. Continuing on, in the region $2\sqrt{3} \leq R < 2\sqrt{6}$, one should use $\hat{R}_3$, whereupon $I(R)$ continues to increase monotonically to a value of $\left(13+4\sqrt{6}\right)/\left(7+4\sqrt{6}\right) = 1.357$ (equivalent to 1.326 dB) at $R = 2\sqrt{6}$. Figure 6-8 is a plot of $I(R)$ versus R, as determined from Eq. (6-69). Note that while $I(R)$ is a continuous function

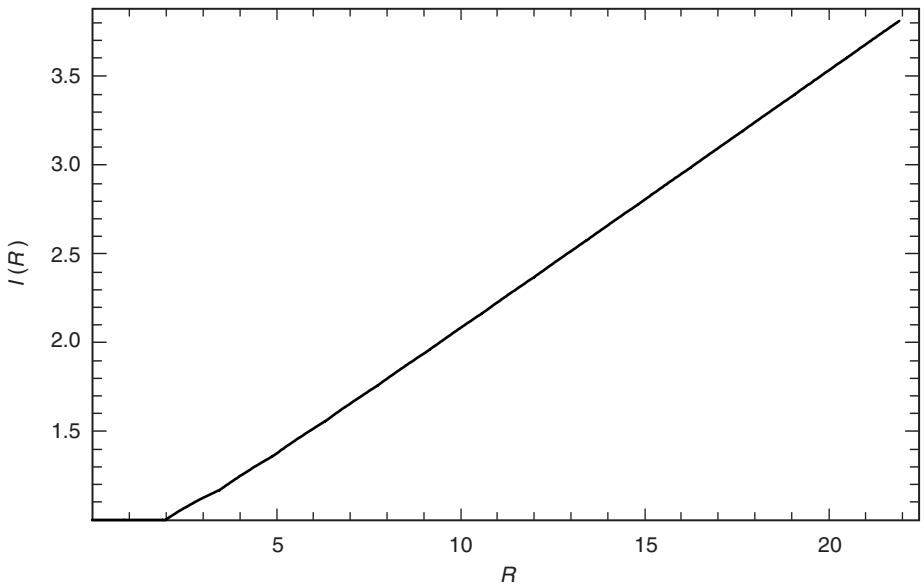


Fig. 6-8. Performance improvement as a function of SNR.

of R, the derivative of $I(R)$ with respect to R is discontinuous at the critical values of R, namely, $R = R_L^+$ for $L \in \{1, 2, 3, \cdots\}$, but the discontinuity becomes monotonically smaller as L increases.

It is also instructive to compare the performance of the optimized realizable estimator $\hat{R}_*$ with that of the fictitious estimator $\hat{R}_\bullet$. The corresponding variance ratio is computed directly from Eq. (6-66), as long as we are careful to delineate the range of validity from Eq. (6-68), where each integer value of L contributes to the optimized estimator $\hat{R}_*$:

$$\frac{\text{var}\left\{\hat{R}_*\right\}}{\text{var}\left\{\hat{R}_\bullet\right\}} = \frac{2/r + 4 + r}{4 + 2\sqrt{2}}, \quad \sqrt{1 - 1/L_*(R)} \le \frac{r}{\sqrt{2}} \le \sqrt{1 + 1/L_*(R)} \tag{6-70}$$

where for the optimized realizable estimator $\hat{R}_*$ the short symbol SNR r is evaluated explicitly in terms of R as $r = R/L_*(R)$. We see that for any value of R the corresponding interval of validity in Eq. (6-70) always includes the optimal point $r = \sqrt{2}$, at which the ratio of variances is unity. Furthermore, since the width of these intervals (measured in terms of r) shrinks to zero as $L_*(R) \to \infty$, the ratio of variances makes smaller and smaller excursions from its value of unity at $r = \sqrt{2}$ as $R \to \infty$, implying $L_*(R) \to \infty$ from Eq. (6-68). Thus, the asymptotic performance for large R and large N of the optimized realizable estimator $\hat{R}_*$ is the same as that of the fictitious estimator $\hat{R}_\bullet$ given in Eq. (6-65). In particular, we see from Eq. (6-65) that $\text{var}\{\hat{R}_*\}$ grows only linearly in the limit of large R, whereas $\text{var}\{\hat{R}_L\}$ for any fixed L eventually grows quadratically for large enough R/L.

As can be seen from Eq. (6-63), the generalized SSME $\hat{R}_L$ is asymptotically unbiased (in the limit as $N \to \infty$). As shown in [6], it is possible to completely remove the bias of the conventional SSME $\hat{R}$ and to define a perfectly unbiased estimator as $\hat{R}^o = \hat{R} - (\hat{R} + 1)/N$. Similarly, we can now define a precisely unbiased version $\hat{R}_L^o$ of our generalized estimator $\hat{R}_L$ by

$$\hat{R}_L^o = \hat{R}_L - \frac{\hat{R}_L + L}{LN} \tag{6-71}$$

Again we note that the original unbiased SSME $\hat{R}^o$ is just a special case of our generalized unbiased SSME, $\hat{R}^o = \hat{R}_1^o$. Using the definition of Eq. (6-71) together with the expressions in Eq. (6-62) for the exact mean and variance of $\hat{R}_L$, we find that the exact mean and variance of the unbiased estimator $\hat{R}_L^o$ are given by

$$E\left\{\hat{R}_L^o\right\} = R$$

$$\text{var}\left\{\hat{R}_L^o\right\} = \frac{L^2}{LN-2}\left[\left(1+\frac{4R}{L}\right)\left(\frac{LN-1/2}{LN}\right)+\left(\frac{R}{L}\right)^2\right] \tag{6-72}$$

For large N, the asymptotic variance expression obtained from Eq. (6-72) is identical to that already shown in Eq. (6-63) for the biased estimator. Thus, all of the preceding conclusions about the optimal choice of L for a given R, and the resulting optimal estimator performance, apply equally to the unbiased versions $\hat{R}_L^o$ of the estimators $\hat{R}_L$.

6.7 A Method for Improving the Robustness of the Generalized SSME

For any fixed L, our generalized SSME $\hat{R}_L$ is only optimal when the true SNR R lies in the range $\sqrt{2L\,(L-1)} \le R \le \sqrt{2L\,(L+1)}$. Indeed $\hat{R}_L$ for any $L > 1$ is inferior to the original SSME $\hat{R}_1$ for small enough R (at least for $0 \le R \le 2$). The range of optimality for a given value of L, measured in decibels, is just $10\log_{10}\left[\sqrt{2L(L+1)}/\sqrt{2L(L-1)}\,\right] = 5\log_{10}\left[(L+1)/(L-1)\right]$ dB, which diminishes rapidly toward 0 dB with increasing L. In order to achieve the exact performance of the optimized estimator $\hat{R}_*$ over an unknown range of values of the true SNR R, one would need to select, and then implement, the optimal symbol subdivision based on arbitrarily precise knowledge (measured in decibels) of the very parameter being estimated! Fortunately, there is a more robust version of the generalized SSME that achieves nearly the same performance as $\hat{R}_*$, yet requires only very coarse knowledge about the true SNR R.

To define the robust generalized SSME, we use the same set of estimators $\left\{\hat{R}_L\right\}$ as defined before for any fixed integers L, but now we restrict the allowable choices of L to the set of integers $\{b^\ell, \ell = 0, 1, 2, \cdots\}$, for some integer base $b \ge 2$. The optimal choice of L restricted to this set is denoted by $L_{b*}(R)$, and the corresponding optimized estimator is denoted by $\hat{R}_{b*}$. Because our various estimators differ only in the amount of freedom allowed for the choice of L, their performances are obviously related as

$$\text{var}\left\{\hat{R}_\bullet\right\} \le \text{var}\left\{\hat{R}_*\right\} \le \text{var}\left\{\hat{R}_{b*}\right\} \le \text{var}\left\{\hat{R}_1\right\} \tag{6-73}$$

In this section, we will show analytically that the variance achieved by the robust estimator $\hat{R}_{b*}$ with $b = 2$ comes very close to that achieved by the fictitious estimator $\hat{R}_\bullet$ for all $R \ge 2$, and hence Eq. (6-73) implies that for this range

of R it must be even closer to the less analytically tractable variance achieved by the optimized realizable estimator $\hat{R}_*$. Conversely, for all $R \leq 2$, we have already seen that the optimized realizable estimator $\hat{R}_*$ is the same as the original SSME $\hat{R}_1$, and hence so is the optimized robust estimator $\hat{R}_{b*}$ for any b, since $L = b^0 = 1$ is a permissible value for the robust estimator as well.

The convexity of the general asymptotic variance expression in Eq. (6-63) again allows us to test the optimality of $\hat{R}_{b^\ell}$ by simply comparing its performance versus that of its nearest permissible neighbors, $\hat{R}_{b^{\ell-1}}$ and $\hat{R}_{b^{\ell+1}}$. The lower and upper endpoints of the region of optimality for any particular $\hat{R}_{b^\ell}$ are determined by equating $\text{var}\{\hat{R}_{b^\ell}\}$ with $\text{var}\{\hat{R}_{b^{\ell-1}}\}$ and $\text{var}\{\hat{R}_{b^{\ell+1}}\}$, respectively. This leads to the following definition of the optimal $L_{b*}(R)$ for L restricted to the set $\{b^\ell, \ell = 0, 1, 2, \cdots\}$:

$$L_{b*}(R) = \begin{cases} b^\ell, & \text{if } \sqrt{2b^{2\ell-1}} \leq R \leq \sqrt{2b^{2\ell+1}} \text{ for integer } \ell \geq 1 \\ b^0 = 1, & \text{if } 0 \leq R \leq \sqrt{2b} \end{cases} \tag{6-74}$$

For all $R \leq \sqrt{2b}$, the optimized estimator $\hat{R}_{b*}$ is the same as the original SSME $\hat{R}_1$. For all $R \geq \sqrt{2/b}$ (which includes the upper portion of the interval over which $l = 0$ is optimum), the variance achieved by $\hat{R}_{b*}$, normalized to that of the fictitious estimator $\hat{R}_\bullet$, is obtained from Eqs. (6-66) and (6-74) in terms of $r = R/L_{b*}(R)$, and upper bounded by

$$\frac{\text{var}\{\hat{R}_{b*}\}}{\text{var}\{\hat{R}_\bullet\}} = \frac{2/r + 4 + r}{4 + 2\sqrt{2}} \leq \frac{4 + \sqrt{2}\left(\sqrt{b} + 1/\sqrt{b}\right)}{4 + 2\sqrt{2}}, \quad \frac{1}{\sqrt{b}} \leq \frac{r}{\sqrt{2}} \leq \sqrt{b} \tag{6-75}$$

As with the earlier expression of Eq. (6-70) for the variance of $\hat{R}_*$, the intervals of validity in Eq. (6-75) for any value of R always include the optimal point $r = \sqrt{2}$ at which the ratio of variances is unity. But unlike Eq. (6-70), the width of the intervals in Eq. (6-75) stays constant independently of r. The upper limit on the variance ratio shown in Eq. (6-75) occurs at the end points of these intervals, i.e., for SNR values expressible as $R = \sqrt{2b^{2\ell-1}}$ for some integer $\ell \geq 0$. This upper limit is the maximum excursion from unity of the variance ratio for all $R \geq \sqrt{2/b}$. For all $R \leq 2$ and any $b \geq 2$, there is no limit on the suboptimality of $\hat{R}_{b*}$ with respect to the fictitious estimator $\hat{R}_\bullet$, but in this range $\hat{R}_{b*}$ suffers no suboptimality with respect to the optimized realizable estimator $\hat{R}_*$, since

both are equivalent to the original SSME $\hat{R}_1$ for $R \leq 2$. Finally, reiterating our earlier conclusion based on the simple inequalities in Eq. (6-73), we conclude that the maximum degradation $D(R)$ of the robust estimator $\hat{R}_{b*}$ with respect to the optimized realizable estimator $\hat{R}_*$ is upper bounded for all R by

$$D(R) = \frac{\text{var}\left\{\hat{R}_{b*}\right\}}{\text{var}\left\{\hat{R}_*\right\}} \leq \frac{\text{var}\left\{\hat{R}_{b*}\right\}}{\text{var}\left\{\hat{R}_\bullet\right\}} \leq \frac{4 + \sqrt{2}\left(\sqrt{b} + 1/\sqrt{b}\right)}{4 + 2\sqrt{2}} \quad \text{for all } R \tag{6-76}$$

For example, we consider the case of $b = 2$, which yields permissible values of L given by $L = 1, 2, 4, 8, 16, \cdots$ and corresponding decision region boundaries at $R = 1, 2, 4, 8, 16, \cdots$, i.e., regions separated by 3 dB. From Eq. (6-76), the maximum degradation $D_{\max}$ for using the coarsely optimized estimator $\hat{R}_{2*}$ instead of the fully optimized realizable estimator $\hat{R}_*$ is no more than

$$D_{\max} \leq \frac{7}{4 + 2\sqrt{2}} = 1.02513 \tag{6-77}$$

i.e., a penalty of only 2.5 percent. Even if we were to enlarge the regions of constant $L_{b*}(R)$ to a width of 9 dB in R (corresponding to $b = 8$), the maximum penalty would increase only to

$$D_{\max} \leq \frac{8.5}{4 + 2\sqrt{2}} = 1.245 \tag{6-78}$$

i.e., a penalty just under 25 percent. Thus, even though the optimized generalized SSME $\hat{R}_*$ requires (in principle) very precise prior knowledge of the true value of R, its performance can be reasonably well approximated by that of a robust estimator $\hat{R}_{b*}$ requiring only a very coarse prior estimate of R.

6.8 Special Case of the SSME for BPSK-Modulated Data

We can define an analogous sequence of generalized SSMEs $\{\tilde{R}_L, L = 1, 2, \cdots\}$ corresponding to the original SSME $\tilde{R} = \tilde{R}_1$ developed for BPSK signals using real-valued in-phase samples only. In this case, the (exact) mean and variance of the original SSME $\tilde{R}$ are given by [4]

$$E\left\{\tilde{R}\right\} = R + \frac{2R+1}{N-2}$$
$$\text{var}\left\{\tilde{R}\right\} = \frac{1}{N-4}\left(\frac{N}{N-2}\right)^2\left[(1+4R)\left(\frac{N-1}{N}\right)+2R^2\right] \tag{6-79}$$

The mean and variance of the generalized SSME $\tilde{R}_L$ based on real-valued samples are obtained from Eq. (6-79) by following the same reasoning that led to Eq. (6-62):

$$E\left\{\tilde{R}_L\right\} = L\left[\frac{R}{L}+\frac{2R/L+1}{LN-2}\right] = R + \frac{2R+L}{LN-2}$$
$$\text{var}\left\{\tilde{R}_L\right\} = \frac{L^2}{LN-4}\left(\frac{LN}{LN-2}\right)^2\left[\left(1+\frac{4R}{L}\right)\left(\frac{LN-1}{LN}\right)+2\left(\frac{R}{L}\right)^2\right] \tag{6-80}$$

and the asymptotic forms for large N, i.e., $N \gg 1$, are within $O(1/N^2)$ of

$$E\left\{\tilde{R}_L\right\} = R + \frac{2R+L}{LN}$$
$$\text{var}\left\{\tilde{R}_L\right\} = \frac{L}{N}\left[1+4\left(\frac{R}{L}\right)+2\left(\frac{R}{L}\right)^2\right] \tag{6-81}$$

We can argue as in [5] that the first- and second-order statistics of the SSME $\hat{R}_L$ based on complex samples are derivable from those of the SSME $\tilde{R}_L$ based on real samples. Specifically, since $\hat{R}_L$ is obtained from twice as many real observables as $\tilde{R}_L$, with (on average) only half the SNR (since the SNR is zero in the quadrature component for BPSK signals), we have the following (exact) equalities:

$$E\left\{\frac{\hat{R}_L}{2}\right\}\Big|_{(R,N)} = E\left\{\tilde{R}_L\right\}\Big|_{([R/2],2N)}$$
$$\text{var}\left\{\frac{\hat{R}_L}{2}\right\}\Big|_{(R,N)} = \text{var}\left\{\tilde{R}_L\right\}\Big|_{([R/2],2N)} \tag{6-82}$$

where now we have explicitly denoted the dependence of $\hat{R}_L$ and $\tilde{R}_L$ on the SNR and the number of symbols. The equalities in Eq. (6-82) can be verified by direct comparison of Eq. (6-80) with Eq. (6-62) and Eq. (6-81) with Eq. (6-63).

As in our earlier analysis of the generalized SSME $\hat{R}_L$ based on complex-valued samples, we can also optimize the generalized SSME $\tilde{R}_L$ based on real-valued samples with respect to its asymptotic performance expressions in Eq. (6-81). We define for any fixed value of R an optimum integer $L = \tilde{L}_*(R)$ and an optimum real number $L = \tilde{L}_\bullet(R)$ to minimize the asymptotic variance expression in Eq. (6-81), and corresponding optimal realizable and fictitious estimators $\tilde{R}_*$ and $\tilde{R}_\bullet$. For the optimum realizable estimate, we find, corresponding to Eq. (6-68), that the optimum integer $\tilde{L}_*(R)$ is evaluated as

$$\tilde{L}_*(R) = L, \quad \text{if} \quad \sqrt{L\,(L-1)\,/2} \le R \le \sqrt{L\,(L+1)\,/2} \tag{6-83}$$

We find, corresponding to Eqs. (6-64) and (6-65), that the optimal real value of L is $\tilde{L}_\bullet(R) = R\sqrt{2}$ and the corresponding variance is

$$\text{var}\left\{\tilde{R}_\bullet\right\} = \frac{R}{N}\left(4 + 2\sqrt{2}\right) = \text{var}\left\{\hat{R}_\bullet\right\} \tag{6-84}$$

In other words, the fictitious estimator achieves identical variance using either real samples or complex samples.

Finally, we observe from a comparison of Eqs. (6-62) and (6-80) an interesting (exact) relationship between the means and variances of the two generalized SSMEs for different values of the symbol rate oversampling factor L:

$$\begin{aligned} E\left\{\hat{R}_L\right\} &= E\left\{\tilde{R}_{2L}\right\} \\ \text{var}\left\{\hat{R}_L\right\} &= \text{var}\left\{\tilde{R}_{2L}\right\} \end{aligned} \tag{6-85}$$

Thus, the estimators $\tilde{R}_L$ based on real samples can be viewed as a more finely quantized sequence than the estimators $\hat{R}_L$ based on complex samples, in that any mean and variance achievable by an estimator in the latter sequence is also achievable by taking twice as many subintervals in a corresponding estimator from the former sequence. This implies, for example, that the maximum deviation of the variances of $\tilde{R}_*$ and $\tilde{R}_\bullet$ is no greater than that calculated in Eq. (6-70) for the deviation between the variances of $\hat{R}_*$ and $\hat{R}_\bullet$.

6.9 Comparison with the Cramer–Rao Lower Bound on the Variance of SNR Estimators

A good benchmark for the performance of a given SNR estimator is the Cramer–Rao (C-R) lower bound on its variance [11]. Here we present for comparison the C-R lower bound for any SNR estimator using a given number of observables (samples) per symbol interval, with or without knowledge of the data. For simplicity, we consider only estimators based on real observables, since a number of C-R bounds reported elsewhere [1,12,13] have explicitly considered that case.

It has been shown in [13] that the C-R lower bound on the variance of an arbitrary unbiased estimator of SNR, R^*, in the presence of unknown binary equiprobable data and K independent real observations per symbol (K subinterval samples) is given by

$$\operatorname{var}\{R^*\} \geq \frac{2R^2}{N}\left[\frac{2K + 2R - E_2(2R)}{2KR - (4R + K)\,E_2(2R)}\right] \tag{6-86}$$

where

$$E_2(2R) = E\left\{X^2 \operatorname{sech}^2 X\right\} \tag{6-87}$$

with X a Gaussian random variable with mean and variance both equal to $2R$. The expectation in Eq. (6-87), which depends only on R, cannot be determined in closed form but is easily evaluated numerically. Figure 6-9, described at the end of this section, compares the C-R bounding variance in Eq. (6-86) with the actual asymptotic variance in Eq. (6-81) achieved by the generalized SSME $\tilde{R}_L$ based on real samples. For this comparison, we substitute $K = 2L$ in the C-R bound expression (because there are $K = 2L$ subinterval integrations contributing to the SSME $\tilde{R}_L$), and we plot the cases $L = 1, 2, 4, \infty$.

We can also perform analytic comparisons in the limits of low and high SNR. The low- and high-SNR behavior of the C-R bounding variance in Eq. (6-86) is given by [13]

$$\operatorname{var}\{R^*\} \geq \begin{cases} \dfrac{1}{2N}\left(\dfrac{K}{K-1}\right), & R \ll 1 < K \\ \dfrac{2R}{N}\left(1 + \dfrac{R}{K}\right), & R \gg K \end{cases} \tag{6-88}$$

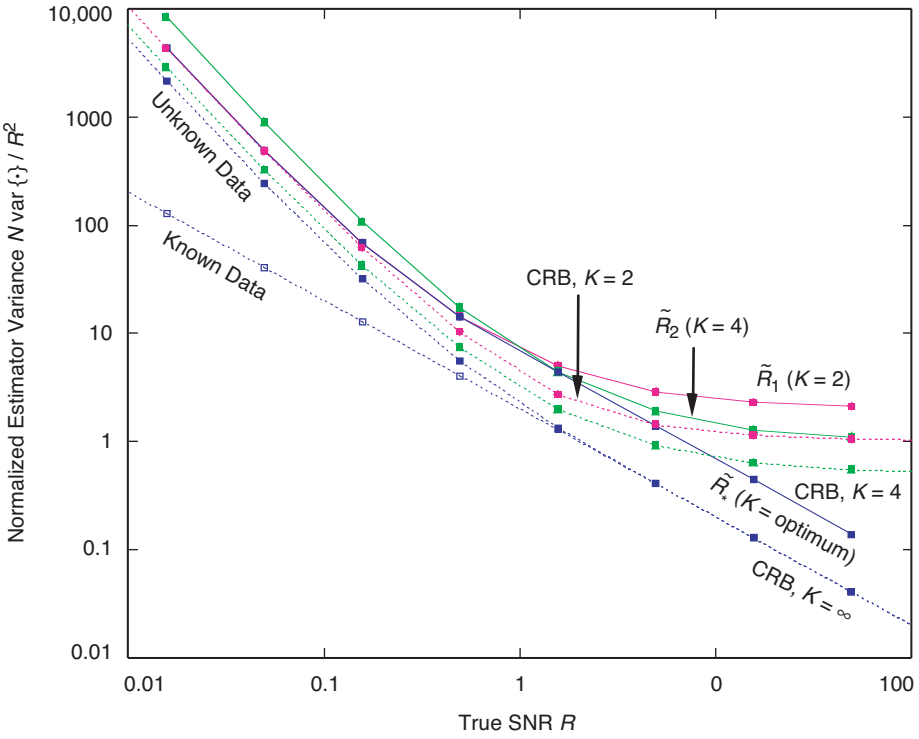


Fig. 6-9. A comparison of the performance of several SNR estimators with the Cramer–Rao bound.

By comparison, the asymptotic expression in Eq. (6-81) for the variance of $\tilde{R}_L$ for any fixed L reduces in the low- and high-SNR limits to

$$\text{var}\left\{\tilde{R}_L\right\} \cong \begin{cases} \dfrac{L}{N} = \dfrac{K}{2N}, & L \gg R \\ \dfrac{2R^2}{NL}\left(1 + \dfrac{2L}{R}\right) = \dfrac{4R}{N}\left(1 + \dfrac{R}{K}\right), & R \gg L \end{cases} \tag{6-89}$$

Compared to the C-R bounding variance in Eq. (6-88), the actual variance in Eq. (6-89) is higher by a factor of $K - 1$ in the low-SNR limit and by a factor of two in the high-SNR limit.

For any fixed K, the C-R bounding variance in Eq. (6-86) becomes quadratic in R as R approaches infinity, as evidenced by the second expression in Eq. (6-89). On the other hand, the limiting behavior of the bound for K approaching infinity with fixed R is given by

$$\mathrm{var}\left\{R^*\right\} \geq \frac{1}{N}\left[\frac{4R^2}{2R - E_2\left(2R\right)}\right], \quad K \gg \max\left(R, 1\right) \tag{6-90}$$

Since $E_2(2R) = 2R + 8R^2 + O\left(R^3\right)$ for small R and is exponentially small for large R [13], the C-R bounding variance on the right side of Eq. (6-90) approaches a constant at low SNR and becomes linear in R at high SNR:

$$\mathrm{var}\left\{R^*\right\} \geq \begin{cases} \dfrac{1}{2N}, & K \gg 1 \gg R \\ \dfrac{2R}{N}, & K \gg R \gg 1 \end{cases} \tag{6-91}$$

Since the C-R bounding expressions in Eqs. (6-90) and (6-91) for large values of $K = 2L$ reflect the best possible performance of an estimator with access to a continuum of samples within each symbol, they are suitably compared to the performance of the optimized estimator $\tilde{R}_*$, rather than to the performance of $\tilde{R}_L$ for any fixed L. As an approximation to $\tilde{R}_*$, we use a stand-in estimator equal to $\tilde{R}_1$ for $R \leq 2$ (i.e., where $\tilde{L}_*(R) = 1$) and to the fictitiously optimized estimator $\tilde{R}_\bullet$ for $R > 2$. The corresponding asymptotic variances computed from Eq. (6-81) for the limits corresponding to those in Eq. (6-91) are

$$\begin{cases} \mathrm{var}\left\{\tilde{R}_1\right\} = \dfrac{1}{N}, & 1 \gg R \\ \mathrm{var}\left\{\tilde{R}_\bullet\right\} = \dfrac{R}{N}\left(4 + 2\sqrt{2}\right), & R \gg 1 \end{cases} \tag{6-92}$$

The estimator variances in Eq. (6-92) are higher than the corresponding C-R bounding variances in Eq. (6-91) by a factor of 2 in the low-SNR limit and by a factor of $2 + \sqrt{2} \cong 3.4$ in the high-SNR limit. The optimized *realizable* estimator $\tilde{R}_*$ suffers an additional small suboptimality factor with respect to the performance of the fictitious estimator $\tilde{R}_\bullet$ used as its stand-in in Eq. (6-92).

Finally we consider for purposes of comparison the C-R bound on an arbitrary unbiased estimator when the data are perfectly known. The C-R bound under this assumption is well known, e.g., [11]. Here we continue with the notation of [13] by noting that the derivation there for the case of unknown data is easily modified to the known data case by skipping the average over the binary equiprobable data. The result is equivalent to replacing the function $E_2\left(2R\right)$ by zero in the C-R bound expression in Eq. (6-41), i.e.,

$$\mathrm{var}\left\{\hat{R}\right\} \geq \frac{2R^2}{N}\left[\frac{2K + 2R}{2KR}\right] = \frac{2R}{N}\left(1 + \frac{R}{K}\right), \quad \text{for all } K, R \tag{6-93}$$

We compare this bound for known data, which is valid for all K and R, with the high-SNR bound for unknown data given by the second expression in Eq. (6-88), which is valid for any fixed K as $R \to \infty$. These two variance expressions are identical because the second expression in Eq. (6-88) was obtained from Eq. (6-86) using the approximation that $E_2(2R)$ is exponentially small for large R. Thus, we reach the interesting and important conclusion that, based on the C-R bounds, *knowledge of the data is inconsequential in improving the accuracy of an optimized SNR estimator at high enough SNR*! We also note that the limiting fractional variance, $\mathrm{var}\{R^*\}/(R^*)^2$, in either case is simply $2/(NK)$, i.e., it falls in proportion to the total number NK of samples collected. In this limit, therefore, it does not matter to an optimal estimator whether it collects the same total number of samples in step with the symbol rate or faster. From the second expression in Eq. (6-89), we see that our generalized SSME $\tilde{R}_L$ behaves similarly to an optimum estimator in this respect, because the ratio of its fractional variance to the C-R bounding variance is a constant factor of 2 when $R \gg K$. Whereas with the original SSME one might need to wait N_{req} symbol periods to reach a desired estimator variance, our generalized SSME now offers the capability at high enough SNR to reach this same variance within N_{req}/L symbol periods. Since any practical system will impose limits on the integrate-and-dump rate, and hence on L, this waiting time for acceptable estimator variance cannot be made arbitrarily small. However, at high SNR our generalized SSME allows this waiting time to be reduced down to the limits arising from the system's sampling rate if so desired.

At low SNR, we have seen from the first expression in Eq. (6-88) that the C-R bounding variance (not the fractional variance) for the case of unknown data hits a nonzero floor at $(1/2N)K/(K-1)$ no matter how closely R approaches zero, whereas the bounding variance in Eq. (6-93) for the case of known data goes to zero linearly in R. Thus, knowledge of the data fundamentally changes the behavior of the C-R bound at low SNR, and it can be quite helpful in this region for improving the accuracy of the estimator. Inspection of Eqs. (6-93) and (6-88) in the limit of small R shows that, in contrast to the case for high SNR, oversampling confers no benefit (with known data) or virtually no benefit (with unknown data) to the performance of an optimized estimator at low SNR. Indeed, we see from Eq. (6-89) that the performance of our generalized SSME in this limit is actually worsened by oversampling. Thus, the waiting time to achieve acceptable estimator variance at low SNR is dictated by the symbol rate, even if the system's sampling rate capabilities are significantly faster.

Figure 6-9 summarizes the comparisons of our generalized SSME with the relevant C-R bounds (CRB). This figure plots the CRB as a function of true SNR R, for $K = 2, 4, \infty$ with unknown data, and for $K = \infty$ with known data.

Also shown for comparison are the actual asymptotic variances achieved by the original SSME $\tilde{R}_1$, the generalized SSME $\tilde{R}_2$ using four subinterval integrations within each symbol, and the optimized generalized SSME $\tilde{R}_*$. In each case, the asymptotic variance is plotted in normalized form as $N\text{var}\{\cdot\}/R^2$, which can be interpreted as the number of symbols N that must be observed to achieve a *fractional* estimator variance of 100 percent; smaller fractional variances require inversely proportionately larger numbers of symbols. Also, since asymptotically for large N, $\text{var}\{\cdot\}/R^2$ is an inverse measure of the "quality" of the estimator as previously defined, which for large R varied inversely with N, then asymptotically for large R, $N\text{var}\{\cdot\}/R^2$ is an inverse measure of this quality with the dependence on observation time normalized out.

6.10 Improvement in the Presence of Frequency Uncertainty

Earlier in the chapter and in [6] we considered the performance of the conventional ($L = 1$) SSME in the presence of carrier phase and frequency uncertainties for a variety of cases corresponding to the degree to which the frequency uncertainty is estimated and compensated for. Here we extend these results to the generalized SSME, i.e., we examine the improvement in performance when frequency uncertainty is present, obtained by optimally partitioning the symbol interval in accordance with the value of the true SNR. In the case where the frequency uncertainty is not estimated, one has no choice other than to use the SNR boundaries determined in the no-frequency-uncertainty case, i.e., those given in Eq. (6-68) or Eq. (6-74). For the cases where an estimate of the frequency uncertainty is available, and therefore can be compensated for, one can use this information, if desired, to modify the SNR boundaries. However, to a first-order approximation, we shall assume in what follows that we always determine the boundaries for the symbol regions of fixed partitioning from their zero-frequency uncertainty values. This allows one to implement a fixed-SSME configuration independent of the knowledge of the frequency error and yet still to obtain the possibility of a performance advantage relative to the conventional half-symbol split structure. To illustrate the application of the principles involved and resulting performance gains obtained, we shall consider only a few of the cases previously treated.

•Case 1: Frequency Uncertainty, No Frequency Estimation (and thus No Phase Compensation)

For this case, it was shown earlier that the variance of the conventional SSME is given by Eq. (6-26). To modify this expression for the case of $2L$ partitions

of the symbol interval, we proceed as before by replacing R by R/L, N by LN, δ by δ/L, and then multiplying the result by L^2, resulting in[5]

$$\text{var}\left\{\hat{R}_L\right\} =$$

$$L^2\left(\frac{LN}{LN-1}\right)^2\left\{\left(\frac{LN-1}{LN-2}\right)\left[\frac{\left(1+2h^+\left(\frac{\delta}{L}\right)\frac{R}{L}\right)}{LN}+\left(1+h^+\left(\frac{\delta}{L}\right)\frac{R}{L}\right)^2\right]\right.$$

$$\times\, {}_1F_1\left(2;LN;-Nh^-\left(\frac{\delta}{L}\right)R\right)-\left(1+h^+\left(\frac{\delta}{L}\right)\frac{R}{L}\right)^2$$

$$\left.\times\left[{}_1F_1\left(1;LN;-Nh^-\left(\frac{\delta}{L}\right)R\right)\right]^2\right\} \tag{6-94}$$

Then, the improvement in performance is obtained by taking the ratio of Eq. (6-26) to Eq. (6-94), i.e.,

$$I(R)=\frac{\text{var}\left\{\hat{R}\right\}}{\text{var}\left\{\hat{R}_L\right\}} \tag{6-95}$$

where, for a value of R in the interval $R_L^- \le R < R_L^+$, the value of L to be used corresponds to that determined from Eq. (6-68) or alternatively from Eq. (6-74). We note that since the boundaries of the SNR regions of Eqs. (6-68) and (6-74) are determined from the asymptotic (large N) expressions for the estimator variance, a plot of $I(R)$ versus R determined from Eq. (6-95) will exhibit small discontinuities at these boundaries. These discontinuities will become vanishingly small as N increases.

Figures 6-10 and 6-11 illustrate such a plot for values of N equal to 20 and 100, respectively, with δ as a parameter. We make the interesting observation that, although on an absolute basis the variance of the estimator monotonically improves with increasing N, the improvement factor as evaluated from Eq. (6-95), which makes use of the exact expression for the estimator variance, shows a larger improvement for smaller values of N. To see how this comes about analytically,

[5] To make matters clear, we now include the dependence of $h^\pm$ on δ in the notation.

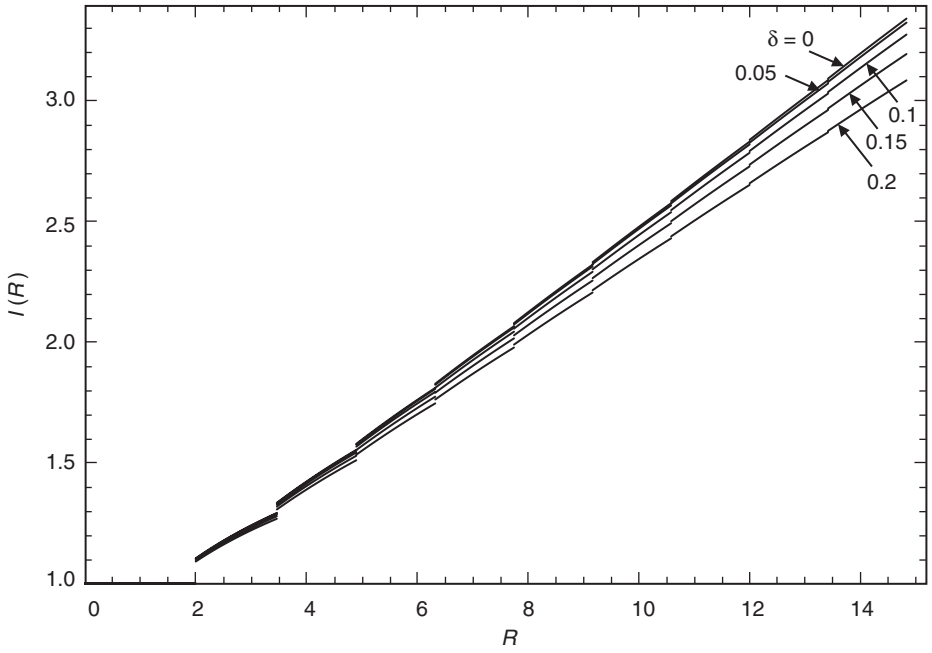


Fig. 6-10. Improvement factor versus SNR with normalized frequency uncertainty as a parameter; Case 1; $N = 20$.

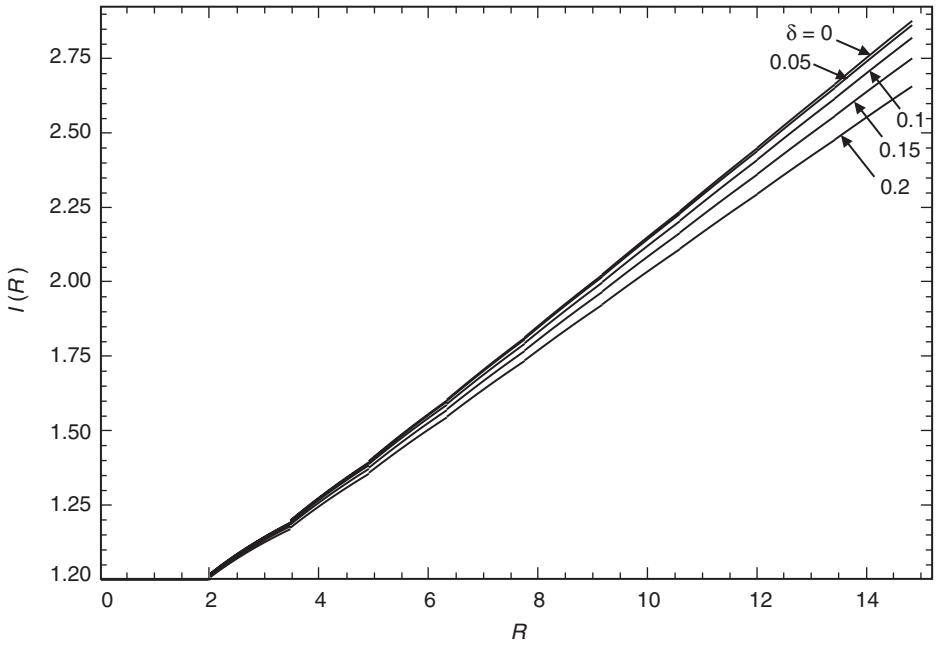


Fig. 6-11. Improvement factor versus SNR with normalized frequency uncertainty as a parameter; Case 1; $N = 100$.

we examine the behavior of the zero-frequency uncertainty improvement factor for large SNR. For sufficiently large SNR (equivalently, large L), we obtain from Eq. (6-62) the same asymptotic expression as given in Eq. (6-63) when assuming N large. Also, since for large SNR L and R are approximately related by $L = R/\sqrt{2}$, then substituting this in Eq. (6-63) gives the asymptotic result

$$\text{var}\left\{\hat{R}_L\right\} \cong \frac{R}{N}\left(4+2\sqrt{2}\right) \tag{6-96}$$

From Eq. (6-61), we have for sufficiently large SNR

$$\text{var}\left\{\hat{R}\right\} = \frac{1}{N-2}\left(\frac{N}{N-1}\right)^2 R^2 \tag{6-97}$$

Thus, the improvement factor for large SNR is the ratio of Eq. (6-97) to Eq. (6-96), namely,

$$I\left(R\right) = \frac{\dfrac{1}{N-2}\left(\dfrac{N}{N-1}\right)^2 R^2}{\dfrac{R}{N}\left(4+2\sqrt{2}\right)} = \frac{R}{4+2\sqrt{2}}\left(\frac{N}{N-2}\right)\left(\frac{N}{N-1}\right)^2 \tag{6-98}$$

which, for a given R, is a monotonically decreasing function of N approaching $I(R) = R/\left(4+2\sqrt{2}\right)$ in the limit as $N \to \infty$.

•Case 2b: Frequency Uncertainty, Perfect Frequency Estimation, Fractional-Symbol Phase Compensation

The case where the frequency uncertainty is perfectly estimated and then used to compensate for the phase shift caused by this uncertainty in the second half of the symbol interval variance of the estimator was given in Eq. (6-30). Making the same substitutions as before, for a $2L$-partition of the symbol interval we obtain

$$\text{var}\left\{\hat{R}_L\right\} =$$

$$L^2 \frac{1}{\left(h^+\left(\delta/L\right)\right)^2}\,\frac{1}{LN-2}\left(\frac{LN}{LN-1}\right)^2\left[\left(1+2h^+\left(\frac{\delta}{L}\right)\frac{R}{L}\right)\left(\frac{2LN-1}{LN}\right)\right.$$

$$\left.+\left(h^+\left(\frac{\delta}{L}\right)\frac{R}{L}\right)^2\right] \tag{6-99}$$

Comparing Eq. (6-30) with Eq. (6-61), we observe that, in this case, the variance of $h^+(\delta)\hat{R}$ for the conventional SSME is identical to the variance of $\hat{R}$ in the zero-frequency uncertainty case. From a comparison of Eqs. (6-99) and (6-62), a similar equivalence can be made between the variance of $h^+(\delta/L)\hat{R}$ and the variance of $\hat{R}$ for the $2L$-partition estimator.

Analogous to what was done for Case 1, the improvement factor, $I(R)$, here can be obtained from the ratio of Eq. (6-30) to Eq. (6-99). Figures 6-12 and 6-13 are plots of $I(R)$ versus true SNR, R, for values of N equal to 20 and 100, respectively, with δ as a parameter. Once again we make the observation that a larger improvement is obtained for smaller values of N. An analytical justification for this observation can be demonstrated by examining the behavior of I for large SNR. Specifically, the analogous expression to Eq. (6-98) now becomes

$$I(R) = \frac{\left(\dfrac{1}{\left(h^+(\delta)\right)^2 R} + \dfrac{2}{h^+(\delta)}\right)\left(\dfrac{2N-1}{N}\right) + R}{\dfrac{4}{h^+\left(\sqrt{2}\delta/R\right)} + \sqrt{2}\left(1 + \dfrac{1}{\left(h^+\left(\sqrt{2}\delta/R\right)\right)^2}\right)} \left(\frac{N}{N-2}\right)\left(\frac{N}{N-1}\right)^2 \tag{6-100}$$

which for sufficiently large R relative to δ (i.e., $h^+\left(\sqrt{2}\delta/R\right) \cong 1$) becomes

$$I(R) = \frac{\left(\dfrac{1}{\left(h^+(\delta)\right)^2 R} + \dfrac{2}{h^+(\delta)}\right)\left(\dfrac{2N-1}{N}\right) + R}{4 + 2\sqrt{2}} \left(\frac{N}{N-2}\right)\left(\frac{N}{N-1}\right)^2 \tag{6-101}$$

Once again we see in Figs. 6-12 and 6-13 the same dependence on N as before approaching

$$I(R) = \frac{2\left(\dfrac{1}{\left(h^+(\delta)\right)^2 R} + \dfrac{2}{h^+(\delta)}\right) + R}{4 + 2\sqrt{2}} \tag{6-102}$$

in the limit as $N \to \infty$. We also note that, whereas in the previous figures for a given value of R the improvement factor decreased with increasing frequency uncertainty, here it increases, which is consistent with Eq. (6-102) since

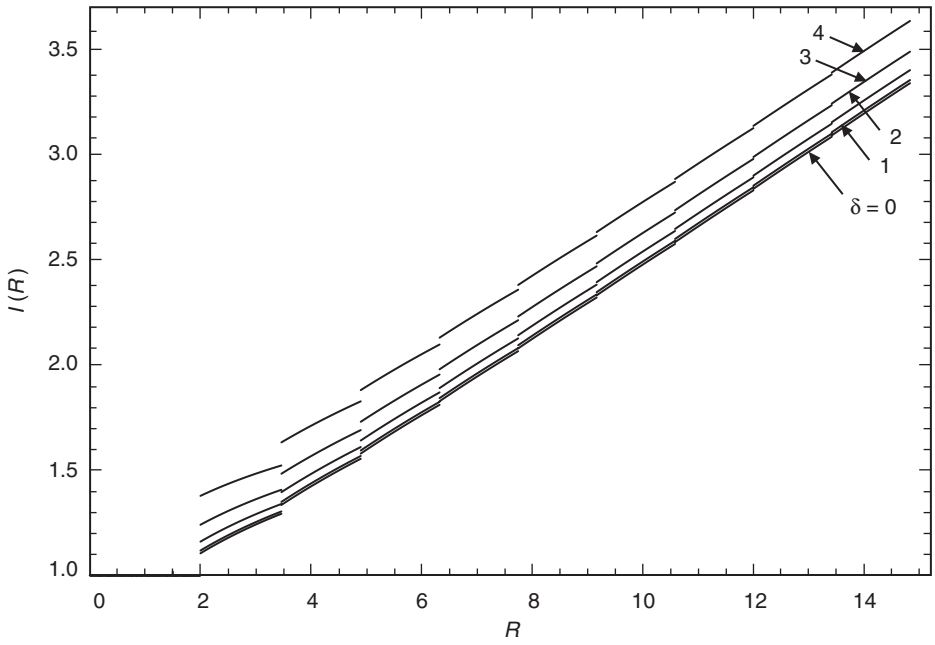


Fig. 6-12. Improvement factor versus SNR with normalized frequency uncertainty as a parameter; Case 2b; $N = 20$**.**

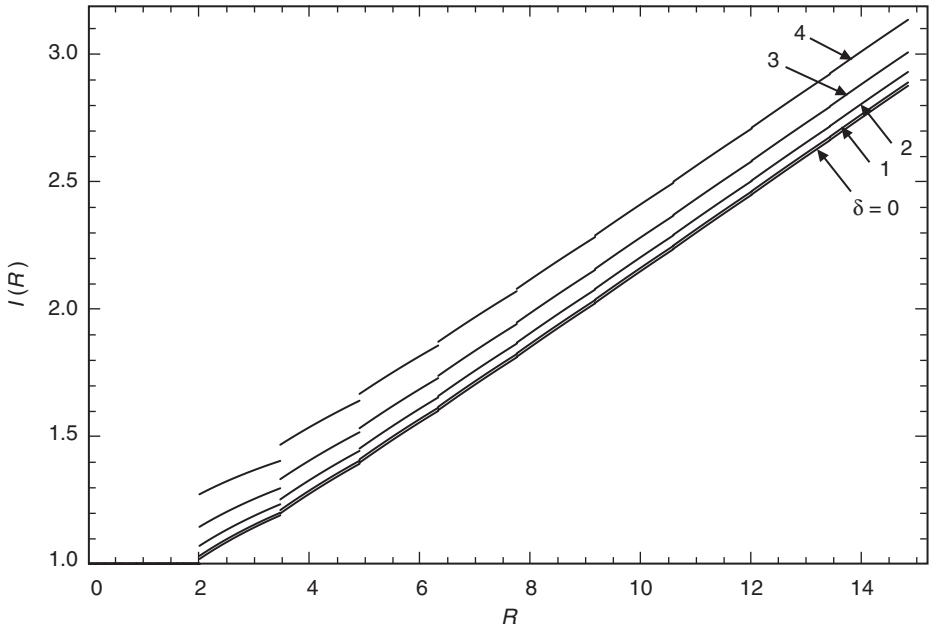


Fig. 6-13. Improvement factor versus SNR with normalized frequency uncertainty as a parameter; Case 2b; $N = 100$**.**

$h^+(\delta) = \text{sinc}^2(\delta/4)$ is a monotonically decreasing function of δ. The intuitive reason for this occurrence is that, for the conventional SSME, the performance degrades much more severely in the presence of large frequency uncertainty than for the improved SSME since for the former the degradation factor $h^+(\delta)$ operates out on its tail, whereas for the latter the effective frequency uncertainty is reduced by a factor of L, and thus for large L the degradation factor $h^+(\delta/L) \cong h\left(\sqrt{2}\delta/R\right)$ operates near its peak of unity. Eventually, for sufficiently large R, the improvement approaches $I(R) = R/\left(4+2\sqrt{2}\right)$ as in Case 1. Finally, comparing Figs. 6-12 and 6-13 with Figs. 6-10 and 6-11, we observe that much larger frequency uncertainties can be tolerated for Case 2b than for Case 1.

6.11 The Impact of the Oversampling Factor on the Performance of the Modified SSME in the Presence of Symbol Timing Error

In Section 6.5 we investigated the performance of the conventional SSME in the presence of symbol timing error. From the results given there, we see for example that if the fractional symbol timing error ε were equal to $1/2$, then from Eqs. (6-49) and (6-51) we would have that

$$E\left\{U^{\pm}\right\} = 2\sigma^2\left(1+\frac{R}{2}\right) \tag{6-103}$$

in which case the performance of the SSME completely degenerates. Since it is desirable to perform SNR estimation prior to obtaining symbol synchronization, it would be advantageous to reduce the sensitivity of the operation of the SSME to knowledge of the symbol timing offset. As we shall show shortly, interestingly enough this can be accomplished by employing an oversampling factor L greater than unity. In fact, the larger the value of L, the less the sensitivity, and in the limit of sufficiently large L, the SSME performance becomes independent of knowledge of the symbol timing.

To illustrate the above statements, assume that for a given L the fractional symbol timing error ε is quantized to $\varepsilon = L_\varepsilon/L$, where for L even, L_ε can take on any of the integer values $0, 1, 2, \cdots, L/2$, and for L odd, L_ε can take on any of the integer values $0, 1, 2, \cdots, (L-1)/2$. Under these circumstances, in the absence of frequency error, the first and second half-symbol I&D outputs would be given by

$$Y_{akl} = md_k \frac{L}{T} \int_{(k-1+(l-1+L_\epsilon)/L)T}^{(k-1+(l-1/2+L_\epsilon)/L)} e^{j\phi} dt + \frac{L}{T} \int_{(k-1+(l-1+L_\epsilon)/L)T}^{(k-1+(l-1/2+L_\epsilon)/L)} n(t)\, dt$$

$$= \frac{md_k}{2} e^{j\phi} + n_{akl} \tag{6-104}$$

$$Y_{\beta l} = md_k \frac{L}{T} \int_{(k-1+(l-1/2+L_\epsilon)/L)T}^{(k-1+(l+L_\epsilon)/L)} e^{j\phi} dt + \frac{L}{T} \int_{(k-1+(l-1/2+L_\epsilon)/L)T}^{(k-1+(l+L_\epsilon)/L)} n(t)\, dt$$

$$= \frac{md_k}{2} e^{j\phi} + n_{\beta kl}, \quad l = 1, 2, \cdots, L - L_e$$

and

$$Y_{akl} = md_{k+1} \frac{L}{T} \int_{(k-1+(l-1+L_\epsilon)/L)T}^{(k-1+(l-1/2+L_\epsilon)/L)} e^{j\phi} dt + \frac{L}{T} \int_{(k-1+(l-1+L_\epsilon)/L)T}^{(k-1+(l-1/2+L_\epsilon)/L)} n(t)\, dt$$

$$= \frac{md_{k+1}}{2} e^{j\phi} + n_{akl} \tag{6-105}$$

$$Y_{\beta l} = md_{k+1} \frac{L}{T} \int_{(k-1+(l-1/2+L_\epsilon)/L)T}^{(k-1+(l+L_\epsilon)/L)} e^{j\phi} dt + \frac{L}{T} \int_{(k-1+(l-1/2+L_\epsilon)/L)T}^{(k-1+(l+L_\epsilon)/L)} n(t)\, dt$$

$$= \frac{md_{k+1}}{2} e^{j\phi} + n_{\beta kl}, \quad l = L - L_\epsilon + 1, L - L_\epsilon + 2, \cdots, L$$

where $n_{\alpha kl}$ and $n_{\beta kl}$ are zero-mean Gaussian RVs with variance independent of the value of ε. Thus, in so far as the modified SSME is concerned, the partitioning of each symbol into L pairs of subdivisions occurs as before with, however, the first $L - L_\epsilon$ now containing the data symbol d_k and the remaining L_ε ones containing the data symbol d_{k+1}. However, since the statistics of the SSME are independent of the data symbols themselves, then we conclude that for the assumed quantization of ε, *the performance of the SSME is independent of the value of symbol timing error.*

Next assume that for a given L the fractional symbol timing error ε is quantized to $\varepsilon = (L_\varepsilon + 1/2)/L$, where again for L even, L_ε can take on any of the integer values $0, 1, 2, \cdots, L/2$, and for L odd, L_ε can take on any of the integer values $0, 1, 2, \cdots, (L-1)/2$. Under these circumstances, in the absence of frequency error, the first and second half-symbol I&D outputs would be given by

the results in Eqs. (6-104) and (6-105) for all values of l with the exception of $l = L - L_\varepsilon$, in which case these outputs become

$$\begin{aligned} Y_{akl}\,|_{l=L-L_\varepsilon} &= \frac{md_k}{2}e^{j\phi} + n_{akl}\,|_{l=L-L_\varepsilon} \\ Y_{\beta l}\,|_{l=L-L_\varepsilon} &= \frac{md_{k+1}}{2}e^{j\phi} + n_{\beta kl}\,|_{l=L-L_\varepsilon} \end{aligned} \tag{6-106}$$

In this case, the sum and difference of the first and second half-symbol I&D outputs become

$$\begin{aligned} u_{kl}^{+} &= md_k e^{j\phi} + n_{akl} + n_{\beta kl} \\ u_{kl}^{-} &= n_{akl} - n_{\beta kl}, \quad l = 1, 2, \cdots, L - L_\varepsilon - 1 \\ u_{kl}^{+}\,|_{l=L-L_\varepsilon} &= m\left(\frac{d_k + d_{k+1}}{2}\right)e^{j\phi} + n_{akl}\,|_{l=L-L_\varepsilon} + n_{\beta kl}\,|_{l=L-L_\varepsilon} \\ u_{kl}^{-}\,|_{l=L-L_\varepsilon} &= m\left(\frac{d_k - d_{k+1}}{2}\right)e^{j\phi} + n_{akl}\,|_{l=L-L_\varepsilon} - n_{\beta kl}\,|_{l=L-L_\varepsilon} \\ u_{kl}^{+} &= md_{k+1}e^{j\phi} + n_{akl} + n_{\beta kl} \\ u_{kl}^{-} &= n_{akl} - n_{\beta kl}, \quad l = L - L_\varepsilon + 1, L - L_\varepsilon + 2, \cdots, L \end{aligned} \tag{6-107}$$

Thus, for the kth symbol, $L - 1$ sum and difference pairs contribute values whose statistics are independent of the value of ε (and thus the same as in the ideal SSME), whereas one sum and difference pair contributes values whose statistics are different from those of the ideal SSME and thus will result in some degradation of performance. To quantify this performance degradation, we need to compute the statistics of the accumulated squared norms of the sum and difference RVs in Eq. (6-107), namely, $U^{\pm} = (1/NL)\sum_{k=1}^{N}\sum_{l=1}^{L}\left|u_{kl}^{\pm}\right|^2$. After some effort, the results for the means and variances, assuming for simplicity BPSK modulation, are as follows:

$$E\left\{U^{+}\right\} = m^2\left(\frac{L-1/2}{L}\right) + 2\sigma^2 L = 2\sigma^2\left[L + \left(\frac{L-1/2}{L}\right)R\right]$$
$$E\left\{U^{-}\right\} = \frac{m^2}{2L} + 2\sigma^2 L = 2\sigma^2\left(L + \frac{R}{2L}\right) \tag{6-108}$$

and

$$\begin{aligned}\operatorname{var}\left\{U^{+}\right\} &= \frac{4}{N}\left[\sigma^4 L + \left(\frac{L-1/2}{L}\right)m^2\sigma^2 + \frac{m^4}{16L^2}\right] \\ &= \frac{4}{N}\sigma^4\left[L + 2\left(\frac{L-1/2}{L}\right)R + \frac{R^2}{4L^2}\right] \\ \operatorname{var}\left\{U^{-}\right\} &= \frac{4}{N}\left(\sigma^4 L + \frac{m^2\sigma^2}{2L} + \frac{m^4}{16L^2}\right) \\ &= \frac{4}{N}\sigma^4\left(L + \frac{R}{L} + \frac{R^2}{4L^2}\right)\end{aligned} \tag{6-109}$$

Note that for $L = 1$ (the conventional SSME) and thus $L_\varepsilon = 0$, i.e., $\varepsilon = 1/2$, Eq. (6-108) agrees with Eq. (6-103) and Eq. (6-109) agrees with the combination of Eqs. (6-49) and (6-52). Furthermore, for sufficiently large L, the moments of $U^{\pm}$ given in Eqs. (6-108) and (6-109) reduce to

$$\begin{aligned}E\left\{U^{+}\right\} &= 2\sigma^2(L+R), \quad E\left\{U^{-}\right\} = 2\sigma^2 L \\ \operatorname{var}\left\{U^{+}\right\} &= \frac{4}{N}\sigma^4(L+2R), \quad \operatorname{var}\left\{U^{-}\right\} = \frac{4}{N}\sigma^4 L\end{aligned} \tag{6-110}$$

which correspond to those of the ideal (perfect symbol timing) SSME.

Finally, we note that for other values of ε between the best quantized ones, namely, $\varepsilon = L_\varepsilon/L$ which yield the same performance as the ideal SSME, and the worst quantized ones, namely, $\varepsilon = (L_\varepsilon + 1/2)/L$ which yield the most degradation in performance, the modified SSME will have a performance between these two extremes.

6.12 Other Modulations

Thus far, we have considered the behavior and performance of the SSME for the class of M-PSK ($M \geq 2$) modulations with and without frequency uncertainty. As we shall show in this section, it is also possible to use the same basic SSME structure (with perhaps slight modification) to provide SNR estimation for offset quadrature phase-shift keying (OQPSK) as well as non-constant envelope modulations such as QAM. As before, the performance of the estimator is still independent of the data symbol sequence as well as the carrier phase and allows for the same enhancement by increasing the number of pairs of observables per symbol in accordance with the true value of SNR.

6.12.1 Offset QPSK

For the case of M-PSK, we indicated in Section 6.1 that the kth transmitted complex symbol in the interval $(k-1)T \leq t \leq kT$ can be represented in the form $d_k = e^{j\phi_k}$, where ϕ_k takes on one of M phases uniformly spaced around the unit circle. A special case of the above, corresponding to $M = 4$, results in conventional quadrature phase-shift keying (QPSK). It is well-known that on nonlinear channels OQPSK provides a performance advantage since it reduces the maximum fluctuation in the signal amplitude by limiting the maximum phase change to 135 deg rather than 180 deg. Since for OQPSK the complex representation of a symbol extends over one and one-half symbols (because of the offset between the I and Q channels), it cannot conveniently be represented in the polar form $d_k = e^{j\phi_k}$ as above. Rather, one should consider the I and Q channel modulations separately. Thus, it is of interest to investigate whether the SSME can be easily modified to accommodate OQPSK and, if so, how its performance is affected. For convenience, we consider only the I&D implementation of the SSME since the same conclusions that will be reached also apply to the multiple samples per symbol version.

Corresponding to the kth QPSK symbol $d_k = e^{j\phi_k} = (a_k + jb_k)/\sqrt{2}$, where a_k and b_k are independent binary (± 1) symbols, the OQPSK transmitter sends $a_k/\sqrt{2}$ during the interval $(k-1)T \leq t \leq kT$ and $b_k/\sqrt{2}$ during the interval $(k-1/2)T \leq t \leq (k+1/2)\,T$. Thus, after complex demodulation by the receiver carrier with frequency uncertainty ω and unknown phase ϕ, the kth complex baseband received signal in the I channel is described by

$$y_I(t) = \frac{1}{\sqrt{2}} m a_k e^{j(\omega t + \phi)} + n_I(t), \quad (k-1)T \leq t \leq kT \tag{6-111}$$

where as before $n_I(t)$ is a zero-mean AWGN process. The signal in Eq. (6-111) is, as before, input to first and second I-channel half-symbol I&Ds operating over

the intervals $(k-1)T \le t \le (k-1/2)T$ and $(k-1/2)T \le t \le kT$, respectively. Analogous to Eq. (6-14), the outputs of these I&Ds are given by

$$\begin{aligned} Y_{I_{\alpha k}} &= \frac{1}{\sqrt{2}} m a_k \frac{1}{T} \int_{(k-1)T}^{(k-1/2)T} e^{j(\omega t+\phi)} dt + \frac{1}{T} \int_{(k-1)T}^{(k-1/2)T} n_Q(t)\, dt \\ &= \left[m a_k / \left(2\sqrt{2}\right) \right] e^{j\phi} e^{j\omega(k-3/4)T}\, \mathrm{sinc}\,(\delta/4) + n_{I_{\alpha k}} \\ Y_{I_{\beta k}} &= \left(\frac{1}{\sqrt{2}} m a_k \frac{1}{T} \int_{(k-1/2)T}^{kT} e^{j(\omega t+\phi)} dt + \frac{1}{T} \int_{(k-1/2)T}^{kT} n_I(t)\, dt \right) e^{-j\theta_k} \\ &= \left(\left[m a_k / \left(2\sqrt{2}\right) \right] e^{j\phi} e^{j\omega(k-3/4)T} e^{j\omega T/2}\, \mathrm{sinc}\,(\delta/4) + n_{I_{\beta k}} \right) e^{-j\theta_k} \end{aligned} \tag{6-112}$$

where $n_{\alpha k}$ and $n_{\beta k}$ are complex Gaussian noise variables with zero mean and variance σ^2, and $e^{-j\theta_k}$ is a phase compensation that accounts for the possible adjustment of the kth second-half sample for phase variations across a given symbol due to the frequency offset.

Similarly, the kth complex baseband received signal in the Q channel is described by

$$y_Q(t) = \frac{1}{\sqrt{2}} m b_k e^{j(\omega t+\phi)} + n_Q(t), \quad (k-1/2)T \le t \le (k+1/2)T \tag{6-113}$$

where $n_Q(t)$ is also a zero-mean AWGN process independent of $n_I(t)$. The signal in Eq. (6-113) is input to first and second Q-channel half-symbol I&Ds operating over the intervals $(k-1/2)T \le t \le kT$ and $kT \le t \le (k+3/2)T$, respectively. Analogous to Eq. (6-112), the outputs of these I&Ds are given by

$$\begin{aligned} Y_{Q_{\alpha k}} &= \frac{1}{\sqrt{2}} m b_k \frac{1}{T} \int_{(k-1/2)T}^{kT} e^{j(\omega t+\phi)} dt + \frac{1}{T} \int_{(k-1/2)T}^{kT} n_Q(t)\, dt \\ &= \left[m b_k / \left(2\sqrt{2}\right) \right] e^{j\phi} e^{j\omega(k-1/4)T}\, \mathrm{sinc}\,(\delta/4) + n_{Q_{\alpha k}} \\ Y_{Q_{\beta k}} &= \left(\frac{1}{\sqrt{2}} m b_k \frac{1}{T} \int_{kT}^{(k+3/2)T} e^{j(\omega t+\phi)} dt + \frac{1}{T} \int_{kT}^{(k+3/2)T} n_Q(t)\, dt \right) e^{-j\theta_k} \\ &= \left(\left[m b_k / \left(2\sqrt{2}\right) \right] e^{j\phi} e^{j\omega(k-1/4)T} e^{j\omega T/2}\, \mathrm{sinc}\,(\delta/4) + n_{Q_{\beta k}} \right) e^{-j\theta_k} \end{aligned} \tag{6-114}$$

Separately taking the half-symbol sums and differences of the Y_I's and Y_Q's results in the following:

$$\begin{aligned} u_{I_k}^{\pm} &= Y_{I_{\alpha k}} \pm Y_{I_{\beta k}} e^{-j\theta_k} \\ &= \left(\frac{ma_k}{2\sqrt{2}}\right) e^{j\phi} e^{j\omega(k-3/4)T} \operatorname{sinc}\left(\frac{\delta}{4}\right) \left[1 \pm e^{j([\delta/2]-\theta_k)}\right] + n_{I_{\alpha k}} \pm n_{I_{\beta k}} e^{-j\theta_k} \\ &\triangleq s_{I_k}^{\pm} + n_{I_k}^{\pm} \end{aligned} \tag{6-115}$$

and

$$\begin{aligned} & u_{Q_k}^{\pm} \\ &= Y_{Q_{\alpha k}} \pm Y_{Q_{\beta k}} e^{-j\theta_k} \\ &= \left(\frac{mb_k}{2\sqrt{2}}\right) e^{j\phi} e^{j\omega(k-1/4)T} \operatorname{sinc}\left(\frac{\delta}{4}\right) \left[1 \pm e^{j([\delta/2]-\theta_k)}\right] + n_{Q_{\alpha k}} \pm n_{Q_{\beta k}} e^{-j\theta_k} \\ &\triangleq s_{Q_k}^{\pm} + n_{Q_k}^{\pm} \end{aligned} \tag{6-116}$$

Note by comparison of Eq. (6-112) with Eq. (6-114) that an additional phase shift of an amount $\omega T/2$ exists in the $u_Q^{\pm}$'s relative to the $u_I^{\pm}$'s, which would not be present if one were to generate the comparable I&D outputs for conventional QPSK. In principle, this phase shift could be perfectly compensated for if one had knowledge of the frequency uncertainty ω. However, in the absence of this exact knowledge, the best one could do at this point would be to multiply the $u_Q^{\pm}$'s by $e^{-j\hat{\omega}T/2}$, which ultimately would result in a degradation in performance if one were first to combine the $u_I^{\pm}$'s and $u_Q^{\pm}$'s into a complex quantity and then to proceed with the formation of the SSME in the same manner as for QPSK. Rather than compensate the phase shift at this point in the implementation, we proceed instead to separately form the averages of the squared norms of the $u_I^{\pm}$'s and $u_Q^{\pm}$'s over the N-symbol duration of the observation resulting in

$$\begin{aligned} U_I^{\pm} &= \frac{1}{N}\sum_{k=1}^{N} \left|u_{I_k}^{\pm}\right|^2 = \frac{1}{N}\sum_{k=1}^{N} \left[\left|s_{I_k}^{\pm}\right|^2 + \left|n_{I_k}^{\pm}\right|^2 + 2\,\mathrm{Re}\left\{s_{I_k}^{\pm}\left(n_{I_k}^{\pm}\right)^*\right\}\right] \\ U_Q^{\pm} &= \frac{1}{N}\sum_{k=1}^{N} \left|u_{I_k}^{\pm}\right|^2 = \frac{1}{N}\sum_{k=1}^{N} \left[\left|s_{Q_k}^{\pm}\right|^2 + \left|n_{Q_k}^{\pm}\right|^2 + 2\,\mathrm{Re}\left\{s_{Q_k}^{\pm}\left(n_{Q_k}^{\pm}\right)^*\right\}\right] \end{aligned} \tag{6-117}$$

Since taking the magnitude of the $u_I^{\pm}$'s and the $u_Q^{\pm}$'s eliminates the relative phase shift between these quantities noted above, then it is straightforward to show that combining $U_I^{\pm}$ (delayed by $T/2$) with $U_Q^{\pm}$ results in a pair of signals $U^{\pm}$ that have the identical statistics as those for conventional QPSK. In particular, setting the half-symbol phase compensation $\theta_k = \omega_{sy}T/2$ (independent of k), then the signal term corresponding to the kth term in the average would be given by

$$\left|s_k^{\pm}\right|^2 = \left|s_{I_k}^{\pm}\right|^2 + \left|s_{Q_k}^{\pm}\right|^2 = m^2 \left(\frac{a_k^2 + b_k^2}{2}\right) \operatorname{sinc}^2\left(\frac{\delta}{4}\right) \left[\frac{1 \pm \cos\left(\delta_{sy}/2\right)}{2}\right]$$

$$= m^2 \operatorname{sinc}^2\left(\frac{\delta}{4}\right) \left[\frac{1 \pm \cos\left(\delta_{sy}/2\right)}{2}\right] \triangleq m^2 h^{\pm} \tag{6-118}$$

where as before $\delta_{sy} = \delta - \omega_{sy}T$.

To see how one can implement a universal SSME structure that will handle OQPSK as well as conventional QPSK, we proceed as follows. Consider partitioning the results of inputting the I- and Q-channel baseband signals to half-symbol I&Ds into even and odd outputs. That is, we define $Y_{I_{\alpha k}}$ and $Y_{Q_{\beta,k-1}}$, which correspond to half-symbol integrations in the interval $(k-1)T \le t \le (k-1/2)T$, as *odd* outputs, and $Y_{I_{\beta k}}$ and $Y_{Q_{\alpha k}}$, which correspond to half-symbol integrations in the interval $(k-1/2)T \le t \le kT$, as *even* outputs. Then, for conventional QPSK, since $u_{I_k}^{\pm}$ is formed from the sum and difference of $Y_{I_{\alpha k}}$ and $Y_{I_{\beta k}}$ and $u_{Q_k}^{\pm}$ is formed from the sum and difference of $Y_{Q_{\alpha k}}$ and $Y_{Q_{\beta,k-1}}$, we can say that $u_{I_k}^{\pm}$ is formed from the kth even and odd outputs, whereas $u_{Q_k}^{\pm}$ is formed from the kth even and (k-1)st (i.e., the preceding) odd outputs. On the other hand, since for OQPSK $u_{I_k}^{\pm}$ is still formed from the sum and difference of $Y_{I_{\alpha k}}$ and $Y_{I_{\beta k}}$ but $u_{Q_k}^{\pm}$ is formed from the sum and difference of $Y_{Q_{\alpha k}}$ and $Y_{Q_{\beta k}}$, we can say that both $u_{I_k}^{\pm}$ and $u_{Q_k}^{\pm}$ are formed from the kth even and odd outputs. Thus, from this viewpoint, the only difference in the SSME implementation between OQPSK and conventional QPSK is that for the former the Q-channel sum and difference signals are formed from the corresponding even and *succeeding* odd half-symbol I&D outputs, whereas for the latter the Q-channel sum and difference signals are formed from the *same* even but the *preceding* odd half-symbol I&D outputs. Other than this minor difference in implementation, the two SSMEs would yield performances identical to that given previously in this chapter.

6.12.2 QAM

For the case of QAM with an M-symbol square signal constellation, the kth transmitted complex symbol in the interval $(k-1)T \leq t \leq kT$ can be represented in the form $d_k = d_{Ik} + jd_{Qk}$ where d_{Ik} and d_{Qk} are independent, identically distributed (iid) RVs that take on the values $\pm 1, \pm 3, \cdots, \pm\left(\sqrt{M}-1\right)$ with equal probability. It is straightforward to show that the mean and variance of $U^{\pm}$ are, analogous to Eq. (6-8), given by

$$\begin{aligned} E\left\{U^{\pm}\right\} &= 2\sigma^2 + E\left\{\left|s_k^{\pm}\right|^2\right\} \\ \operatorname{var}\left\{U^{\pm}\right\} &= \frac{4}{N}\sigma^2\left(E\left\{\left|s_k^{\pm}\right|^2\right\} + \sigma^2\right) \end{aligned} \tag{6-119}$$

where now $\left|s_k^{\pm}\right|^2 = m^2 h\left|d_k\right|^2$ and thus

$$E\left\{\left|s_k^{\pm}\right|^2\right\} = m^2 h E\left\{|d_k|^2\right\} = \frac{2}{3}\left(M-1\right)m^2 h \tag{6-120}$$

However, since in the case of QAM the average SNR is given by

$$\begin{aligned} R &= \frac{m^2}{2\sigma^2} \\ R &= \frac{\frac{2}{3}(M-1)m^2}{2\sigma^2} \end{aligned} \tag{6-121}$$

then combining Eq. (6-121) with Eq. (6-121) and substituting the result in Eq. (6-119), we obtain

$$\begin{aligned} E\left\{U^{\pm}\right\} &= 2\sigma^2\left(1 + h^{\pm}R\right) \\ \operatorname{var}\left\{U^{\pm}\right\} &= \frac{4}{N}\sigma^4\left(1 + 2h^{\pm}R\right) \end{aligned} \tag{6-122}$$

which is identical with the second relations in Eq. (6-8). Thus, solving for R from Eq. (6-122) and following the same logic that led to the ad hoc SSME in Eq. (6-10), we conclude that no modification of this SSME is required to allow its use for estimating SNR when QAM is transmitted. Similarly, in view of the equivalence between Eqs. (6-122) and (6-8), we conclude that the performance is identical to that previously determined for M-PSK modulations.

6.13 The Time-Multiplexed SSME

In Section 6.6, we described a means for potentially improving the performance of the conventional SSME by increasing the number of subdivisions (observables) per symbol beyond two (but still an even number). In particular, we showed that the variance of the so-modified estimator tracks (with a fixed separation from it) the Cramer–Rao bound on the variance of an SNR estimator over the entire range of SNR values. Implicit in the derivation of the expression for the variance of the SNR estimator was the assumption that the even number of subdivisions was the same for all symbols in the observation from which the SNR estimator was formed, and as such an optimum value of the number of subdivisions, denoted by $2L$, was determined for a given true SNR region, the totality of which spans the entire positive real line. Moreover, it was shown that, if one ignores the requirement of having the number of subdivisions be an even integer and proceeds to minimize with respect to L the expression for the variance derived as mentioned above, an optimum value of L can be determined *for every value of true SNR*. The resulting estimator was referred to as the *fictitious* SSME and resulted in a lower bound on the performance of the practical realizable SSME corresponding to integer L.

In this section, we show how one can in practice turn the fictitious SNR estimator into a non-fictitious one. In particular, we demonstrate an implementation of the SSME that allows one to approach the unrestricted optimum value of L (to the extent that it can be computed as the average of a sum of integers) at every true SNR value. More specifically, the proposed approach, herein referred to as the *time-multiplexed SSME*, allows each symbol to possess its own number of subdivisions arranged in any way that, on the average (over all symbols in the observed sequence), achieves the desired optimum value of L. Furthermore, we propose an algorithm for *adaptively* achieving this optimum value of L when in fact one has no a priori information about the true value of SNR. Once again for simplicity of the discussion, we consider the case wherein the symbol pulse shape is assumed to be rectangular, and thus the observables from which the estimator is formed are the outputs of I&Ds.

A block diagram of the complex baseband time-multiplexed SSME is illustrated in Fig. 6-14 with the input signal in the kth interval $(k-1)T \le t \le kT$ as described by Eq. (6-13). Consider uniformly subdividing the kth symbol interval into $2L_k$ (L_k integer) subdivisions each of length $T_k/2 = (T/L_k)/2$. In each of these L_k pairs of split symbol intervals, we apply the signal in Eq. (6-9) to first and second half-symbol normalized (by the integration interval) I&Ds, the outputs of which are summed and differenced to form the signals $\left\{u_{kl}^{\pm} \triangleq s_{kl}^{\pm} + n_{kl}^{\pm}, l = 1, \cdots, L_k\right\}$. For each k, the $u_{kl}^{\pm}$'s are iid; however, their statistics vary from symbol to symbol. Denote the relevant symbol-dependent

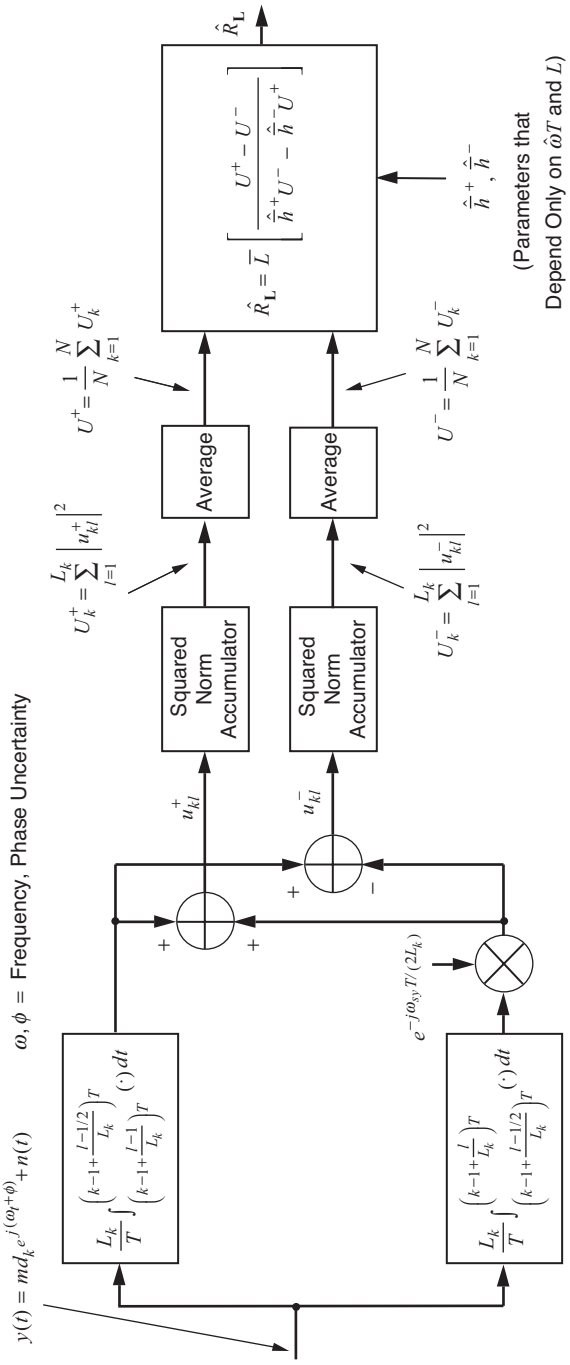


Fig. 6-14. Time-multiplexed split-symbol moments estimator of SNR for *M*-PSK modulation.

parameters of the signal and noise components of $u_{kl}^{\pm}$ as $m_k^2, \sigma_k^2, h_k^{\pm}$, and the SNR in the kth symbol as $R_k = m_k^2/\left(2\sigma_k^2\right)$. In particular, $\sigma_k^2 = \sigma^2 L_k$ is the variance per component (real and imaginary) of $n_{kl}^{\pm}$, and the mean-squared value of $s_{kl}^{\pm}$ can be expressed as[6]

$$\left|s_{kl}^{\pm}\right|^2 = m_k^2 h_k^{\pm} \tag{6-123}$$

where, because of the normalization of the I&Ds, $m_k^2 = m^2$ independently of k, and $h_k^{\pm}$ is again a parameter that reflects the amount of frequency offset and the degree to which it is compensated for. Specifically,

$$h_k^{\pm} = \operatorname{sinc}^2\left(\frac{\delta_k}{4}\right)\frac{1 \pm \cos\left(\delta_{k_{sy}}/2\right)}{2} \tag{6-124}$$

where $\delta_k \triangleq \omega T_k, \delta_{k_{sy}} \triangleq \delta_k - \omega_{sy} T_k = (\omega - \omega_{sy})T_k$, with ω_{sy} the compensation frequency applied to the second half-symbol I&D outputs.

Based on the above, each $\left|u_{kl}^{\pm}\right|^2 = \sigma_k^2 \chi_2^2\left(2h_k^{\pm} R_k\right)$, where $\chi_n^2(\mu)$ denotes a (generally non-central) chi-squared RV with n degrees of freedom, non-centrality parameter μ, and unit variances for each degree of freedom. In general, we know that $E\left\{\chi_n^2(\mu)\right\} = n + \mu$ and $\operatorname{var}\left\{\chi_n^2(\mu)\right\} = 2n + 4\mu$ for all n and μ. Furthermore, using [8, Eq. (2.39)] for the inverse moments of central chi-squared RVs, we have for even n and $\mu = 0$, $E\left\{[\chi_n^2(0)]^{-1}\right\} = (n-2)^{-1}$ and $E\left\{[\chi_n^2(0)]^{-2}\right\} = \left[(n-2)(n-4)\right]^{-1}$. Expressions for higher-order moments of $\chi_n^2(\mu)$ or its reciprocal can be determined using [8, Eq. (2.47)].

Now for each k define $U_k^{\pm} = \sum_{l=1}^{L_k}\left|u_{kl}^{\pm}\right|^2/L_k$. Then, based on the above chi-squared characterization of $\left|u_{kl}^{\pm}\right|^2$, and recognizing that the true SNR to be estimated is given by

$$R = R_k L_k = \frac{m^2}{2\sigma^2} \tag{6-125}$$

we have $U_k^{\pm} = \left(\sigma_k^2/L_k\right)\chi_{2L_k}^2\left(2h_k^{\pm} R_k L_k\right) = \sigma^2 \chi_{2L_k}^2\left(2h_k^{\pm} R\right)$ with first mean and variance

$$\begin{aligned} E\left\{U_k^{\pm}\right\} &= 2\sigma^2\left(L_k + h_k^{\pm} R\right) \\ \operatorname{var}\left\{U_k^{\pm}\right\} &= 4\sigma^4\left(L_k + 2h_k^{\pm} R\right) \end{aligned} \tag{6-126}$$

[6] Note that σ^2 is the variance per component of the $u_k^{\pm}$'s in the conventional SSME corresponding to $L = 1$ in each symbol interval.

Solving for R in terms of $E\{U_k^{\pm}\}$ from the first equation in Eq. (6-126), we obtain

$$R = L_k \left[\frac{E\{U_k^+\} - E\{U_k^-\}}{h_k^+ E\{U_k^-\} - h_k^- E\{U_k^+\}} \right] \tag{6-127}$$

At this point, we could proceed as we did in Section 6.1 by replacing expected values of $U_k^{\pm}$ with their sample values to obtain estimates of R from each symbol, and then averaging over the N estimates obtained from the N symbols, resulting in the ad hoc estimator

$$\hat{R}'_{\mathbf{L}} = \frac{1}{N} \sum_{k=1}^{N} L_k \left[\frac{U_k^+ - U_k^-}{h_k^+ U_k^- - h_k^- U_k^+} \right] \tag{6-128}$$

where $\mathbf{L} = (L_1, L_2, \cdots, L_N)$ denotes the oversampling vector for the N-symbol observation. Unfortunately, this has the potential of being a very bad estimator, because from our previous analyses we have observed that both the bias and the variance of the split-symbol estimate become unbounded if it is based on only a single symbol, i.e., $N = 1$. If $\{L_k\}$ takes on only a few discrete values, we could avoid this singularity by grouping symbols with the same L_k, obtaining an estimate from each group, and then averaging the estimates from all the groups. A better approach is to first average the $U_k^{\pm}$'s prior to forming them into an ad hoc estimator. Specifically, we form $U^{\pm} = (1/N)\sum_{k=1}^{N} U_k^{\pm}$, which has the chi-squared characterization $U^{\pm} = (\sigma^2/N)\chi^2_{2\bar{L}N}(2\bar{h}^{\pm}NR)$, where $\bar{L} = (1/N)\sum_{k=1}^{N} L_k$ and $\bar{h}^{\pm} = (1/N)\sum_{k=1}^{N} h_k^{\pm}$. The mean and variance of $U^{\pm}$ are immediately given by

$$\begin{aligned} E\{U^{\pm}\} &= 2\sigma^2 \left(\bar{L} + \bar{h}^{\pm} R\right) \\ \operatorname{var}\{U^{\pm}\} &= \left(4\sigma^4/N\right)\left(\bar{L} + 2\bar{h}^{\pm} R\right) \end{aligned} \tag{6-129}$$

Solving for R in terms of $E\{U^{\pm}\}$, we obtain

$$R = \bar{L} \left[\frac{E\{U^+\} - E\{U^-\}}{\bar{h}^+ E\{U^-\} - \bar{h}^- E\{U^+\}} \right] \tag{6-130}$$

Now we replace expected values with sample values and $\bar{h}^{\pm}$ with estimates $\hat{\bar{h}}^{\pm}$ based on an estimate $\hat{\omega}$ of the frequency offset ω in this single equation to get our SNR estimate:

$$\hat{R}_{\mathbf{L}} = \bar{L} \left[\frac{U^+ - U^-}{\hat{\bar{h}}^+ U^- - \hat{\bar{h}}^- U^+} \right] \tag{6-131}$$

The equations defining both the estimator $\hat{R}_{\mathbf{L}}$ and the underlying observables $U^{\pm}$ in terms of standard chi-squared random variables are identical in form to those obtained for the special case of uniform subsampling of all the symbols, $\mathbf{L} = (L, L, \cdots, L)$. The parameters $\bar{L}$, $\bar{h}^{\pm}$ for the general case reduce to the constants L, $h^{\pm}$ for the special case. The special case $\mathbf{L} = (L, L, \cdots, L)$ produces the estimator $\hat{R}_L$ of Section 6.6, where we assumed constant L for all symbols. Thus, we can apply our previous performance calculations for $\text{var}\{\hat{R}_L\}$ to obtain the corresponding expressions for $\text{var}\{\hat{R}_{\mathbf{L}}\}$ by simply replacing L and $h^{\pm}$ in those expressions with $\bar{L}$ and $\bar{h}^{\pm}$, respectively. In the case of zero frequency offset, we now can achieve the variance expression for any value of $\bar{L}$ achievable by *averaging* integers, not just integer values of L themselves. For large N, this means that we can achieve the performance of our fictitious estimator $\hat{R}_{\bullet}$ for a very dense set of values of R satisfying $L_{\bullet}(R) = R/\sqrt{2} \geq 1$. Of course, the fictitious estimator remains fictitious for $L_{\bullet}(R) = R/\sqrt{2} < 1$ (i.e., the region of R where we did not attempt to use it as a benchmark).

6.13.1 An Adaptive SSME

Given that $\hat{R}_{\mathbf{L}}$ achieves the performance of $\hat{R}_{\bar{L}}$, we now have a method for adaptively selecting the oversampling factor L. We can start with an initial guess, and then increase or decrease L in response to intermediate SNR estimates $\hat{R}_{\mathbf{L}}$ based on the symbols observed up to now. The key point is that the estimator $\hat{R}_{\mathbf{L}}$ at any point in time achieves exactly the same performance as the estimator $\hat{R}_L$ with $L = \bar{L}$, based on the same cumulative number of symbols. Thus, no symbols are wasted if an adaptive SNR estimation algorithm starts out with a non-optimum value of L but adapts over time to generate a vector sequence $\mathbf{L}$ for which the average $\bar{L}$ approaches the optimum value of L, namely, $L = L_{\bullet}(R) = R/\sqrt{2}$. Figure 6-15 is a flow diagram of such an adaptive scheme modeled after the robust version of the generalized SSME discussed in Section 6.7, wherein the integer values of L are restricted to the set $b^l, l = 0, 1, 2, 3, \cdots$ for some integer base b. The operation of the scheme is described as follows.

Initially, consider an observation of n symbols and set $L_k = L = 1, k = 1, 2, \cdots, n$. Next, evaluate the sum and difference accumulated variables $U^{\pm}$ for the n symbol observation. Proceed to evaluate the SNR estimator $\hat{R} = \hat{R}_{\mathbf{L}}(U^{\pm})$ in accordance with Eq. (6-131) taking note of the fact that, for this choice of $\mathbf{L}$, $\bar{L} = 1$. Next, we compare the current value of L, namely $L = 1$, to the

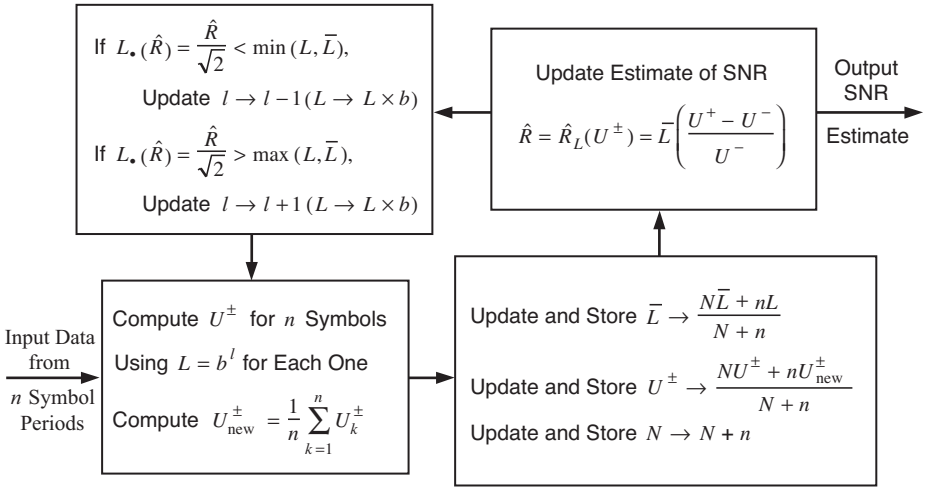


Fig. 6-15. A robust adaptive SSME scheme.

desired optimum value of L, based however on the current estimate of R, i.e., $L_\bullet(\hat{R}) = \hat{R}/\sqrt{2}$, to get an indication of how close we are to where we are headed. If $L_\bullet(\hat{R})$ exceeds unity, which on the average is likely to be the case if the true SNR is greater than $\sqrt{2}$, increment L by multiplying it by b and proceed to process the next n input symbols, as will be described momentarily. On the other hand, if $L_\bullet(\hat{R})$ is less than or equal to unity, which on the average is likely to be the case if the true SNR is less than or equal to $\sqrt{2}$, then leave L unchanged[7] and again proceed to process the next n input symbols. Moving on to the next set of n symbols, compute new values of $U^{\pm}$, denoted by $U^{\pm}_{\text{new}}$, using the updated value of L as determined above for all $L_k, k = n+1, n+2, \cdots 2n$. Let N denote the running average of the number of symbols. (Assume that initially N was set equal to n corresponding to the first set of observed symbols.) Update the current values of $U^{\pm}$ with the new $U^{\pm}_{\text{new}}$ values according to the weighted average $\left(NU^{\pm} + nU^{\pm}_{\text{new}}\right)/(N+n)$ and store these as $U^{\pm}$. Update the running average of L in accordance with $\left(N\bar{L} + nL\right)/(N+n)$ and store this as $\bar{L}$. Finally, update the value of N to $N+n$ and store this new value. Using the updated $U^{\pm}$, compute an updated SNR estimate $\hat{R} = \hat{R}_{\mathbf{L}}\left(U^{\pm}\right)$ in accordance with Eq. (6-131). Next, using this updated SNR estimate, compute the updated estimate of the optimum L, namely, $L_\bullet(\hat{R}) = \hat{R}/\sqrt{2}$ and use it to update the current value of L in accordance with the following rule:

---

[7] As we shall see shortly, in all other circumstances of this nature, we would proceed to decrement L by dividing it by b. However, since the current value of L is already equal to unity, which is the smallest nonzero integer, we cannot reduce it any further.

$$\begin{aligned}
&\text{If } L_\bullet\left(\hat{R}\right) < \min\left(L, \bar{L}\right), \text{ then divide } L \text{ by } b \\
&\text{If } L_\bullet\left(\hat{R}\right) > \max\left(L, \bar{L}\right), \text{ then multiply } L \text{ by } b \\
&\text{If } \min\left(L, \bar{L}\right) \le L_\bullet\left(\hat{R}\right) \le \max\left(L, \bar{L}\right), \text{ do not change } L
\end{aligned} \tag{6-132}$$

Finally, using the updated value of L, proceed to process the next n symbols, whereupon the algorithm repeats as described above.

To illustrate the behavior of the robust adaptive SSME scheme, simulations were conducted to demonstrate the rate at which $\bar{L}$ converges to the true optimum L and also the manner in which this convergence takes place. The first simulation, illustrated in Fig. 6-16, demonstrates the ideal performance of the scheme assuming no frequency error, i.e., $\hat{\bar{h}}^+ = 1, \hat{\bar{h}}^- = 0$, and the following parameters: $R = 10, b = 2, n = 10$. By "ideal" is meant that the same adaptive feedback rule for updating L as in Eq. (6-132) is used except that a magic genie is assumed to be available to provide the true SNR, R, to the update rule rather than using the estimate of R. That is, the update of L in accordance with Eq. (6-132) is carried out using $L_\bullet(R)$ rather than $L_\bullet\left(\hat{R}\right)$. The horizontal axis in Fig. 6-16 is measured in discrete units of time corresponding to the cumulative number of n-symbol batches processed each with a fixed value of L. The vertical axis represents two different indicators of the performance corresponding to the behavior of $\log_2 L$ and $\log_2 \bar{L}$ as they are updated in each cycle through the feedback loop. For the assumed parameters, the optimum value of L to which the scheme should adapt is given in logarithmic terms by $\log_2 L_\bullet(10) = \log_2\left(10/\sqrt{2}\right) = 2.822$. From the plots in Fig. 6-16, we observe that $\log_2 L$ quickly rises (in three steps) from its initial value of $\log_2 1 = 0$ to $\log_2 8 = 3$ and then eventually fluctuates between $\log_2 8 = 3$ and $\log_2 4 = 2$ with a 3:1 or 4:1 duty cycle. At the same time, $\log_2 \bar{L}$ smoothly rises toward the optimum $\log_2 \bar{L}_\bullet$, converging asymptotically to this limit (with indistinguishable difference) in fewer than 20 cycles of the feedback loop or, equivalently, 200 symbol intervals.

Figure 6-17 is an illustration of the actual performance of the scheme as illustrated in Fig. 6-15, i.e., in the absence of a magic genie to provide the true SNR. The same parameter values as in Fig. 6-16 were assumed, and 10 different trials were conducted. Also superimposed on this figure for the purpose of comparison is the $\log_2 \bar{L}$ magic genie performance obtained from Fig. 6-16. For 6 out of the 10 trials, the actual performance was indistinguishable from that corresponding to the magic genie. For the remaining 4 trials, $\log_2 \bar{L}$ overshoots its target optimum value but still settles toward this value within 20 cycles of the algorithm. For all 10 trials, there is a small dispersion from the optimum level even after 40 cycles. This is due to residual error in estimating R after $N = 400$ symbols since the variance only decreases as $1/N$.

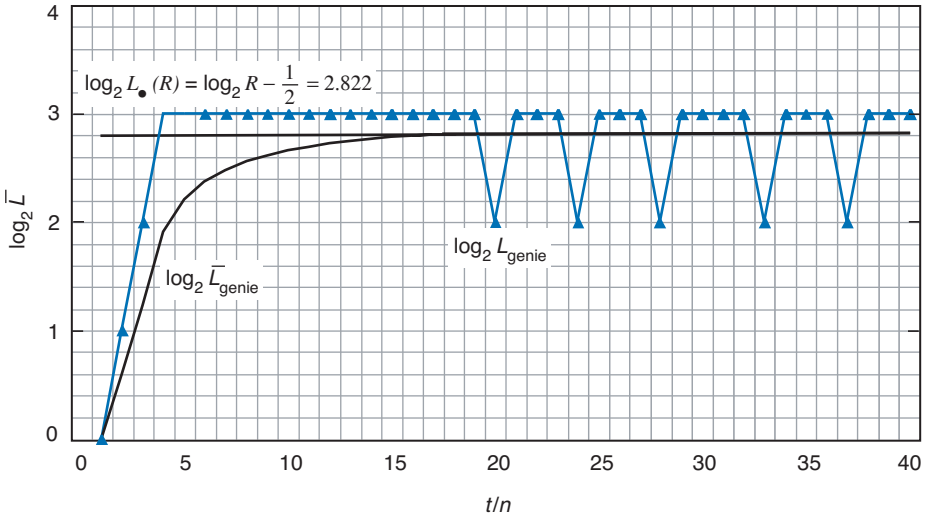


Fig. 6-16. Ideal performance of the robust adaptive SSME scheme. (Adaptive SSME with magic genie estimate of true SNR, $R = 10$.)

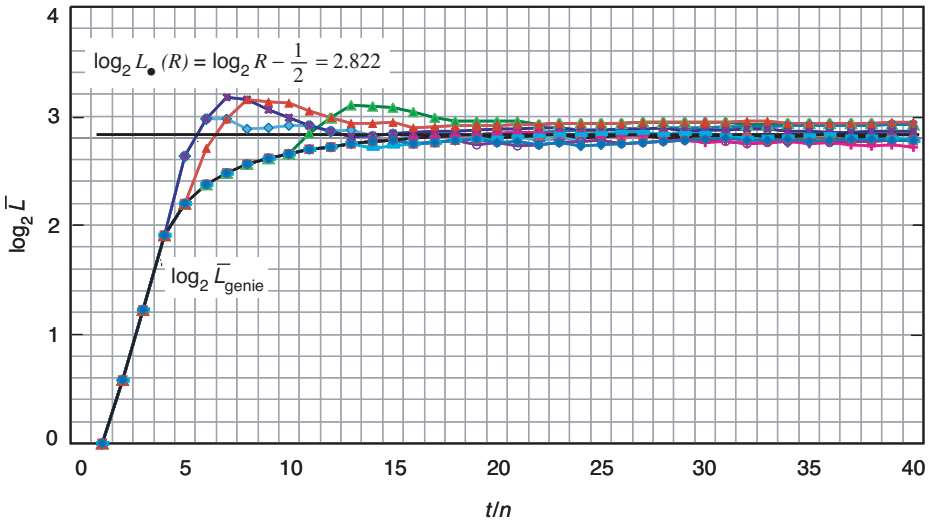


Fig. 6-17. Actual performance of the robust adaptive SSME scheme. (Adaptive SSME: 10 trials with $N = 400$, $n = 10$, true $R = 10$.)

References

[1] D. R. Pauluzzi and N. C. Beaulieu, "A Comparison of SNR Estimation Techniques for the AWGN Channel," *IEEE Trans. Commun.*, vol. 48, pp. 1681–1691, October 2000.

[2] M. K. Simon and A. Mileant, "SNR Estimation for the Baseband Assembly," *The Telecommunications and Data Acquisition Progress Report 42-85, January–March 1986*, Jet Propulsion Laboratory, Pasadena, California, pp. 118–126, May 15, 1986. http://ipnpr.jpl.nasa.gov/

[3] B. Shah and S. Hinedi, "The Split Symbol Moments SNR Estimator in Narrow-Band Channels," *IEEE Trans. Aerosp. Electron. Syst.*, vol. AES-26, pp. 737–747, September 1990.

[4] S. Dolinar, "Exact Closed-Form Expressions for the Performance of the Split-Symbol Moments Estimator of Signal-to-Noise Ratio," *The Telecommunications and Data Acquisition Progress Report 42-100, October–December 1989*, Jet Propulsion Laboratory, Pasadena, California, pp. 174–179, February 15, 1990. http://ipnpr.jpl.nasa.gov/

[5] Y. Feria, "A Complex Symbol Signal-to-Noise Ratio Estimator and its Performance," *The Telecommunications and Data Acquisition Progress Report 42-116, October–December 1993*, Jet Propulsion Laboratory, Pasadena, California, pp. 232–245, February 15, 1994. http://ipnpr.jpl.nasa.gov/

[6] M. K. Simon and S. Dolinar, "Signal-to-Noise Ratio Estimation for Autonomous Receiver Operation," *GLOBECOM 2004 Conference Record*, Dallas, Texas, November 2004.

[7] M. Simon and S. Dolinar, "Improving Signal-to-Noise Ratio Estimation for Autonomous Receivers," *The Interplanetary Network Progress Report*, vol. 42-159, Jet Propulsion Laboratory, Pasadena, California, pp. 1–19, November 15, 2004. http://ipnpr.jpl.nasa.gov/

[8] M. K. Simon, *Probability Distributions Involving Gaussian Random Variables: A Handbook for Engineers and Scientists*, Norwell, Massachusetts: Kluwer Academic Publishers, 2002.

[9] M. Abramowitz and I. A. Stegun, *Handbook of Mathematical Functions with Formulas, Graphs, and Mathematical Tables*, 9th ed., New York: Dover Press, 1972.

[10] A. Papoulis, *Probability, Random Variables, and Stochastic Processes*, New York: McGraw-Hill, 1965.

[11] H. L. V. Trees, *Detection, Estimation, and Modulation Theory*, vol. 1, New York: Wiley, 1968.

[12] C. M. Thomas, *Maximum Likelihood Estimation of Signal-to-Noise Ratio*, Ph.D. thesis, University of Southern California, Los Angeles, 1967.

[13] S. J. Dolinar, "Cramer-Rao Bounds for Signal-to-Noise Ratio and Combiner Weight Estimation," *The Telecommunications and Data Acquisition Progress Report 42-86, April–June 1986*, Jet Propulsion Laboratory, Pasadena, California, pp. 124–130, August 15, 1986. http://ipnpr.jpl.nasa.gov/

Appendix 6-A
Derivation of Asymptotic Mean and Variance of SSME

In this appendix, we derive the asymptotic expressions for the mean and variance of the SSME as given by Eqs. (6-32) and (6-33), respectively. For convenience, we repeat the expressions for the mean and variance of $U^{\pm}$, namely,

$$\begin{aligned} E\left\{U^{\pm}\right\} &= 2\sigma^2 + \left|s_k^{\pm}\right|^2 = 2\sigma^2\left(1 + h^{\pm}R\right) \\ \text{var}\left\{U^{\pm}\right\} &= \frac{4}{N}\sigma^2\left(\left|s_k^{\pm}\right|^2 + \sigma^2\right) = \frac{4}{N}\sigma^4\left(1 + 2h^{\pm}R\right) \end{aligned} \tag{A-1}$$

Starting from the definition of $g\left(U^+, U^-\right)$ in Eq. (6-10), we evaluate its first and second partial derivatives as

$$\begin{aligned} \frac{\partial g}{\partial U^{\pm}} &= \frac{\pm\left(\hat{h}^+ - \hat{h}^-\right)U^{\mp}}{\left(\hat{h}^+U^- - \hat{h}^-U^+\right)^2} \\ \frac{1}{2}\frac{\partial^2 g}{\partial\left(U^{\pm}\right)^2} &= \frac{\left(\hat{h}^+ - \hat{h}^-\right)\hat{h}^{\mp}U^{\mp}}{\left(\hat{h}^+U^- - \hat{h}^-U^+\right)^3} \end{aligned} \tag{A-2}$$

The quantity $\hat{h}^+U^- - \hat{h}^-U^+$ that appears in the denominator of $g\big(U^+, U^-\big)$ and its partial derivatives is evaluated at the point $\big(U^+, U^-\big) = \big(E\{U^+\}, E\{U^-\}\big)$ as

$$\hat{h}^+E\left\{U^-\right\} - \hat{h}^-E\left\{U^+\right\} = 2\sigma^2\left[\left(\hat{h}^+ - \hat{h}^-\right) + \left(\hat{h}^+h^- - \hat{h}^-h^+\right)R\right] \tag{A-3}$$

The second term in parentheses in Eq. (A-3) evaluates to zero for cases 0, 2a, 2b, and 2c for which the frequency estimate is perfect, i.e., $\hat{\omega} = \omega$, since in this instance $\hat{h}^{\pm} = h^{\pm}$. The numerators of $g\left(U^+, U^-\right)$ and its partial derivatives evaluated at the point $\left(U^+, U^-\right) = \left(E\left\{U^+\right\}, E\left\{U^-\right\}\right)$ are, respectively,

$$
\begin{aligned}
E\left\{U^{+}\right\} - E\left\{U^{-}\right\} &= 2\sigma^2\left(h^{+} - h^{-}\right)R \\
\pm\left(\hat{h}^{+} - \hat{h}^{-}\right)E\left\{U^{\mp}\right\} &= \pm\left(\hat{h}^{+} - \hat{h}^{-}\right)2\sigma^2\left(1 + h^{\mp}R\right) \\
\left(\hat{h}^{+} - \hat{h}^{-}\right)h^{\mp}E\left\{U^{\mp}\right\} &= \left(\hat{h}^{+} - \hat{h}^{-}\right)h^{\mp}2\sigma^2\left(1 + h^{\mp}R\right)
\end{aligned}
\tag{A-4}
$$

Substituting the expressions in Eqs. (A-3) and (A-4) into Eqs. (6-10) and (A-2), we obtain

$$
g\,\big|_{(E\{U^{+}\},E\{U^{-}\})} = \frac{\left(h^{+} - h^{-}\right)R}{\hat{h}^{+} - \hat{h}^{-} + \left(\hat{h}^{+}h^{-} - \hat{h}^{-}h^{+}\right)R}
$$

$$
\frac{\partial g}{\partial U^{\pm}}\,\big|_{(E\{U^{+}\},E\{U^{-}\})} = \pm\frac{1}{2\sigma^2}\frac{\left(\hat{h}^{+} - \hat{h}^{-}\right)\left(1 + h^{\mp}R\right)}{\left[\hat{h}^{+} - \hat{h}^{-} + \left(\hat{h}^{+}h^{-} - \hat{h}^{-}h^{+}\right)R\right]^2} \tag{A-5}
$$

$$
\frac{1}{2}\frac{\partial^2 g}{\partial\left(U^{\pm}\right)^2}\,\big|_{(E\{U^{+}\},E\{U^{-}\})} = \frac{1}{4\sigma^4}\frac{\left(\hat{h}^{+} - \hat{h}^{-}\right)h^{\mp}\left(1 + h^{\mp}R\right)}{\left[\hat{h}^{+} - \hat{h}^{-} + \left(\hat{h}^{+}h^{-} - \hat{h}^{-}h^{+}\right)R\right]^3}
$$

Finally, substituting the expression for $\mathrm{var}\{U^{\pm}\}$ in Eq. (A-1) along with the expressions in Eq. (A-5) into Eq. (6-31) results after some simplification in

$$
\begin{aligned}
E\left\{\hat{R}\right\} = {} & \frac{\left(h^{+} - h^{-}\right)R}{\hat{h}^{+} - \hat{h}^{-} + \left(\hat{h}^{+}h^{-} - \hat{h}^{-}h^{+}\right)R} \\
& + \frac{1}{N}\frac{\left(\hat{h}^{+} - \hat{h}^{-}\right)\left(\hat{h}^{+} + \hat{h}^{-}\right)}{\left[\hat{h}^{+} - \hat{h}^{-} + \left(\hat{h}^{+}h^{-} - \hat{h}^{-}h^{+}\right)R\right]^3} \\
& \times\left\{1 + \left(h^{+} + h^{-} + \frac{\hat{h}^{+}h^{-} + \hat{h}^{-}h^{+}}{\hat{h}^{+} + \hat{h}^{-}}\right)R + 2h^{+}h^{-}R^2\right\} + O\left(\frac{1}{N^2}\right)
\end{aligned}
\tag{A-6}
$$

and

$$\text{var}\left\{\hat{R}\right\} =$$

$$\frac{1}{N}\frac{\left(\hat{h}^{+}-\hat{h}^{-}\right)^{2}}{\left[\hat{h}^{+}-\hat{h}^{-}+\left(\hat{h}^{+}h^{-}-\hat{h}^{-}h^{+}\right)R\right]^{4}}$$

$$\times\left\{2+4\left(h^{+}+h^{-}\right)R+\left[\left(h^{+}+h^{-}\right)^{2}+6h^{+}h^{-}\right]R^{2}+4h^{+}h^{-}\left(h^{+}+h^{-}\right)R^{3}\right\}$$

$$+O\left(\frac{1}{N^{2}}\right) \tag{A-7}$$

which are repeated as Eqs. (6-32) and (6-33) in Section 6.4.2.

Chapter 7
Data Rate Estimation

Andre Tkacenko and Marvin K. Simon

In an autonomous radio operation setting, one of the first parameters that we would like to estimate reliably would be the data rate of the received signal. Knowledge of this parameter is required to carry out maximum-likelihood (ML) detection [1] of other parameters, such as the carrier phase or modulation type. Although ML estimation of the data rate itself is statistically optimal, given that there is little to no a priori knowledge of the incoming signal, this approach is often difficult if not impossible to do in practice.

One mitigating factor for the autonomous radio under consideration is the fact that the data rates are assumed to come from a set of known values, such as the data rates used in the Electra radio (see [2] and Chapter 2). In particular, the data rates here are assumed to be related by integer powers of an integer base B. This assumption, as will soon be shown, allows us to estimate the *true* data rate based on estimates of the signal-to-noise ratio (SNR) computed for various *assumed* data rates. The method for estimating the SNR here is the split-symbol moments estimator (SSME) discussed in [3] and Chapter 6. This estimator is appealing in that the only parameter required for its operation is the assumed data rate. Hence, estimation of the data rate can be done *jointly* with that of the SNR.

Although this approach provides us with a way to estimate both the data rate and SNR together, it will be shown that it is sensitive to symbol-timing error or jitter. In fact, the presence of symbol-timing error can severely degrade the performance of this estimator, as shown in Chapter 6. To overcome this, a modification is proposed in which the jitter is quantized and estimated alongside the data rate and SNR. This approach, based on a so-called generalized likelihood ratio test (GLRT) [4], is robust in the presence of symbol-timing error and

can be used to jointly estimate the data rate, SNR, and symbol-timing error all at once. The estimates of the symbol-timing error obtained can then be used as coarse initial estimates for the data-transition tracking loop (DTTL), which is used later in the receiver to obtain a fine estimate of the timing jitter (see [5] and Chapter 10 for more details).

In Section 7.1, we review the received signal model assumptions and show how the SSME can be used to obtain an estimate of the data rate in the absence of symbol-timing error. This leads to an algorithm for estimating the data rate, which we present in Section 7.1.3. A slight modification to this algorithm that resembles a GLRT-type approach is presented in Section 7.1.4.

In Section 7.2, we investigate the effects of the presence of symbol-timing error on the data rate estimation algorithm. There, it is shown that the presence of a severe jitter can in fact cause the data rate estimator to unequivocally fail.

By quantizing the symbol-timing error, we show in Section 7.3 how to modify the algorithms in Sections 7.1.3 and 7.1.4 to account for the presence of symbol-timing error. There, an all-digital implementation of the SSME-based data rate estimation system is presented in Section 7.3.1. This leads to a joint data rate/SNR/symbol-timing error estimation technique that we describe in Section 7.3.2 and a GLRT-type modification to this method described in Section 7.3.3.

Simulation results for the joint data rate/SNR/symbol-timing error estimation techniques of Sections 7.3.2 and 7.3.3 are presented in Section 7.4. There, the strengths and weaknesses of each of the proposed techniques are revealed in terms of probability of data rate misclassification, SNR estimation error, and jitter estimation error.

7.1 Data Rate Estimation Based on the Mean of the SSME SNR Estimator

7.1.1 Signal Model and Assumptions

The baseband signal received at the autonomous radio is assumed to consist of a constant amplitude digital data stream corrupted only by artifacts due to the conversion from intermediate frequency (IF) to baseband as well as to additive noise. Mathematically, the received signal $\tilde{r}(t)$ is assumed to have the following form in the complex baseband representation:

$$\tilde{r}(t) = A\left(\sum_{k=-\infty}^{\infty} d_k p\big(t-(k+\varepsilon)T\big)\right) e^{j(\omega_r t+\theta_c)} + \tilde{n}(t) \tag{7-1}$$

Here, we have the following:

$A =$ signal amplitude

$d_k =$ kth data symbol, typically assumed to be an M-PSK symbol [1]

$p(t) =$ data pulse shape, typically either a non-return to zero (NRZ) or a Manchester pulse [1]

$T =$ symbol period of the data

$\varepsilon =$ symbol-timing error (jitter), assumed to be uniform over the interval $[0, 1)$

$\omega_r =$ residual frequency offset after demodulation and frequency correction

$\theta_c =$ carrier phase, assumed to be uniform over the interval $[0, 2\pi)$

$\tilde{n}(t) =$ complex additive white Gaussian noise (AWGN) whose real and imaginary parts are uncorrelated, zero-mean processes with two-sided power spectral density (psd)[1] $N_0/2$

Prior to estimating parameters such as the carrier phase θ_c or the frequency offset ω_r, we would like to estimate the data rate given by $\mathcal{R} \triangleq 1/T$. As with the Electra radio (see Chapter 2), we assume that the set of possible data rates $\{\mathcal{R}\}$ comes from a *known* finite set of values of the form

$$\mathcal{R} = B^{\ell}\mathcal{R}_b, \quad 0 \le \ell \le \ell_{\max} \tag{7-2}$$

where B, ℓ, and $\ell_{\max}$ are nonnegative integers and $\mathcal{R}_b \triangleq 1/T_b$ is the *basic* (or lowest) data rate. In other words, every possible data rate is a *base power* of the lowest basic rate. Here B is called the *rate base*, whereas ℓ is referred to as the *rate power*. We denote the maximum rate power by $\ell_{\max}$, and so the number of possible data rates is given by $(\ell_{\max} + 1)$, as can be seen from Eq. (7-2). For the Electra radio [2], we have

$B = 2$

$\ell_{\max} = 12$

$\mathcal{R}_b = 1$ ksymbol/s

With regard to estimating the data rate of the signal $\tilde{r}(t)$ from Eq. (7-1), it is assumed that we know both the rate base B and the basic data rate $\mathcal{R}_b$.

[1] The two-sided psd of the real and imaginary parts is defined to be $N_0/2$ here to set the two-sided psd of the *complex* baseband noise process $n(t)$ at N_0. This is a standard notational convention for a complex baseband AWGN process [1]. It should be noted here that the parameter N_0 differs from that used for the passband process $n(t)$ appearing in other chapters of the book by a factor of two.

Hence, from Eq. (7-2), the only ambiguity of the data rate that exists is the rate power ℓ. This greatly simplifies the data rate estimation problem, since ℓ only varies over a finite set of *known* integers. In what follows, we will assume that the symbol-timing error ε is zero. The case for which $\varepsilon \neq 0$ will be considered in Section 7.2.

7.1.2 Relation of the SSME SNR Estimator to Data Rate Estimation

A block diagram of the SSME system for estimating the SNR of the signal $\tilde{r}(t)$ from Eq. (7-1) is shown in Fig. 7-1 for the case of a rectangular NRZ pulse shape. (For different pulse shapes, the only thing that needs to be changed is that the half-symbol integrate-and-dump (I&D) circuits need to be replaced with half-symbol matched filters [1].) Here, T_s denotes the assumed symbol period of the system (i.e., the sample period); N_s denotes the number of observations; and ω_{sy}, $\hat{h}^+$, and $\hat{h}^-$ denote frequency and phase compensation factors as described in Chapter 6.

From Chapter 6, it is known that if the assumed data rate $\mathcal{R}_s \triangleq 1/T_s$ and N_s satisfy

$$\mathcal{R}_s = L\mathcal{R}, \quad N_s = LN$$

for some positive integers L and N, then the mean of the SNR estimate $\hat{R}_\ell$ is given as follows:

$$E\left[\hat{R}_\ell\right] = \frac{RN+1}{LN-1} = \frac{\dfrac{R}{L} + \dfrac{1}{LN}}{1 - \dfrac{1}{LN}} = \frac{R}{L} + \frac{1}{LN}\left(\frac{R}{L} + 1\right) + O\left(\frac{1}{N^2}\right)$$

where R is the true SNR given by $R = (A^2T)/N_0$. For large N, this simplifies within $O(1/N)$ to become

$$E\left[\hat{R}_\ell\right] \cong \frac{R}{L} \tag{7-3}$$

In other words, if the assumed data rate $\mathcal{R}_s$ is an integer multiple L of the true data rate $\mathcal{R}$, then the SSME still works as before, but it formulates an estimate of the reduced SNR R/L when the number of observations is large enough. As we shall soon see, it is this property that will allow us to use the SSME system to estimate the data rate.

To see how the SSME can be used to estimate the data rate, suppose first that the SSME operates at the highest possible rate, which is simply $\mathcal{R}_s = B^{\ell_{\max}}\mathcal{R}_b$

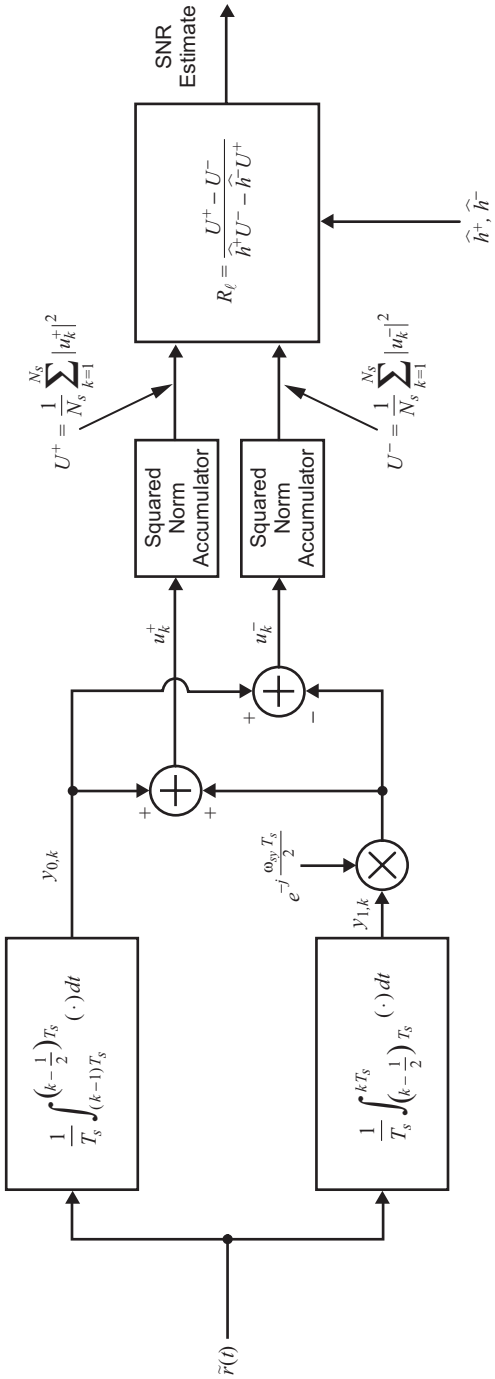


Fig. 7-1. Split-symbol moments estimator (SSME) for a rectangular NRZ pulse shape.

from Eq. (7-2). As $\mathcal{R} = B^{\ell}\mathcal{R}_b$, we have $\mathcal{R}_s = L\mathcal{R}$, where $L = B^{\ell_{\max}-\ell}$. Then, from Eq. (7-3), we have,

$$E\left[\hat{R}_0\right] = \frac{R}{B^{\ell_{\max}-\ell}} + O\left(\frac{1}{N}\right)$$

If the SSME is operated at the next lower rate (i.e., $\mathcal{R}_s = B^{\ell_{\max}-1}\mathcal{R}_b$), then we have $L = B^{\ell_{\max}-\ell-1}$, and so from Eq. (7-3) we have

$$E\left[\hat{R}_1\right] = \frac{R}{B^{\ell_{\max}-\ell-1}} + O\left(\frac{1}{N}\right) = BE\left[\hat{R}_0\right] + O\left(\frac{1}{N}\right)$$

In other words, lowering the rate by one step *increases* the mean of the SNR estimate by a factor of B.

If we continue to run the SSME, lowering the assumed data rate $\mathcal{R}_s$ by a factor of B at each run, then on the $(\ell_{\max} - \ell)$th run, we will obtain an SNR estimate based on the *true* data rate $\mathcal{R}$, in which case we have

$$E\left[\hat{R}_{\ell_{\max}-\ell}\right] = R + O\left(\frac{1}{N}\right) = B^{\ell_{\max}-\ell}E\left[\hat{R}_0\right] + O\left(\frac{1}{N}\right)$$

Note that up to this point we have

$$E\left[\hat{R}_i\right] = B^i E\left[\hat{R}_0\right] + O\left(\frac{1}{N}\right) \tag{7-4}$$

In other words, the mean of the SNR estimate *monotonically increases* by a factor of B each time the rate is lowered until the true data rate (and hence the true SNR) is reached.

If the assumed data rate is lowered one more step so that $\mathcal{R}_s = B^{\ell-1}\mathcal{R}_b = (1/B)\mathcal{R}$, then the SSME will attempt to create an SNR estimate based on B successive data symbols. This will severely degrade the performance of the estimator since the data symbols fluctuate randomly. To see this, consider the case where $B = 2$ and the data come from a binary phase-shift keying (BPSK) constellation [1]. In this case, the signal portion of the I&D outputs $y_{0,k}$ and $y_{1,k}$ can either constructively or destructively interfere depending on whether adjacent data symbols are the same or different, respectively. This is illustrated in Fig. 7-2.

When two adjacent data symbols are the same, as in Fig. 7-2(a), we will get a valid contribution to the SNR estimate, since $|u_k^+|^2$ from Fig. 7-1 will be an approximate measure of the signal power plus the noise power, whereas $|u_k^-|^2$

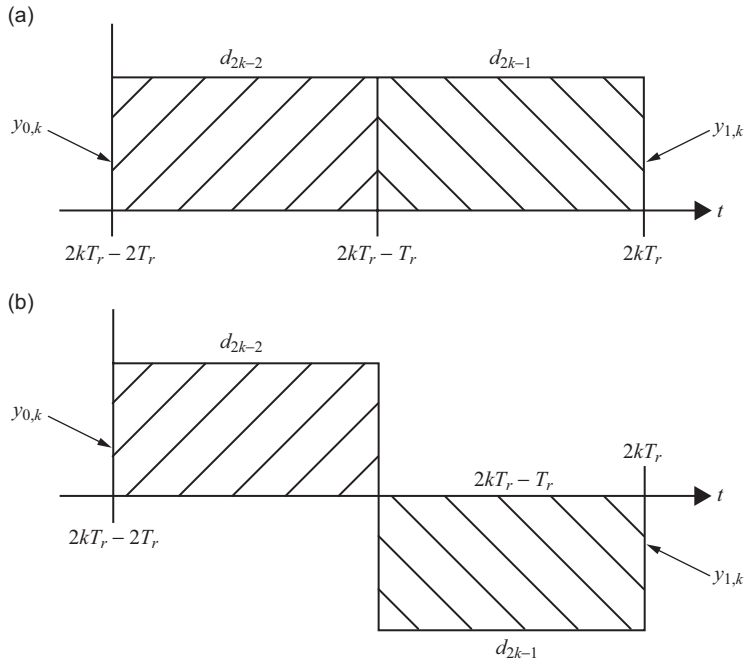


Fig. 7-2. Signal component of the I&D outputs $y_{0,k}$ and $y_{1,k}$ when the SSME assumed data rate is half of the true data rate for the case of (a) identical and (b) different adjacent data symbols.

will be a measure of the noise power. However, when two adjacent data symbols are different, as in Fig. 7-2(b), the opposite scenario takes place, i.e., $|u_k^+|^2$ becomes a measure of the noise power whereas $|u_k^-|^2$ becomes a measure of the signal-plus-noise power. This will result in a severely degraded estimate of the SNR since half of the time adjacent data symbols will be the same and half of the time they will be different. (The reason for this is that the data sequence is assumed to come from an independent, identically distributed (iid) source [1].) This degradation may even lead to negative estimates of the SNR which are clearly absurd.

For the purpose of data rate estimation, this degradation can be used to indicate that the assumed data rate of the SSME system was lowered excessively by one step. The elegance of this method of estimating the data rate is the rapid degradation that is expected once the assumed data rate has been lowered beyond the true data rate. Recall from Eq. (7-4) that up until the true data rate is reached, the mean of the SNR estimate will *increase* by a factor of B until the true SNR is reached. Once the assumed data rate is lowered by one more step, however, the mean of the SNR estimate will *decrease* significantly. Hence, the

SSME provides us with a way to estimate the data rate via a sharp transition in the estimate of the SNR.

An algorithm to estimate the data rate based on this phenomenon is presented below.

7.1.3 SSME Data Rate Estimation Algorithm

(1) Assume that the data rate is the maximum rate, i.e., set $\mathcal{R}_s = B^{\ell_{\max}}\mathcal{R}_b$. Run the SSME and compute an estimate of the mean of the SNR and call it $\hat{\mu}_{\hat{R}_0}$. Set $i = 1$.

(2) Lower the SSME data rate by a factor of B, i.e., set $\mathcal{R}_{s,\text{new}} = (1/B)\mathcal{R}_{s,\text{old}}$. Compute an estimate of the SNR mean and call it $\hat{\mu}_{\hat{R}_i}$.

(3) If $\hat{\mu}_{\hat{R}_i} \geq \hat{\mu}_{\hat{R}_{i-1}}$, then increment i by 1 and go to Step (2). Otherwise stop and estimate the SNR to be $\hat{\mu}_{\hat{R}} = \hat{\mu}_{\hat{R}_{i-1}}$ and the data rate to be $\hat{\mathcal{R}} = B^{\ell_{\max}-(i-1)}\mathcal{R}_b$.

In practice, the estimate of the SNR mean $\hat{\mu}_{\hat{R}_i}$ is computed as an *ensemble average* of observed SNR estimates $\hat{R}_i$ calculated over several *blocks* of the received signal. If a large enough ensemble of blocks is used, then we will have $\hat{\mu}_{\hat{R}_i} \approx E[\hat{R}_i]$ as desired.

It should be noted that this algorithm terminates as soon as $\hat{\mu}_{\hat{R}_i} < \hat{\mu}_{\hat{R}_{i-1}}$. In other words, the assumed data rate of the SSME is lowered only until the condition $\hat{\mu}_{\hat{R}_i} \geq \hat{\mu}_{\hat{R}_{i-1}}$ is not satisfied. Although this approach works in theory assuming that the number of observations is large enough, in practice this can often lead to a premature termination of the algorithm depending on the value of the *variance* of the SSME SNR estimate. (See [3] and Chapter 6 for more details.) For cases where the SNR is low, such as often occurs in the Deep Space Network (DSN), this can lead to a perturbation in the calculation of the mean of the SNR such that the condition $\hat{\mu}_{\hat{R}_i} < \hat{\mu}_{\hat{R}_{i-1}}$ will occur before it should, causing the algorithm to halt prematurely.

Since we expect the largest SNR to occur when the assumed data rate is equal to the true data rate, one alternative to this algorithm is to run the SSME for all data rates and estimate the data rate as the one yielding the largest SNR mean. This forms the basis for the GLRT-type data rate estimation algorithm presented below.

7.1.4 GLRT-Type SSME Data Rate Estimation Algorithm

(1) Run the SSME for all data rates and (as before) let $\hat{\mu}_{\hat{R}_i}$ denote the estimate of the mean of the SNR for the ith *largest* data rate.

(2) Define the optimal index i_{opt} to be $i_{\text{opt}} \triangleq \underset{0 \le i \le \ell_{max}}{\text{argmax}}\ \hat{\mu}_{\hat{R}_i}$. Then, estimate the true SNR and data rate as follows:

$$\hat{\mu}_{\hat{R}} = \hat{\mu}_{\hat{R}_{i_{\text{opt}}}}$$

$$\hat{\mathcal{R}} = B^{\ell_{\max} - i_{\text{opt}}} \mathcal{R}_b$$

For a traditional GLRT estimator, the conditional-likelihood function (CLF) [4] of the observables is *maximized* over the unknown parameters, as opposed to being *averaged* over them as is done in ML estimation. In that sense, this algorithm is a GLRT-like approach in that the SNR is chosen to be the maximum value obtained over the *unknown parameter* of the data rate. The data rate, in turn, is estimated as the rate that yields the largest SNR mean.

As will be shown in Section 7.4 through simulations, the GLRT-type data rate estimation algorithm outperforms the algorithm of Section 7.1.3 for low SNR when the true data rate is the lowest data rate. The reason for this is that this algorithm calculates an estimate of the SNR for *all* rates and doesn't prematurely terminate as the previous algorithm may do.

Prior to showing simulation results for these algorithms, we first investigate the effects of the presence of symbol-timing error on estimating the data rate. There, we show that these effects can seriously adversely affect the performance of the above proposed data rate estimation algorithms. In Sections 7.3.2 and 7.3.3, we present modifications to the algorithms of Sections 7.1.3 and 7.1.4, respectively, which account for the presence of symbol-timing error.

7.2 Effects of Symbol-Timing Error on Estimating the Data Rate

In the previous section, we assumed that the symbol-timing error or jitter ε was zero. From Chapter 6, it is known that the presence of jitter will have the effect of degrading the estimate of the SNR of the SSME. Heuristically speaking, the reason for this is that the half-symbol I&D outputs will contain the contributions of two adjacent data symbols. As the data symbols are iid, the signal components of the I&D outputs will be degraded similarly to the way in which they were degraded in Section 7.1.2 when the assumed data rate was lower than

the true data rate. This effect becomes more pronounced as ε reaches its worst case value of $1/2$.

To mitigate the effects of the presence of a nonzero ε, the approach suggested in Chapter 6 was to increase the data rate of the SSME system by a factor of L. By doing so, the vast majority of the half-symbol I&D outputs contain contributions due to only one data symbol, as desired. The effects due to those containing contributions from two adjacent data symbols become negligible, and so the oversampled estimator is then robust to the presence of jitter.

This principle of oversampling is used in the data rate estimation algorithms of Section 7.1. There, the oversampling factor is reduced at each stage until the largest SNR mean is obtained. The problem with these algorithms in the presence of symbol-timing error is that the SNR will appear to be degraded once the assumed data rate is lowered to the *true* data rate and not just afterward. In other words, for non-negligible values of the jitter, the largest SNR mean obtained will not occur when the SSME is operating at the true data rate, and so the data rate will be estimated erroneously. Furthermore, the SNR estimated will be far from its true value (approximately off by a factor of a power of B), since the data rate was incorrectly classified.

As an example to illustrate the adverse effects of symbol-timing error on the estimation of the data rate, consider the special case where $\varepsilon = 1/4$ and we have BPSK data as in the example in Section 7.1.2. Suppose that the system data rate of the SSME is equal to that of the true data rate. Then depending upon whether adjacent data symbols are the same or different, the signal portions of the I&D outputs $y_{0,k}$ and $y_{1,k}$ will be unaltered or degraded, respectively, as shown in Fig. 7-3.

Just as with the example considered in Section 7.1.2, when two adjacent data symbols are the same as in Fig. 7-3(a), we will obtain a valid contribution to the SNR estimate, since $y_{0,k}$ and $y_{1,k}$ will contain the same signal component support and polarity. However, when the adjacent data symbols are different as in Fig. 7-3(b), then we will have $y_{0,k} = 0$, which will severely degrade the SNR estimate. The reason for this is that, in this case, neither $|u_k^+|^2$ will be a good measure of the signal-plus-noise powers nor will $|u_k^-|^2$ be a good measure of the noise power. Instead, $|u_k^+|^2$ and $|u_k^-|^2$ will be measures of essentially the same quantity, namely a combination of *half* of the signal power together with the *full* noise power. This will result in a poor estimate of the SNR.

7.2.1 Accounting for the Symbol-Timing Error

To account for the presence of symbol-timing error, typically a data-transition tracking loop (DTTL) is used (see [5] and Chapter 10 for more details). However, a typical DTTL requires knowledge of *both* the carrier phase and data rate in order to operate properly. Thus, it appears as though there is a

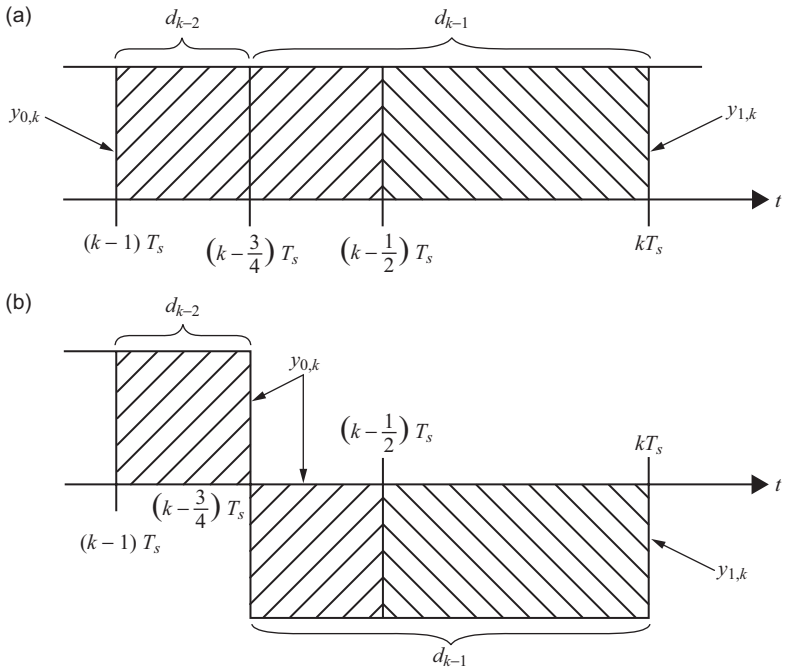


Fig. 7-3. Signal component of the I&D outputs $y_{0,k}$ and $y_{1,k}$ when the symbol timing error is $\varepsilon = 1/4$ for the case of (a) identical and (b) different adjacent data symbols.

dilemma. The data rate estimation algorithms of Sections 7.1.3 and 7.1.4 cannot reliably estimate the data rate (or the SNR for that matter) in the presence of symbol-timing error, and the symbol-timing error cannot be estimated without knowledge of the data rate (as well as the carrier phase).

To overcome this dilemma, we will exploit the fact that on average the presence of symbol-timing error only has a deleterious effect on the SNR estimate as shown in Chapter 6. The approach that will be taken here is to *quantize* the assumed symbol-timing error to a finite number of levels. Then, for each data rate, the SSME is run for each quantized jitter value. The SNR then is estimated to be the largest SNR obtained while the jitter is estimated as the value that yielded the largest SNR mean. In this way, not only do we obtain an improved estimate of the SNR for each assumed data rate, but we also obtain a coarse estimate of the symbol-timing error itself.

Hence, we generalize the data rate estimation algorithms of Sections 7.1.3 and 7.1.4 to jointly estimate the data rate, SNR, and symbol-timing error. Even with a coarse quantization of the symbol-timing error, this leads to a rather robust estimation of the data rate in the presence of jitter, as will be shown through simulations in Section 7.4. Once a reliable estimate of the data rate

has been made, the DTTL then can be used to obtain a finer estimate of the symbol-timing error. Furthermore, the coarse estimate of the jitter can be used as an initial condition for the DTTL which may reduce the computation time required for convergence.

It should be noted that this approach is different from the one suggested in Chapter 10, Section 10.8, in which oversampling is used to obtain a coarse estimate of the symbol-timing error. There, the data rate is assumed to be known, and the jitter is estimated by exploiting the fact that the presence of symbol-timing error becomes less noticeable as the oversampling ratio $L \to \infty$. This approach doesn't necessitate a modification to the SSME structure shown in Fig. 7-1, whereas the method suggested here does, as we show in the next section.

7.3 Quantization of the Symbol-Timing Error

As the data rate of the received signal is not known a priori, at the receiver, we are at liberty to independently quantize only the symbol-timing error corresponding to one specific data rate. The reason for this is that, by quantizing the jitter corresponding to one data rate, the quantized jitters for the remaining rates are automatically determined. In order to ensure that we have, say, at least $N_{\hat{\varepsilon},b}$ quantization levels for all rates, we must quantize the symbol-timing error corresponding to the *highest rate* by at least $N_{\hat{\varepsilon},b}$ levels. The reason for this is that if the highest data rate symbol-timing error is quantized to $N_{\hat{\varepsilon},b}$ steps, then the number of jitter steps at the next lowest rate will be $BN_{\hat{\varepsilon},b}$. By inductive argument, the number of quantization levels of the symbol-timing error at the kth lowest data rate will be $B^k N_{\hat{\varepsilon},b}$.

Following this logic, at the receiver, the symbol-timing error ε will be assumed to be *uniformly* quantized to $\hat{\varepsilon} = n/N_{\hat{\varepsilon},s}$ for some $0 \leq n \leq N_{\hat{\varepsilon},s} - 1$, where we have

$$N_{\hat{\varepsilon},s} = B^{\ell_{\max} - \ell_s} N_{\hat{\varepsilon},b} \tag{7-5}$$

Here, $N_{\hat{\varepsilon},b}$ denotes the *basic* number of jitter quantization steps (i.e., the number of steps at the *highest* data rate), whereas $N_{\hat{\varepsilon},s}$ denotes the *system* number of jitter quantization steps (i.e., when the assumed data rate power is ℓ_s).

Since the number of quantization steps increases as the assumed data rate decreases, it is tempting to think that we will always obtain a better estimate of the data rate, SNR, and symbol-timing error for lower true data rates than for higher rates. However, this is offset by the fact that, for a fixed observation time interval, we will obtain a larger number of observations for higher true data rates than for lower ones. Hence, we have an

implicit trade-off between the number of signal observations and the number of jitter quantization levels for each true data rate.

One of the advantages of uniformly quantizing the symbol-timing error to $N_{\hat{\varepsilon},s}$ steps as in Eq. (7-5) is that it leads to an efficient *all-digital* implementation of the SSME system, as we now proceed to show.

7.3.1 All-Digital Implementation of the SSME-Based Data Rate Estimator

Suppose that, prior to processing the received signal $\tilde{r}(t)$ from Eq. (7-1) through the SSME, it is *finely* integrated and sampled to obtain the discrete-time signal y_m using the system of Fig. 7-4. Here, $T_{\min}$ is the *time resolution period* given to be

$$T_{\min} \triangleq \frac{T_b}{B^{\ell_{\max}} N_{\hat{\varepsilon},b}} = \frac{1}{N_{\hat{\varepsilon},b}\,(B^{\ell_{\max}} \mathcal{R}_b)}$$

Note that $T_{\min}$ is $N_{\hat{\varepsilon},b}$ times *smaller* than the *shortest* possible data symbol interval. Equivalently, $1/T_{\max}$ is $N_{\hat{\varepsilon},b}$ times larger than the highest possible data rate, as can be seen from Eq. (7-2).

To generalize the SSME structure of Fig. 7-1 to account for the quantized symbol-timing error, it is also necessary to generalize it to account for computing an ensemble average of the observed SNRs. Recall from Section 7.1.3 that an ensemble average of the observed SNRs is required in order to estimate the mean of the SNR of the SSME system. To do this, we partition the discrete-time signal y_m into blocks over which the SNR is to be computed. For each block, the SSME computes an estimate of the SNR, and then an ensemble average of the SNR is computed over the blocks.

Let N_{obs} denote the basic number of symbols to observe per block to obtain an SNR estimate (i.e., the number of symbols to observe per block at the *lowest*

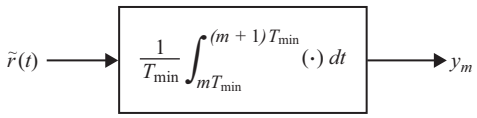


Fig. 7-4. System to finely integrate and sample the continuous-time signal $\tilde{r}(t)$ to obtain the high rate discrete-time signal y_m.

rate) and let N_b denote the number of blocks over which to compute an ensemble average of the SNR. Then, an all-digital implementation of the SSME system of Fig. 7-1 that accounts for the quantized symbol-timing error and ensemble averaging of the observed SNRs is shown in Fig. 7-5.

There are several things to note regarding the structure shown in Fig. 7-5. First, notice that the I&D half symbol integrators from Fig. 7-1 can be replaced with discrete summations, which is analogous to the sampled version of the SNR estimator discussed in Section 6.1.1.[2] Furthermore, note that all of the signals starting from the half-symbol integrator outputs are indexed with a semicolon followed by n. This notation was chosen here to reflect the fact that these quantities are parameterized by the quantized symbol-timing error $\hat{\varepsilon} = n/N_{\hat{\varepsilon},s}$, where the parameter n is an integer in the range

$$0 \leq n \leq N_{\hat{\varepsilon},s} - 1 \Longleftrightarrow 0 \leq n \leq B^{\ell_{\max}-\ell_s} N_{\hat{\varepsilon},b} - 1$$

Finally, note that to form a single SNR estimate, a total of $B^{\ell_s} N_{\text{obs}}$ samples are squared and accumulated. This was chosen as such here to keep the total observation time interval or epoch per block *fixed*.

By tracing the temporal indices m, k, and q from Fig. 7-5 backwards, it can be seen that, in order to have $0 \leq q \leq N_b - 1$ as desired, we need

$$0 \leq k \leq \left(B^{\ell_s} N_{\text{obs}}\right) N_b - 1$$

From this, it is clear that k must vary over an interval of N_b blocks each of size $B^{\ell_s} N_{\text{obs}}$, as desired and expected. Finally, from this range of the index k, in order to be able to accommodate all $N_{\hat{\varepsilon},s}$ values of the parameter n, it can be shown that the time index m should vary over the interval

$$0 \leq m \leq \left(N_{\hat{\varepsilon},b} B^{\ell_{\max}}\right) \left(N_{\text{obs}} N_b + 1\right) - 2 \tag{7-6}$$

To incorporate the estimation of the quantized symbol-timing error, the only required modification to the data rate estimation algorithms of Sections 7.1.3 and 7.1.4 is that the SNR estimate $\hat{\mu}_{\hat{R};n}$ must be calculated for each n. For a

[2] If the pulse shape used has some other piecewise constant shape such as the Manchester pulse, then the half-symbol integrators should be replaced with equivalent half-symbol digital matched filters [1]. In this way, the pulse shape also can be initially classified using an appropriate SSME for each possible pulse shape. This coarse pulse shape estimate can be used to judge the confidence of the estimation process when compared with the statistically optimal ML estimation approach of [6] and Chapter 5.

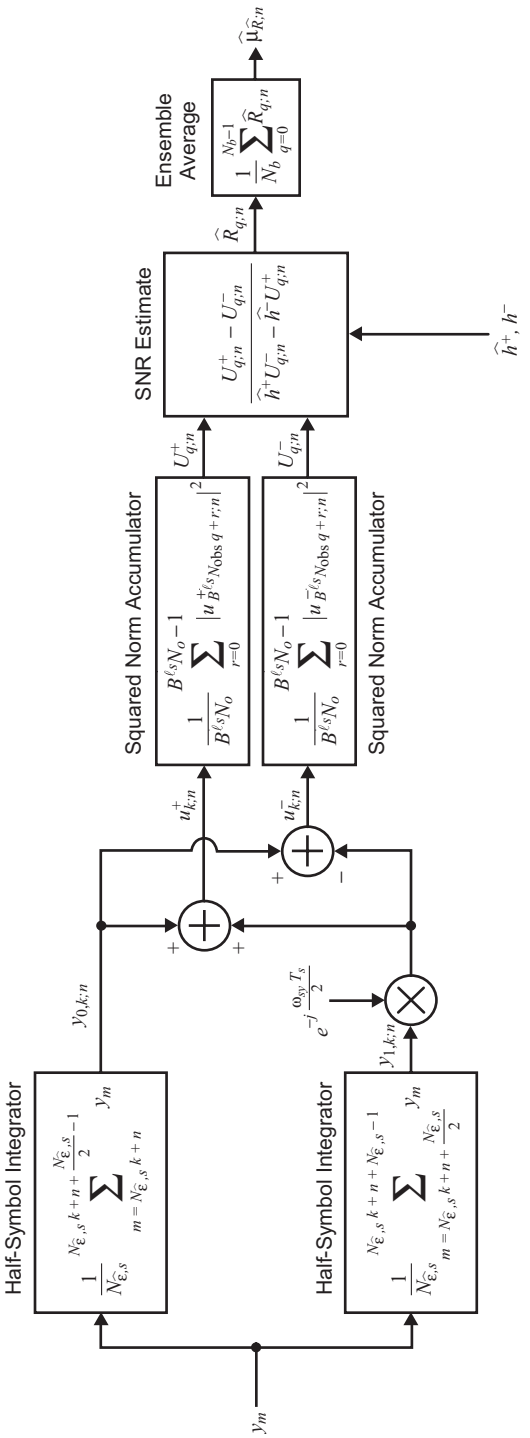


Fig. 7-5. Digital implementation of the SSME-based data rate estimation system that accounts for a quantized symbol-timing error and ensemble averaging of the observed SNRs.

fixed assumed data rate, the SNR is chosen to be the largest value of $\hat{\mu}_{\hat{R};n}$ while n is chosen to be the maximizing value of $\hat{\mu}_{\hat{R};n}$. This modification is described in the following algorithms.

7.3.2 SSME Data Rate/SNR/Symbol-Timing Error Estimation Algorithm

(1) Calculate the sequence y_m from Fig. 7-4 over the range of values given in Eq. (7-6).

(2) Run the SSME of Fig. 7-5 at the highest data rate $\mathcal{R}_s = B^{\ell_{\max}}\mathcal{R}_b$. Calculate $\hat{\mu}_{\hat{R};n}$ for all n and define $n_0 \stackrel{\triangle}{=} \underset{n}{\operatorname{argmax}}\, \hat{\mu}_{\hat{R};n}$ and $\hat{\mu}_{\hat{R}_0} \stackrel{\triangle}{=} \hat{\mu}_{\hat{R};n_0}$. Set $i = 1$.

(3) Lower the assumed data rate by one step, i.e., set $\mathcal{R}_{s,\text{new}} = (1/B)\mathcal{R}_{s,\text{old}}$, and run the SSME. Calculate $\hat{\mu}_{\hat{R};n}$ for all n and define $n_i \stackrel{\triangle}{=} \underset{n}{\operatorname{argmax}}\, \hat{\mu}_{\hat{R};n}$ and $\hat{\mu}_{\hat{R}_i} \stackrel{\triangle}{=} \hat{\mu}_{\hat{R};n_i}$.

(4) If $\hat{\mu}_{\hat{R}_i} \geq \hat{\mu}_{\hat{R}_{i-1}}$, increment i by 1 and go to Step (3). Otherwise, estimate the data rate, SNR, and symbol-timing error as follows:

$$\hat{\mathcal{R}} = B^{\ell_{\max}-(i-1)}\mathcal{R}_b$$

$$\hat{\mu}_{\hat{R}} = \hat{\mu}_{\hat{R}_{i-1}}$$

$$\hat{\varepsilon} = \frac{n_{i-1}}{B^{\ell_{\max}-(i-1)}N_{\hat{\varepsilon},b} - 1}$$

As mentioned above, for each assumed data rate, the SSME is run for each value of the quantized symbol-timing error. The SNR and jitter for that data rate then are estimated to be the largest SNR and the jitter value leading to this maximum SNR. Like the algorithm of Section 7.1.3, this data rate estimation technique halts as soon as the condition $\hat{\mu}_{\hat{R}_i} \geq \hat{\mu}_{\hat{R}_{i-1}}$ is not satisfied. This may lead to a premature termination of the algorithm as described in Section 7.1.3. To prevent a premature halting of the algorithm, a GLRT-type modification to the algorithm of Section 7.3.2 is proposed, similar to what was proposed in Section 7.1.4.

7.3.3 GLRT-Type SSME Data Rate/SNR/Symbol-Timing Error Estimation Algorithm

(1) Calculate the sequence y_m from Fig. 7-4 over the range of values given in Eq. (7-6).

(2) Run the SSME for all data rates and all possible quantized symbol-timing error values. Let $\hat{\mu}_{\hat{R};(i,n)}$ denote the estimate of the mean of the SNR for the ith largest data rate with quantized jitter value n. (Here we have $0 \leq i \leq \ell_{\max} - 1$ and $0 \leq n \leq B^{\ell_{\max}-i} N_{\hat{\varepsilon},b} - 1$.)

(3) Let i_{opt} and n_{opt} denote the indices for which $\hat{\mu}_{\hat{R};(i,n)}$ reaches its maximum value, i.e., i_{opt} and n_{opt} are such that $\hat{\mu}_{\hat{R};(i_{\text{opt}},n_{\text{opt}})} = \max_{i,n} \hat{\mu}_{\hat{R};(i,n)}$. Then, estimate the data rate, SNR, and symbol-timing error as follows:

$$\hat{\mathcal{R}} = B^{\ell_{\max}-i_{\text{opt}}} \mathcal{R}_b$$

$$\hat{\mu}_{\hat{R}} = \hat{\mu}_{\hat{R};(i_{\text{opt}},n_{\text{opt}})}$$

$$\hat{\varepsilon} = \frac{n_{\text{opt}}}{B^{\ell_{\max}-i_{\text{opt}}} N_{\hat{\varepsilon},b} - 1}$$

This GLRT-type estimation algorithm is based on the principle that the true data rate and symbol-timing error should yield the largest value of the mean of the SNR. Incorrect values of these quantities, on the other hand, should lead to a degraded estimate of the SNR mean. As opposed to the previous algorithm, which lowers the assumed data rate until the SNR decreases, this algorithm computes the SNR for all data rates and all jitter values. The advantage to this is that it can prevent the algorithm from prematurely terminating, which easily can happen when the true SNR is low. This is especially the case when the true data rate is low, as we show through simulations in the next section.

7.4 Simulation Results for the SSME-Based Estimation Algorithms

In order to properly evaluate the performance of the estimation algorithms of Sections 7.3.2 and 7.3.3, we must consider different metrics for each of the parameters that we wish to estimate. Prior to presenting simulation results, we introduce these metrics and justify their usage here.

7.4.1 Performance Metrics Used for Evaluating the Estimation Algorithms

For all of the following measures used, we assume that the estimation algorithms have each been run for a total of N_t trials. Parameters estimated at the nth trial (where $0 \leq n \leq N_t - 1$ for simplicity) are denoted with a superscript surrounded by parentheses. For example, the data rate estimated at the nth trial is denoted as $\hat{\mathcal{R}}^{(n)}$.

7.4.1.1. Probability of Data Rate Misclassification. In order to assess the performance of the algorithms with respect to estimating the data rate, one valid measure of performance is the empirical probability of data rate misclassification, which is defined below:

$$P_m \triangleq \frac{1}{N_t} \sum_{n=0}^{N_t-1} \mathcal{I}\left(\hat{\mathcal{R}}^{(n)} \neq \mathcal{R}\right) \tag{7-7}$$

where $\mathcal{I}(X)$ is an indicator function that is unity if the event X is true and zero if X is false. From Eq. (7-7), it is clear that $0 \leq P_m \leq 1$ and that P_m is a linear measure of the number of times each algorithm fails to estimate the data rate correctly.

7.4.1.2. Mean-Squared SNR Decibel Estimation Error. To properly gauge the performance of the estimation algorithms with respect to estimating the SNR, we seek a metric that penalizes the error between the estimated and true SNRs based on the value of the true SNR. In particular, small differences in SNR should be penalized more so if the true SNR is small than if it is large. For example, if the true SNR is 1 and the SNR is estimated to be 0.7, then it is reasonable to penalize this error more so than if the true SNR were 100 and the estimated SNR were 97.

One metric that penalizes the error in the SNR in such a way is the mean-squared error between the estimated and true SNRs in decibels (dB). This measure is the mean-squared SNR dB estimation error and is given below as follows:

$$\xi_R \triangleq \frac{1}{N_t} \sum_{n=0}^{N_t-1} \left|\hat{\mu}_{\hat{R}}^{(n)}\,(\mathrm{dB}) - R\,(\mathrm{dB})\right|^2 = \frac{1}{N_t} \sum_{n=0}^{N_t-1} \left|10\log_{10}\left(\frac{\hat{\mu}_{\hat{R}}^{(n)}}{R}\right)\right|^2 \tag{7-8}$$

From Eq. (7-8), it is clear that for low true SNR a deviation from the true SNR is penalized more so than for high true SNR. For the example from above, the mean-squared SNR dB error for the case of a true SNR of 1 and an estimated

SNR of 0.7 is 2.399, whereas the error for the case of a true SNR of 100 and an estimated SNR of 97 is 0.017.

7.4.1.3. Mean-Squared Minimum Distance Symbol-Timing Estimation Error. In order to quantify the performance of each of the algorithms with respect to symbol-timing error, it is tempting to consider a simple mean-squared error measure between the true and estimated symbol-timing error, which is given below:

$$\xi_\varepsilon = \frac{1}{N_t} \sum_{n=0}^{N_t-1} \left| \varepsilon - \hat{\varepsilon}^{(n)} \right|^2 \tag{7-9}$$

The problem with using the metric given in Eq. (7-9) is that both symbol-timing errors are assumed to be in the interval $[0, 1)$. However, in reality, each symbol-timing error can be shifted by any integer amount without loss of generality. For example, if the estimated jitter is $\hat{\varepsilon}^{(n)} = 0.75$, this is also equivalent to $\hat{\varepsilon}^{(n)} = \cdots, -1.25, -0.25, 0.75, 1.75, 2.75, \cdots$. This shifting property can cause the metric given in Eq. (7-9) to be overly pessimistic in certain cases.

To see this, consider the case where the true symbol-timing error is $\varepsilon = 0.1$ and the estimated value is $\hat{\varepsilon}^{(n)} = 0.9$. Using Eq. (7-9), we find that $\xi_\varepsilon = 0.64$. However, this error is overly pessimistic, since there is a shifted version of the estimated symbol-timing error (namely $\hat{\varepsilon}^{(n)} = -0.1$) that is closer to the true value of $\varepsilon = 0.1$. This is illustrated in Fig. 7-6. Using this shifted value of $\hat{\varepsilon}^{(n)}$, we obtain $\xi_\varepsilon = 0.04$, which is a more appropriate value for the error between ε and $\hat{\varepsilon}^{(n)}$ in this case.

Thus, a more appropriate measure of the jitter estimation error is to find the *minimum distance* between the true and estimated jitters as the jitters vary over all possible shifted values. Equivalently, we can fix the true jitter to be in the interval $[0, 1)$ and find the shifted version of the estimated jitter that is closest to the true jitter. In other words, a more appropriate measure of the jitter estimation error is to replace each term of the summation in Eq. (7-9) with a term of the form

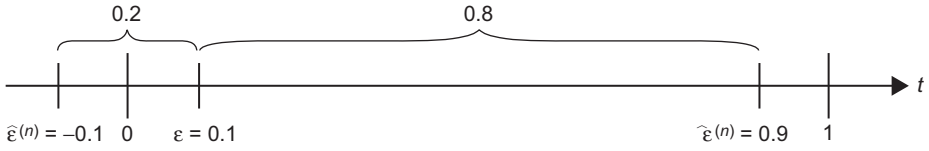


Fig. 7-6. Example of a pessimistic value for the jitter estimation error ξ_ε from Eq. (7.9).

$$\min_{\ell \in \mathbb{Z}} \left\{ \left| \varepsilon - \left(\ell + \hat{\varepsilon}^{\,(n)} \right) \right|^2 \right\} \tag{7-10}$$

where we assume $\varepsilon, \hat{\varepsilon}^{\,(n)} \in [0, 1)$. Fortunately, under the assumption that $\varepsilon, \hat{\varepsilon}^{\,(n)} \in [0, 1)$, we need not look over all values of $\ell \in \mathbb{Z}$ in Eq. (7-10). In particular, we need only look for the minimum value over $\ell = -1, 0, 1$. To see this, note that we have

$$0 \leq \varepsilon < 1, \quad 0 \leq \hat{\varepsilon}^{\,(n)} < 1$$

from which we conclude

$$-1 < \varepsilon - \hat{\varepsilon}^{\,(n)} < 1$$

By adding $-\ell$ to all sides of the inequality, we have

$$-\ell - 1 < \varepsilon - \left(\ell + \hat{\varepsilon}^{\,(n)} \right) < -\ell + 1$$

Now, for $|\ell| \geq 2$, it can be shown that

$$\left| \varepsilon - \left(\ell + \hat{\varepsilon}^{\,(n)} \right) \right|^2 > 1 > \left| \varepsilon - \hat{\varepsilon}^{\,(n)} \right|^2$$

and so the term corresponding to $\ell = 0$ always has a smaller magnitude than those corresponding to $|\ell| \geq 2$. Hence the terms corresponding to $|\ell| \geq 2$ can be ignored in the expression of Eq. (7-10), leaving only $\ell = -1, 0, 1$.

Thus, to ascertain the performance of the algorithms with respect to symbol-timing error, we opted to use the following mean-squared *minimum distance* symbol-timing estimation error:

$$\xi_\varepsilon \triangleq \frac{1}{N_t} \sum_{n=0}^{N_t - 1} \min_{\ell = -1, 0, 1} \left\{ \left| \varepsilon - \left(\ell + \hat{\varepsilon}^{\,(n)} \right) \right|^2 \right\} \tag{7-11}$$

The minimization in each term of Eq. (7-11) ensures that we choose the closest (left, neutral, or right-shifted) estimated jitter to the true one.

We now proceed to present simulation results for the SSME-based data rate estimation algorithms of Sections 7.3.2 and 7.3.3.

7.4.2 Behavior of the SSME-Based Data Rate Estimation Algorithms as a Function of SNR

For all of the simulations considered here, from Eq. (7-1), the data constellation d_k used was quadrature phase-shift keying (QPSK) [1], and the residual frequency offset ω_r was set to zero. To test the data rate estimation algorithms of Sections 7.3.2 and 7.3.3, we opted to choose the following input parameters:

$$\begin{aligned} B &= 2 \\ \ell_{\max} &= 3 \\ N_{\hat{\varepsilon},b} &= 2 \\ N_{\text{obs}} &= 64 \\ N_b &= 16 \end{aligned}$$

It should be noted that the choice of N_{obs} and N_b here implies that we have an observation time epoch equal to $N_{\text{obs}}N_b = 1{,}024$ *lowest rate* symbols. This time epoch was fixed here for all possible data rates in order to reflect the fact that we are assumed to have no a priori knowledge of the data rate. As such, this intuitively implies that on average the SSME will be able to estimate the SNR more accurately for higher data rates. The reason for this is that, for a fixed time epoch, the SSME will have more observations the higher the data rate becomes. This will result not only in an increase in the accuracy of the SNR estimate for higher data rates, but also often in a better probability of misclassification and jitter estimation error, as will soon be shown.

For a preliminary set of simulations, suppose that the symbol-timing error is zero (i.e., $\varepsilon = 0$), and the true SNR R is varied from -10 dB to -3 dB.[3] In Fig.7-7, we have plotted the probability of misclassification, P_m from Eq. (7-7), as a function of SNR using (a) the algorithm of Section 7.3.2 and (b) the algorithm of Section 7.3.3. As can be seen, the algorithm of Section 7.3.2 outperforms that of Section 7.3.3 for the higher data rates, but fails to do so for the lower ones. The reason for this is that the algorithm of Section 7.3.2 will often prematurely terminate, which is beneficial for higher true data rates and detrimental for lower true ones.

One unusual phenomenon that can be observed from Fig. 7-7 is that the curves cross for different values of the true data rate. This appears counterintuitive, since we should expect the higher data rates to be classified correctly more often than the lower data rates (as there are a larger number of observations in

[3] The reason for varying the true SNR over such low values is to reflect the fact that, in the DSN, the SNR is typically rather small.

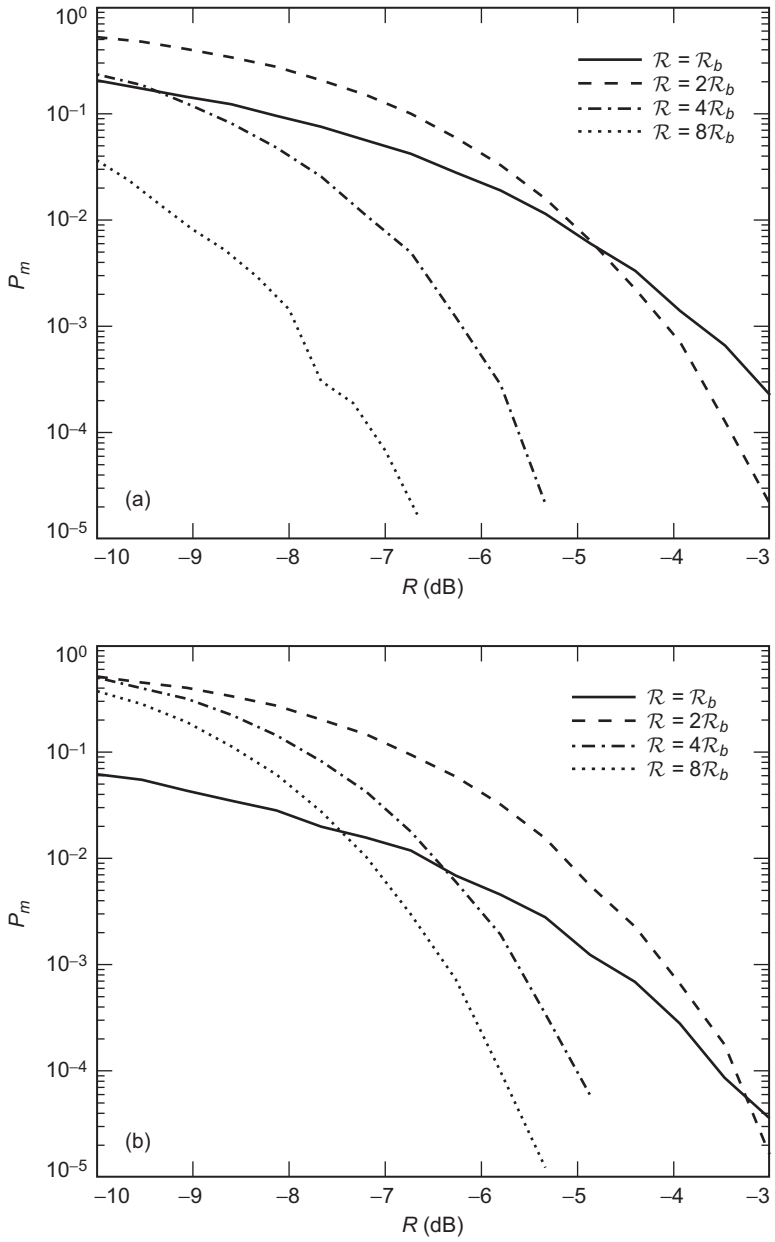


Fig. 7-7. Probability of data rate misclassification as a function of SNR using the algorithms of (a) Section 7.3.2 and (b) Section 7.3.3.

these cases). However, when the true SNR is low, the factor corresponding to the number of observations in the expression for the mean of the SSME SNR estimate becomes non-negligible (see the equation above Eq. (7-3) for more details). This most likely is the reason that the curves cross at lower true SNR. At higher true SNR, the mean of the SNR estimate becomes less sensitive to the number of observations and so we expect the higher rates to be classified correctly more often than the lower rates. This is indeed the case here, as can be seen in Fig. 7-7 when the true SNR is near -3 dB.

In order to accurately compare the two algorithms, one figure of merit that can be used is the *average* probability of misclassification, which we denote here by $\overline{P}_m$. If p_ℓ denotes the probability that the true data rate is $\mathcal{R} = B^\ell \mathcal{R}_b$ and $P_{m|\ell}$ denotes the probability of misclassification given that the true data rate is $B^\ell \mathcal{R}_b$, then by the theorem of total probability [7], we have

$$\overline{P}_m = \sum_{\ell=0}^{\ell_{\max}} p_\ell P_{m|\ell} \tag{7-12}$$

Assuming that the true data rates are equiprobable (i.e., $p_\ell = 1/(\ell_{\max}+1)$ for all ℓ), Eq. (7-12) becomes

$$\overline{P}_m = \frac{1}{\ell_{\max}+1} \sum_{\ell=0}^{\ell_{\max}} P_{m|\ell}$$

A plot of $\overline{P}_m$ as a function of the true SNR R is shown in Fig. 7-8 for equiprobable data rates. From this, it can be seen that, for lower SNR, the algorithm of Section 7.3.2 yields a better average probability of misclassification, whereas for higher SNR (above about -7.3 dB), the algorithm of Section 7.3.3 performs better. Since the desired SNR for a DSN-type application is -6 dB or greater (in order to achieve good performance for the turbo codes expected to be used), this implies that the GLRT-type algorithm of Section 7.3.3 is best suited here.

To further compare the two algorithms, in Fig. 7-9 we have plotted the observed mean-squared SNR dB estimation error, ξ_R from Eq. (7-8), for the algorithms of (a) Section 7.3.2 and (b) Section 7.3.3. As can be seen, the estimation error always decreased monotonically with SNR for each data rate. Furthermore, it can be seen that the error decreased almost geometrically as the data rate increased. These two phenomena are consistent with the fact that the SSME yields a better estimate of the SNR as both the true SNR and number of observations increase.

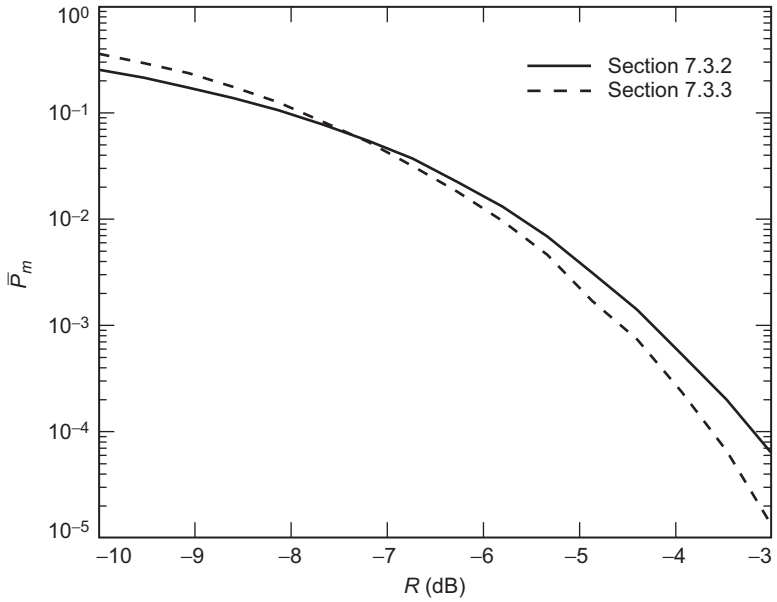


Fig. 7-8. Average probability of misclassification as a function of SNR for the algorithms of Sections 7.3.2 and 7.3.3.

Analogous to the average probability of misclassification $\overline{P}_m$ given in Eq. (7-12), we can quantitatively compare both algorithms in terms of the average mean-squared SNR dB estimation error $\overline{\xi}_R$ given by

$$\overline{\xi}_R = \sum_{\ell=0}^{\ell_{\max}} p_\ell \xi_{R|\ell} \tag{7-13}$$

where $\xi_{R|\ell}$ is the mean-squared SNR dB error given that the true data rate is $\mathcal{R} = B^\ell \mathcal{R}_b$. Assuming equiprobable data rates in Eq. (7-13), a plot of $\overline{\xi}_R$ as a function of the true SNR R is shown in Fig.7-10 for both algorithms. As can be seen, the GLRT-type algorithm of Section 7.3.3 always outperformed that of Section 7.3.2, although for larger SNR (near -3 dB), the two performed nearly identically. This is consistent with the intuition that the two algorithms should be performing increasingly similarly as the true SNR increases since the SNR estimates are more accurate in this case.

As a final measure of comparison between the two algorithms, the observed mean-squared minimum distance symbol-timing estimation error, ξ_ε from Eq. (7-11), is shown in Fig.7-11. From this, it can be seen that the algorithm of Section 7.3.2 yielded a good estimate for the higher data rates but suffered for

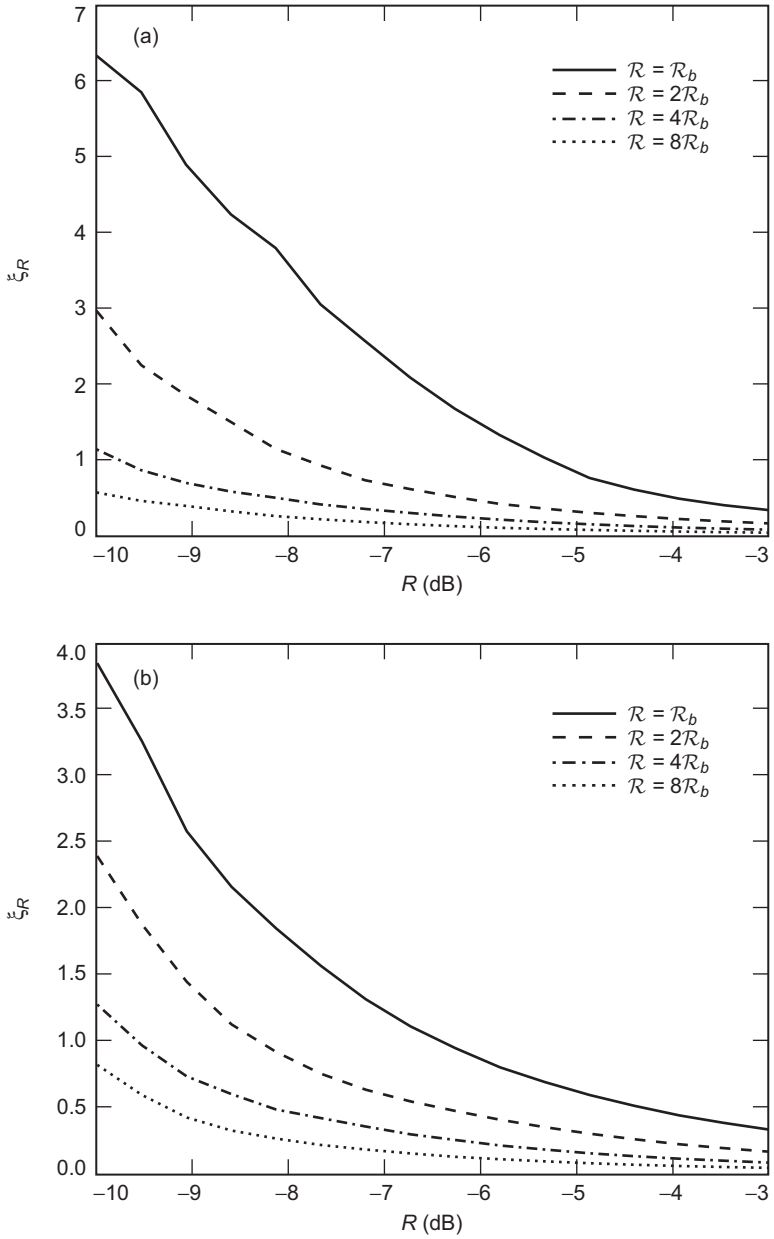


Fig. 7-9. Mean-squared SNR decibel estimation error as a function of the true SNR using the algorithms of (a) Section 7.3.2 and (b) Section 7.3.3.

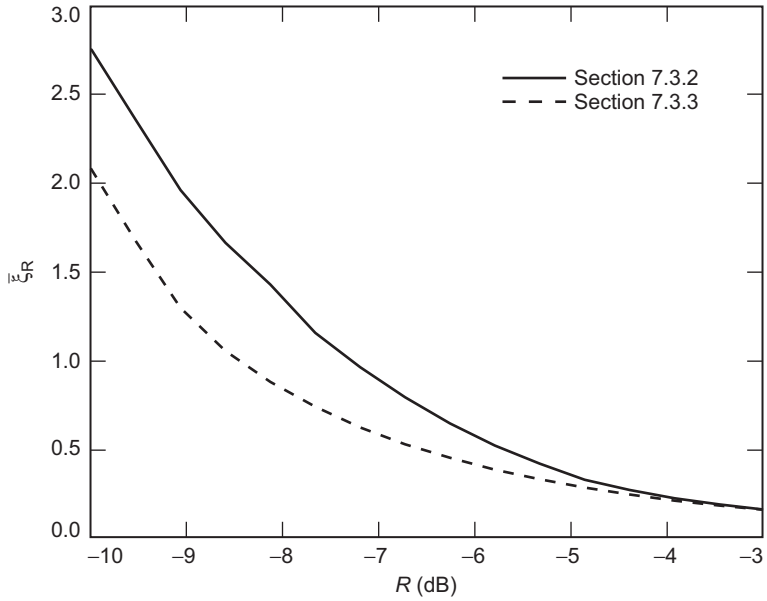


Fig. 7-10. Average mean-squared SNR decibel estimation error as a function of the true SNR for the algorithms of Sections 7.3.2 and 7.3.3.

the lower ones. This perhaps is due to the inherent premature halting possibility of the algorithm, as discussed earlier. For the algorithm of Section 7.3.3, it can be seen that at low SNR the error is large for all rates and that, with the exception of the lowest data rate, for a fixed SNR the error decreased as the rate increased.

As before, to quantitatively compare both algorithms, we can do so by computing the average mean-squared minimum distance symbol-timing estimation error $\overline{\xi}_\varepsilon$ given by

$$\overline{\xi}_\varepsilon = \sum_{\ell=0}^{\ell_{\max}} p_\ell \xi_{\varepsilon|\ell} \tag{7-14}$$

where $\xi_{\varepsilon|\ell}$ denotes the symbol-timing estimation error given that the true data rate is $\mathcal{R} = B^\ell \mathcal{R}_b$. Assuming equiprobable data rates in Eq. (7-14), a plot of $\overline{\xi}_\varepsilon$ as a function of the SNR R is shown in Fig. 7-12. From this, it can be seen that for low SNR the algorithm of Section 7.3.2 notably outperformed the algorithm of Section 7.3.3. Above about -7.1 dB, however, the opposite scenario took place. As the desired mode of operation for the autonomous radio is above -6 dB, this implies that once again the GLRT-type algorithm

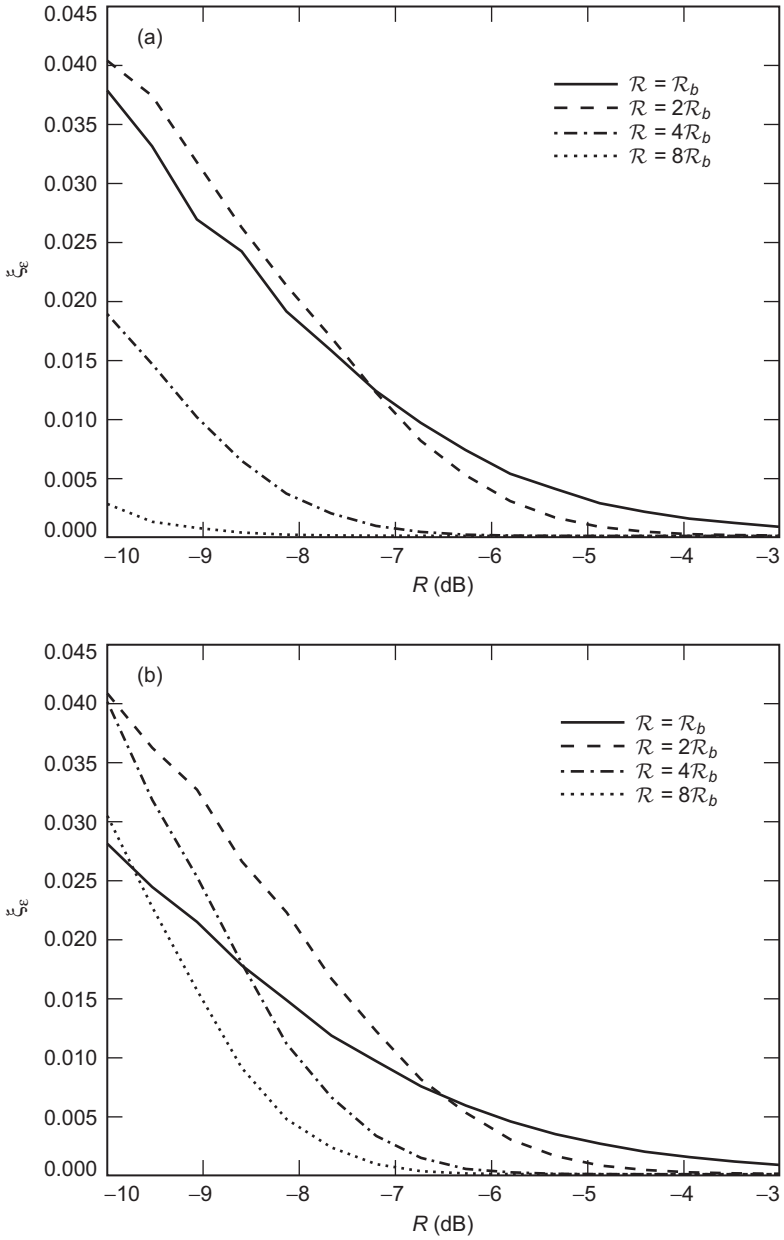


Fig. 7-11. Mean-squared minimum distance symbol timing estimation error as a function of SNR using the algorithms of (a) Section 7.3.2 and (b) Section 7.3.3.

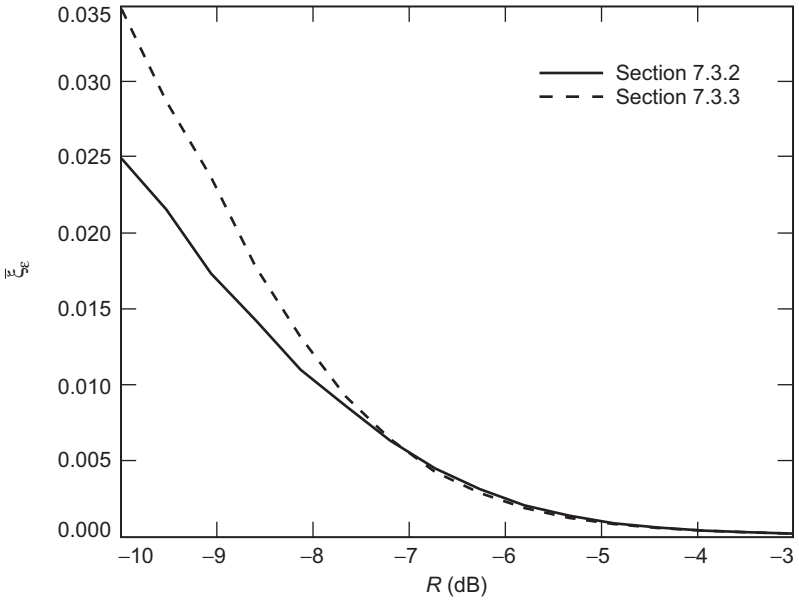


Fig. 7-12. Average mean-squared minimum distance symbol-timing estimation error as a function of SNR for the algorithms of Sections 7.3.2 and 7.3.3.

of Section 7.3.3 is best suited here. It should be noted, however, that these algorithms can be used to obtain only a coarse estimate of the symbol-timing error and that once the data rate has been successfully classified, a finer estimate of the jitter can be obtained through the use of a DTTL [5].

7.4.3 Behavior of the SSME-Based Data Rate Estimation Algorithms as a Function of Symbol-Timing Error

In the previous section, we considered the performance of the data rate estimation algorithms of Sections 7.3.2 and 7.3.3 for a varying SNR and a fixed symbol-timing error. Here, we investigate the performance of the algorithms as a function of the jitter for fixed SNR. As the target SNR for the autonomous radio for the DSN is above -6 dB (in order to achieve good performance from the turbo codes to be used for error correction), the SNR here was fixed at -6 dB.

To illustrate the effects of quantizing and coarsely estimating the symbol-timing error on estimating the data rate, suppose that the true data rate is $\mathcal{R} = 2\mathcal{R}_b$. Plots of the observed probability of misclassification are shown in Fig. 7-13 for (a) the algorithm of Section 7.3.2 and (b) the algorithm of Section 7.3.3. As can be seen, for both methods the probability appears to oscillate back and forth as the jitter varies. It can be seen that both plots

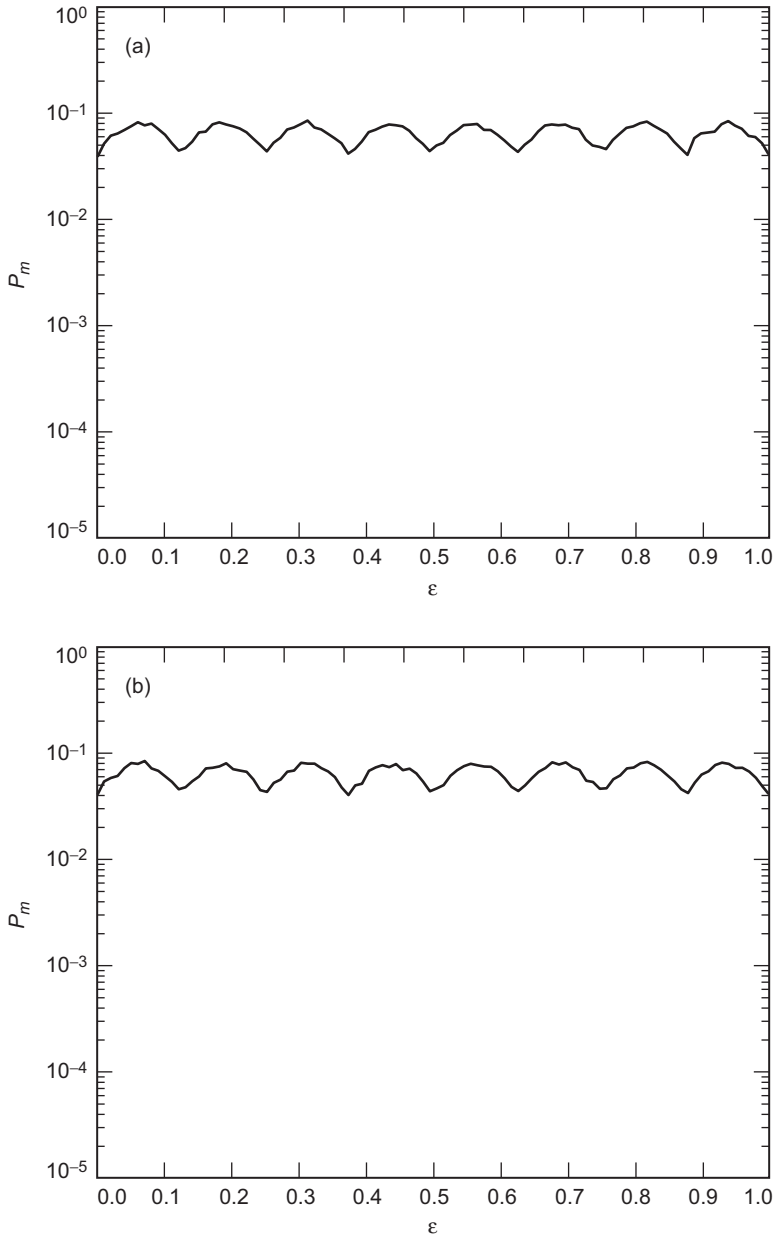


Fig. 7-13. Probability of data rate misclassification as a function of the jitter using the algorithms of (a) Section 7.3.2 and (b) Section 7.3.3. (The true data rate is $\mathcal{R} = 2\mathcal{R}_b$.)

appear to have eight equispaced local maxima. The reason for this is due to the quantization of the symbol-timing error. Recall that from Section 7.4.2 the basic number of quantization levels was $N_{\hat{\varepsilon},b} = 2$. This implies that at the true data rate $\mathcal{R} = 2\mathcal{R}_b = 2^1\mathcal{R}_b$ the symbol-timing error is quantized to $N_{\hat{\varepsilon},s} = 2^{\ell_{\max}-1}N_{\hat{\varepsilon},b} = 2^2 N_{\hat{\varepsilon},b} = 8$ steps by using Eq. (7-5). These steps are equispaced about the interval $[0, 1)$ and are of the form $n/8$ for $0 \leq n \leq 7$. Every time each of the data rate estimation algorithms is run, each method chooses the quantized value of the jitter that is the "best fit" in some sense to the true jitter. As the true jitter itself is varied, it is evident that there will be *ambiguous* values of the symbol-timing error that occur directly in between the quantized values. This is illustrated in Fig.7-14 for the case of 8 quantization steps here. From Fig. 7-13, it is clear that the probability of misclassification becomes locally maximal almost precisely at these ambiguous jitter value locations.

To further observe the effects of varying the symbol-timing error, a plot of the observed mean-squared SNR decibel error is shown in Fig. 7-15 for both algorithms. Note that, unlike the probability of misclassification, for both methods the error in estimating the SNR remains approximately constant as the jitter is varied. The reason for this robustness most likely comes from the fact that, with a sufficient number of quantization steps, the "best fit" jitter value to the true one chosen for the SSME will incur only a small degradation in the mean of the SNR estimate. See Chapter 6 for more details on the quantitative amount of this degradation.

As a final measure of the effects of varying symbol-timing error on the data rate estimation algorithms of Sections 7.3.2 and 7.3.3, a plot of the observed mean-squared minimum distance jitter estimation error for each algorithm is shown in Fig. 7-16. Like the probability of misclassification plots of Fig. 7-13, it can be seen that the error for both algorithms oscillates back and forth as the jitter varies. Also as before, each plot appears to have eight equispaced local maxima that occur approximately at the locations corresponding to the ambiguous values of the symbol-timing error. This observation is consistent with

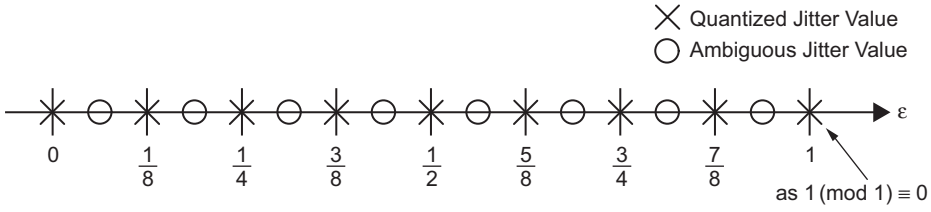


Fig. 7-14. Example showing the quantized symbol-timing error values for $N_{\hat{\varepsilon},s} = 8$ along with the *ambiguous* jitter values.

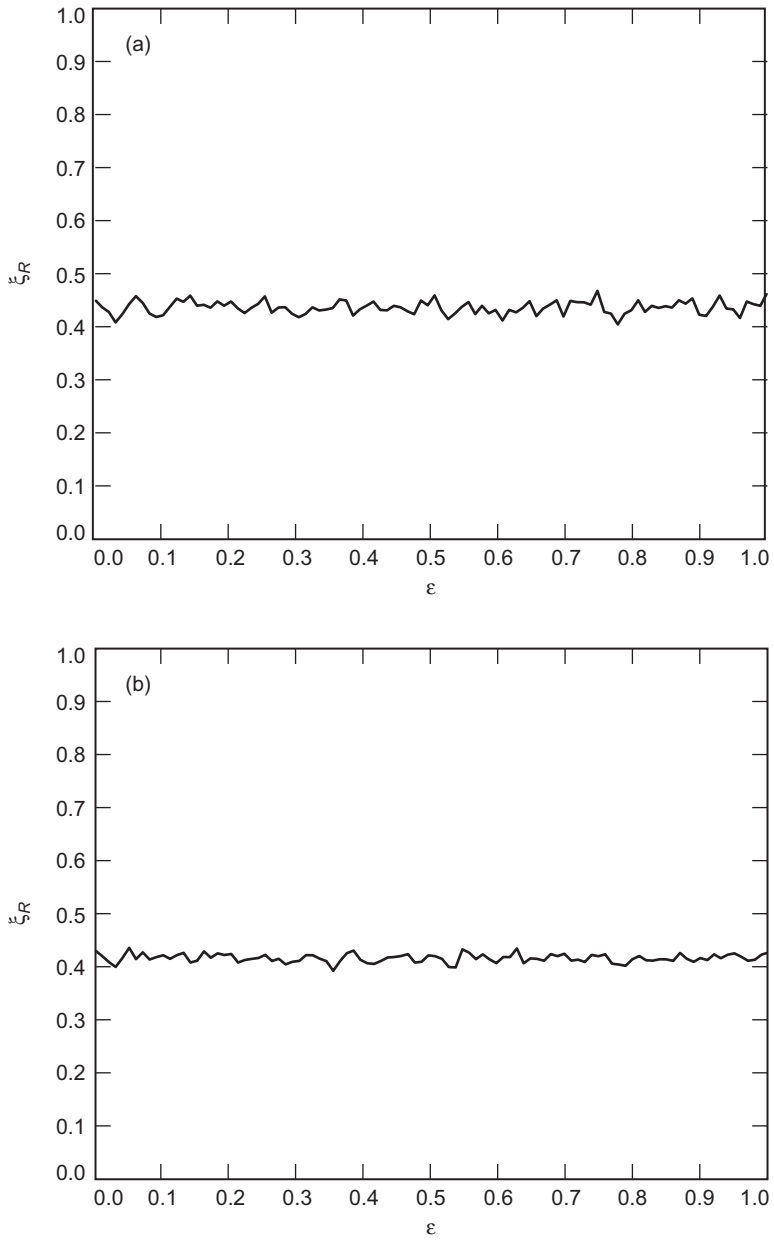


Fig. 7-15. Mean-squared SNR decibel estimation error as a function of the jitter using the algorithms of (a) Section 7.3.2 and (b) Section 7.3.3. (The true data rate is $\mathcal{R} = 2\mathcal{R}_b$.)

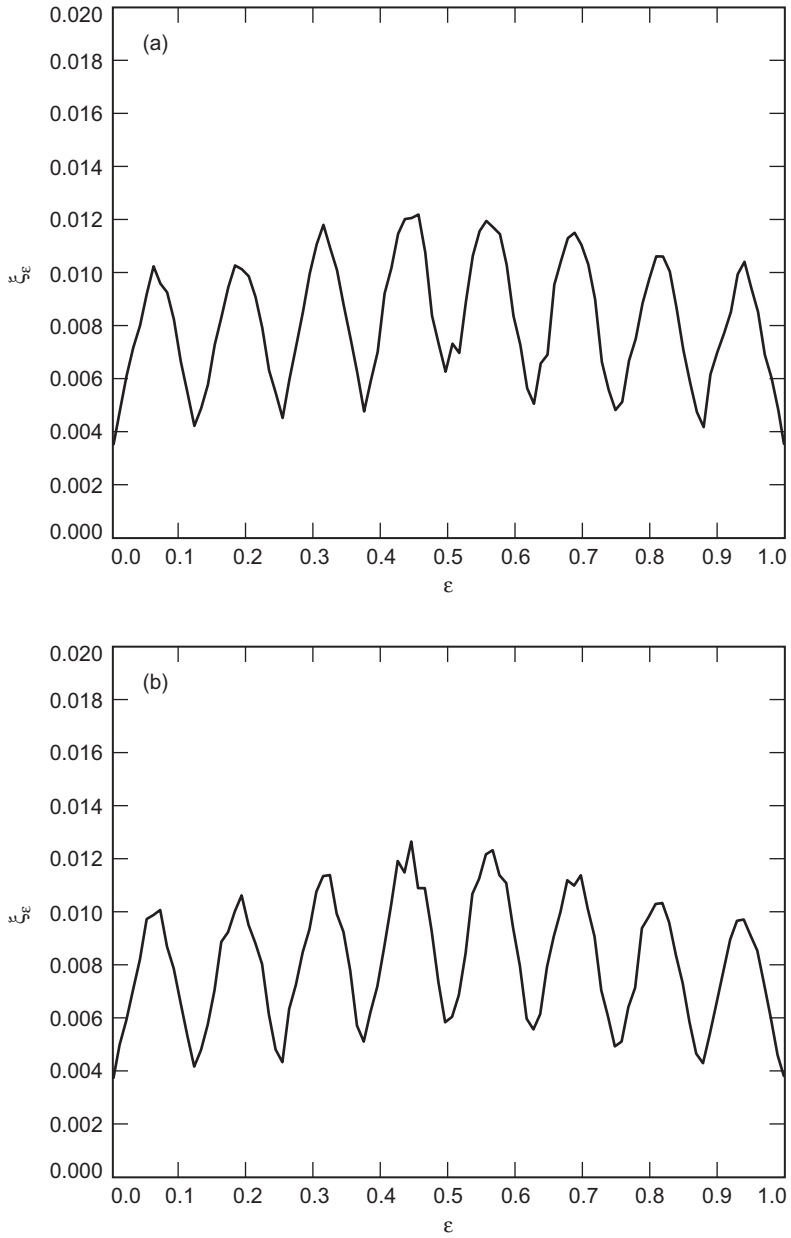


Fig. 7-16. The mean-squared minimum distance symbol-timing estimation error as a function of the jitter using the algorithms of (a) Section 7.3.2 and (b) Section 7.3.3. (The true data rate is $\mathcal{R} = 2\mathcal{R}_b$.)

the intuition that the estimation process should suffer the most degradation at the ambiguous jitter values. One new phenomenon that can be observed from the plots of Fig.7-16 is that, for both algorithms, the error appears symmetric about $\varepsilon = 1/2$ and seems to generally increase as $\varepsilon \to 1/2$ from either direction. The reason for this phenomenon is not clear at this point and requires further investigation.

At this point, a few comments are in order. Had the above simulations been run for another true data rate, say at $\mathcal{R} = 4\mathcal{R}_b$, then there would have been 4 ambiguous jitter values instead of 8, since $N_{\hat{\varepsilon},s} = 2^{3-2}N_{\hat{\varepsilon},b} = 4$ in this case. The same observations regarding the performance metrics would still hold true, with the exception that the degradation in performance due to fewer jitter quantization steps would be more pronounced. In general, with a true data rate of $\mathcal{R} = B^{\ell}\mathcal{R}_b$, the number of symbol-timing error quantization steps at the true data rate is $N_{\hat{\varepsilon},s} = B^{\ell_{\max}-\ell}N_{\hat{\varepsilon},b}$ from Eq. (7-5). This suggests that an implicit trade-off in performance exists between the data rate and the granularity of the symbol-timing error. For a fixed observation time epoch of the received signal, the higher the data rate, the more observations we have to help improve the estimate of the mean of the SNR of the SSME. However, at the same time, we also have an increased sensitivity to the symbol-timing error in this case. Conversely, the lower the data rate, the fewer samples there are to estimate the mean of the SNR. However, at the same time, we also have more robustness with respect to the symbol-timing error.

The effect of increasing the basic number of symbol-timing error quantization steps $N_{\hat{\varepsilon},b}$ is to increase the number of ambiguous jitter values but at the same time to decrease the degradation at these values. Thus, the estimation becomes more robust in this case. However, this comes at the price of increased computational complexity, as well as an increase in the oversampling rate of the received signal. For the Electra radio (see [2] and Chapter 2), the sampling rate is 4 times the highest data rate, and so the maximum value of $N_{\hat{\varepsilon},b}$ that can be used for this system is $N_{\hat{\varepsilon},b} = 4$. Although this value may appear to be small, for most applications this should be sufficient for estimating the data rate and SNR reasonably well. As mentioned above, once the data rate has been classified correctly, the symbol-timing error can be finely estimated through the use of a DTTL (see [5] and Chapter 10).

References

[1] M. K. Simon, S. M. Hinedi, and W. C. Lindsey, *Digital Communications Techniques: Signal Design and Detection*, Upper Saddle River, New Jersey: Prentice Hall PTR, 1994.

[2] S. F. Franklin, J. P. Slonski, Jr., S. Kerridge, G. Noreen, S. Townes, E. Schwartzbaum, S. Synnott, M. Deutsch, C. Edwards, A. Devereaux, R. Austin, B. Edwards, J. J. Scozzafava, D. M. Boroson, W. T. Roberts, A. Biswas, A. D. Pillsbury, F. I. Khatri, J. Sharma, and T. Komarek, "The 2009 Mars Telecom Orbiter Mission," *Proceedings of the IEEE Aerospace Conference*, Big Sky, Montana, pp. 437–456, March 2004.

[3] M. Simon and S. Dolinar, "Improving Signal-to-Noise Ratio Estimation for Autonomous Receivers," *The Interplanetary Network Progress Report*, vol. 42-159, Jet Propulsion Laboratory, Pasadena, California, pp. 1–19, November 15, 2004. http://ipnpr/progress_report/42-159/159D.pdf

[4] S. M. Kay, *Fundamentals of Statistical Signal Processing, Volume II: Detection Theory*, Upper Saddle River, New Jersey: Prentice Hall PTR, 1998.

[5] W. C. Lindsey and M. K. Simon, *Telecommunication Systems Engineering*, New York: Dover Publications, 1973.

[6] M. Simon and D. Divsalar, "Data Format Classification for Autonomous Radio Receivers," *The Interplanetary Network Progress Report*, vol. 42-159, Jet Propulsion Laboratory, Pasadena, California, pp. 1–27, November 15, 2004. http://ipnpr/progress_report/42-159/159G.pdf

[7] A. Papoulis and S. U. Pillai, *Probability, Random Variables and Stochastic Processes*, fourth ed., New York: McGraw-Hill, 2002.

Chapter 8
Carrier Synchronization

Marvin K. Simon and Jon Hamkins

Traditionally, carrier synchronization (sync) techniques have been developed assuming that the modulation format and signal constellation characteristics are known a priori. By modulation format we mean that the modulation index is chosen so that either the carrier is fully suppressed or a residual carrier component remains. By constellation characteristics we refer to the shape of the constellation, e.g., a circle for M-ary phase-shift keying (M-PSK) or a square for quadrature amplitude modulation (QAM), and its size in terms of the number of signal points it contains. Aside from knowing the modulation index and signal constellation structure, it is also customary to have knowledge of the data rate and type (e.g., non-return to zero (NRZ) versus Manchester code) since the true optimum design of the loop depends on this information.

In autonomous radio operation, the most optimistic situation would be that the receiver contain a carrier synchronization structure that is capable of tracking the carrier phase independently of the above-mentioned considerations. Unfortunately, this is not completely possible since, for example, a squaring loop (or equivalently a binary phase-shift keying (BPSK) Costas loop) cannot track a quadrature phase-shift keying (QPSK) modulation and likewise a 4th power loop (or equivalently a QPSK Costas loop, sometimes referred to as an in-phase–quadrature (I-Q) loop) cannot properly track a BPSK signal.[1] Nevertheless, while in principle each carrier synchronization loop developed for a given modulation format, constellation, and data rate/type has certain unique characteristics, they do share a number of similarities, e.g., a common front-end demodulator

[1] The inability of a QPSK Costas loop to properly track a BPSK signal will be treated later on in the chapter since this is an issue that has not been widely discussed in the literature.

structure, that allows one to consider designs that could be operational in the absence of complete a priori knowledge of all of these characteristics. For example, if the modulation is restricted to the M-PSK class, then it is possible to construct a universal structure that performs the carrier synchronization function for all values of M. This structure is derived by first determining the maximum a posteriori (MAP) estimate of carrier phase based on an observation of the received signal, namely, M-PSK plus additive white Gaussian noise (AWGN), and then using this to motivate a closed-loop carrier synchronization loop. Such a structure, referred to as the MAP estimation loop, has been previously proposed in the literature for cases where the modulation is known beforehand [1]. In fact, it can be shown (see Appendix 8-A for a derivation for BPSK modulation) that, by making an analogy between the closed-loop bandwidth and the noise bandwidth of an integrate-and-dump (I&D) filter of duration equal to the observation time for the open-loop MAP estimate, the closed loop approaches the Cramer–Rao lower bound on the variance of an unbiased estimate of the phase of a modulated carrier.

Still further, if the modulation is known to be other than suppressed carrier, i.e., a modulation index less than $\pi/2$ rad, then it is still possible to exploit the power in both the data and residual carrier components for carrier-tracking purposes provided one has knowledge of the modulation index itself. Such knowledge could be derived noncoherently, i.e., in the absence of carrier synchronization, from a suitable modulation index estimator (to be discussed elsewhere in the monograph). Loops of this type have been referred to in the literature as hybrid carrier tracking loops and like their suppressed-carrier counterparts are motivated by the same MAP considerations.

In what follows, we shall primarily restrict ourselves to the class of M-PSK modulations with known data format (pulse shape) that once again could be determined by a separate data format classifier operating noncoherently (to be discussed in another chapter of the monograph). It is also possible with minor modification, e.g., by replacing the matched filters in the I and Q arms of the loop with simple low-pass filters, to make the carrier synchronizer operation somewhat independent of the exact pulse shape but not without some attendant loss in performance. In deriving a generic carrier synchronization structure for this class of modulations, we shall consider a system with fixed modulation bandwidth which implicitly implies a fixed data *symbol* rate for all values of M. This is consistent with the same assumption made for various other classifiers in other chapters of the monograph.

Although the MAP estimation loops mentioned above are optimum in the sense of yielding the best tracking performance as measured by the variance of the loop phase error, their implementation typically involves nonlinearities that depend on other system parameters, such as signal-to-noise ratio (SNR). To circumvent this dependence, the most convenient form for use in the autonomous

radio application is the simplification based on low SNR approximations applied to the nonlinearities inherent in the MAP phase estimate. When this is done, the error signal in the loop for M-PSK is of the form $\sin M\phi$, where ϕ is the loop phase error, which from simple trigonometry can be written as $\sin M\phi = 2\sin[(M/2)\,\phi]\cos[(M/2)\,\phi]$. Thus, it is seen that the error signal in the loop for M-PSK is formed from the *product* of the error signal $\sin[(M/2)\,\phi]$ and the lock detector signal $\cos[(M/2)\,\phi]$ in the loop for $M/2$-PSK modulation. This simple relationship forms the basis for implementing the universal structure and will be discussed and demonstrated later on in the chapter. For a further treatment of this subject, the reader is referred to [2].

8.1 Suppressed versus Residual Carrier Synchronization

In the past, carrier synchronization loops typically have fallen into two categories: those that track a discrete carrier, e.g., the phase-locked loop (PLL), and those that track a fully suppressed carrier, e.g., the Costas loop. A fully suppressed carrier comes about when a digital modulation is impressed on a carrier with a modulation index equal to $\pi/2$ rad, whereas a discrete (residual) carrier component appears in the spectrum when the modulation index is less than $\pi/2$ rad. For example, consider a binary modulation phase modulated onto a carrier with modulation index β, which in mathematical form is described by

$$s(t) = \sqrt{2P_t}\sin\left(\omega_c t + \beta m(t) + \theta_c\right) \tag{8-1}$$

where P_t is the total available transmitter power, ω_c is the radian carrier frequency, θ_c is the unknown carrier phase to be tracked, and $m(t) = \sum_{n=-\infty}^{\infty} c_n \times p(t - nT)$ is the data modulation with $p(t)$ the pulse shape, $\{c_n\}$ the random binary data taking on values ± 1 with equal probability, and T the data (baud) interval ($\mathcal{R} = 1/T$ is the data rate). Since for NRZ data $p(t)$ is a unit rectangle and for Manchester code $p(t)$ is a unit square wave, then because of the purely digital (± 1) nature of $m(t)$, by applying simple trigonometry to Eq. (8-1) we get

$$\begin{aligned} s(t,\theta_c) &= \sqrt{2P_t}\cos\beta\sin(\omega_c t + \theta_c) + \sqrt{2P_t}\sin\beta m(t)\cos(\omega_c t + \theta_c) \\ &= \sqrt{2P_c}\sin(\omega_c t + \theta_c) + \sqrt{2P_d}m(t)\cos(\omega_c t + \theta_c) \end{aligned} \tag{8-2}$$

where $P_c = P_t\cos^2\beta$ denotes the power in the carrier (unmodulated) component and $P_d = P_t\sin^2\beta$ denotes the power in the data (modulated) component. Since the power spectral density (PSD) of an NRZ-formatted signal is of the form

$(\sin \pi fT/\pi fT)^2$, then direct modulation of the carrier with such a waveform would, for any modulation index $\beta < \pi/2$, result in a discrete carrier occurring at the point of maximum energy (i.e., $f = 0$) in the data modulation spectrum. This in turn makes it difficult to extract carrier synchronization from the discrete component with the loop most commonly used for such purposes, namely, a PLL. The loss due to the overlapping spectrum is $1/(1+2E_s)$, where E_s is the symbol energy [3]. Because of this, the National Aeronautics and Space Administration (NASA) proximity-link standard [4] stipulates that direct modulation of a carrier with BPSK having NRZ formatting is always used in a suppressed-carrier mode, i.e., with $\beta = \pi/2$. On the other hand, since a Manchester-coded signal has a PSD of the form $\left[\sin^2(\pi fT/2)/(\pi fT/2)\right]^2$ that has a null at zero frequency, then it is quite natural to allow for insertion of a discrete carrier there, and thus a modulation index $\beta < \pi/2$ is certainly reasonable from a carrier-tracking standpoint.

Since a Manchester-coded waveform is equivalent to the product of an NRZ waveform and a unit square wave at the data rate, one can view the form of the signal in Eq. (8-1) for this case as direct modulation of a carrier with an NRZ data waveform that has first been modulated onto a square-wave subcarrier at the data rate. With that in mind, one could, as is often done (at the expense of an increase in bandwidth of the signal), achieve a non-suppressed-carrier mode of operation with an NRZ signal by first modulating it onto a square-wave[2] subcarrier (not necessarily at the data rate) prior to direct modulation of the carrier. When this is done, the signal takes the form

$$s(t, \theta_c) = \sqrt{2P_t} \sin\left(\omega_c t + \beta m(t) Sq(\omega_{sc} t) + \theta_c\right) \tag{8-3}$$

where ω_{sc} denotes the radian subcarrier frequency. Most of the discussion of this chapter will deal with the absence of subcarriers and, thus, unless otherwise specified, when considering a residual carrier mode of operation, we shall implicitly assume the presence of Manchester coding, whereas for suppressed-carrier operation we shall allow for either NRZ or Manchester formats.

8.2 Hybrid Carrier Synchronization

Despite the fact that a data-modulated suppressed-carrier signal component also exists in Eq. (8-2), it is often neglected in deriving carrier synchronization. In other words, for the case where the total transmitted power is divided between a discrete (unmodulated) carrier and a data-modulated suppressed carrier, the

[2] Often, a sine-wave subcarrier is used with the same purpose of shifting the PSD of the baseband modulation away from the origin to allow insertion of a discrete carrier.

carrier synchronization function is most often accomplished based on the discrete carrier component alone, i.e., with a PLL. Thus, since the power split between discrete and data-modulated carriers results in a carrier power, P_c, that is less than the total transmitted power, P_t, the loop is operating with an SNR less than that which is potentially available if one were to emply both signal components in the carrier synchronization process.

Since a PLL is a closed-loop synchronization scheme motivated by MAP estimation of the phase of a discrete carrier and a Costas loop is a closed-loop synchronization scheme motivated by MAP estimation of the phase of a fully suppressed carrier, one might anticipate that for a signal of the form in Eq. (8-2) the optimum (in the MAP sense) closed-loop scheme would be a combination (hybrid) of the two loops [5]. Indeed such is the case, as is illustrated by the following mathematical development.

Let the signal of Eq. (8-2) received in AWGN be denoted by

$$r\left(t\right) = s\left(t, \theta_c\right) + n\left(t\right) \tag{8-4}$$

Then the likelihood function (conditioned on the unknown phase and data) for the kth interval is given by

$$p(r_k|\theta_c, c_k) = C \exp\left\{-\frac{1}{N_0}\left[\int_{kT}^{(k+1)T}\left(r(t) - \sqrt{2P_d}c_k p(t-kT)\sin(\omega_c t + \theta_c)\right.\right.\right.$$

$$\left.\left.\left. - \sqrt{2P_c}\cos\left(\omega_c t + \theta_c\right)\right)^2\right] dt\right\} \tag{8-5}$$

where N_0 is the single-sided noise power spectral density in W/Hz and C is a constant of proportionality. Averaging over the data and ignoring terms that are not decision-dependent gives

$$p\left(r_k\,|\theta_c\right) \cong \exp\left\{\frac{2\sqrt{2P_c}}{N_0}\int_{kT}^{(k+1)T} r(t)\cos(\omega_c t + \theta_c)dt\right\}$$

$$\times \cosh\left\{\frac{2\sqrt{2P_d}}{N_0}\int_{kT}^{(k+1)T} r(t)p(t-kT)\sin(\omega_c t + \theta_c)dt\right\} \tag{8-6}$$

The log-likelihood function for a sequence of K bits is then

$$\Lambda(\theta_c) \cong \ln \prod_{k=0}^{K-1} p(r_k|\theta_c) = \sum_{k=0}^{K-1} \frac{2\sqrt{2P_c}}{N_0} \int_{kT}^{(k+1)T} r(t)\cos(\omega_c t + \theta_c)dt$$

$$+ \sum_{k=0}^{K-1} \ln\cosh\left\{ \frac{2\sqrt{2P_d}}{N_0} \int_{kT}^{(k+1)T} r(t)\,p(t-kT)\sin(\omega_c t + \theta_c)dt \right\} \quad (8\text{-}7)$$

Thus, the MAP open-loop estimate of θ_c, namely, $\hat{\theta}_c$, based on the K-bit observation of $r(t)$ is that value of θ_c that maximizes $\Lambda(\theta_c)$.

To obtain a closed-loop synchronizer motivated by the MAP estimation approach, one differentiates the log-likelihood function with respect to θ_c and uses this to form the error signal (to be nulled when $\theta_c = \hat{\theta}_c$) in the closed-loop configuration. Thus, differentiating $\Lambda(\theta_c)$ of Eq. (8-7) with respect to θ_c gives

$$\frac{d\Lambda(\theta_c)}{d\theta_c} \cong -\sum_{k=0}^{K-1} \frac{2\sqrt{2P_c}}{N_0} \int_{kT}^{(k+1)T} r(t)\sin(\omega_c t + \theta_c)dt$$

$$+ \sum_{k=0}^{K-1} \left(\frac{2\sqrt{2P_d}}{N_0} \int_{kT}^{(k+1)T} r(t)\,p(t-kT)\cos(\omega_c t + \theta_c)dt \right)$$

$$\times \tanh\left\{ \frac{2\sqrt{2P_d}}{N_0} \int_{kT}^{(k+1)T} r(t)\,p(t-kT)\sin(\omega_c t + \theta_c)dt \right\} \quad (8\text{-}8)$$

The expression in Eq. (8-8) suggests the hybrid closed loop illustrated in Fig. 8-1. As is typical in actual implementations, the hyperbolic tangent nonlinearity is approximated by either its large argument variant, namely, a signum function (bipolar hard-limiter), or its small argument variant, a linear function. In the former case, we obtain the so-called polarity-type Costas loop, whereas in the latter case we obtain the conventional Costas loop. Also, implicit in Fig. 8-1 is knowledge of the data rate and symbol synchronization, both of which are necessary to implement the matched arm filters, which are of the I&D type. In the next section, we discuss alternative implementations of these arm filters using low-pass filters (LPFs) that are suboptimum but that provide additional robustness to the implementation in terms of the absence of perfect knowledge of the data rate and actual pulse shape and as such do not require symbol synchronization information. In this regard, Fig. 8-2 is the equivalent structure to Fig. 8-1, now using passive arm filters.

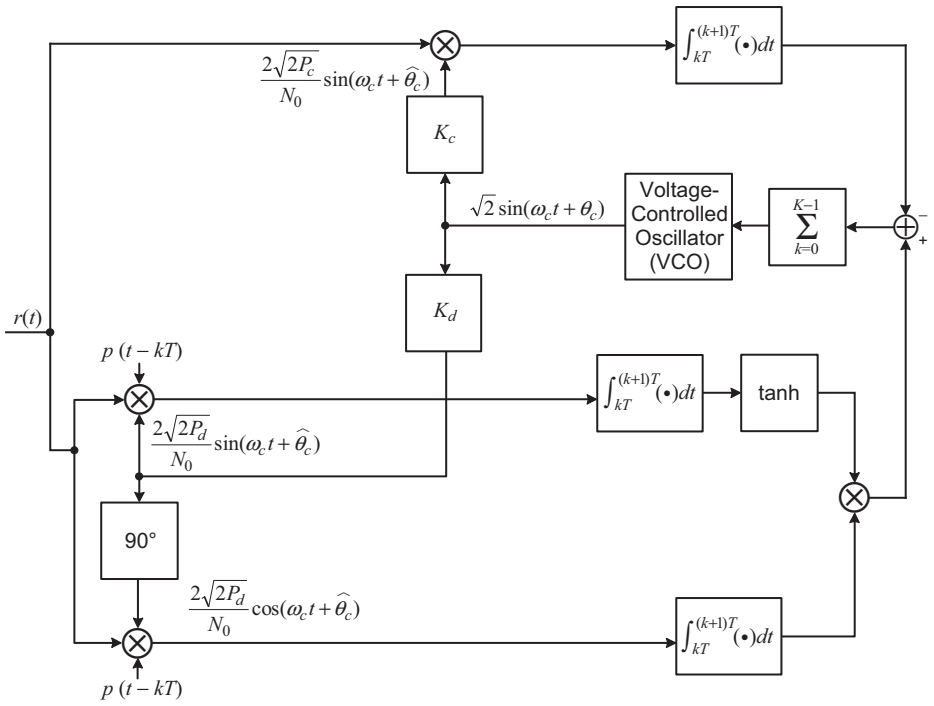


Fig. 8-1. Hybrid closed loop motivated by MAP estimation.

In theory, the gains K_c and K_d would be chosen proportional to $\sqrt{P_c}$ and $\sqrt{P_d}$, which in turn implies knowledge of the modulation index β. In the absence of such perfect knowledge, one would set the gains based on an estimate of β. Even in the absence of such information, one could possibly still use just the Costas-loop component of the hybrid loop alone since, under certain circumstances, it is capable of tracking a residual carrier signal whereas a data-modulated suppressed carrier cannot be tracked by a PLL. Rather than develop the conditions under which this is possible now, we delay this discussion until after we first compare the behavior of Costas loops using active (matched-filter) arm filters to those using passive low-pass filters. At that point, the behavior of the Costas loop with matched arm filters when tracking a residual carrier signal will simply become a special case of that discussion.

8.3 Active versus Passive Arm Filters

The most common measure of performance for a carrier synchronization loop is the variance of the phase error $\phi = \theta_c - \hat{\theta}_c$. For suppressed-carrier tracking loops such as the Costas loop (or the Costas-loop component of the hybrid loop),

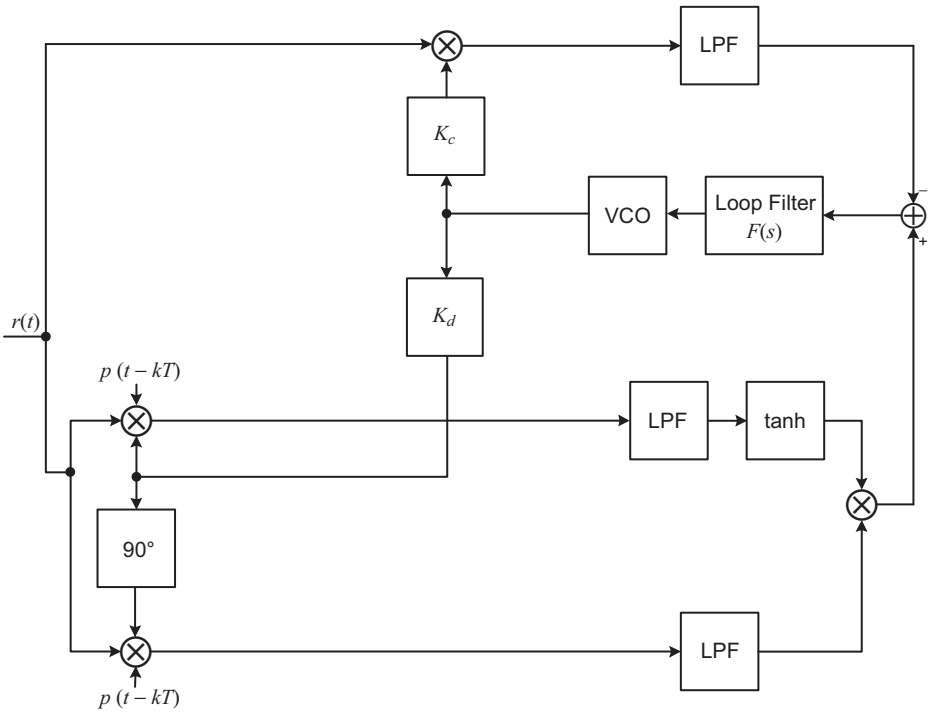


Fig. 8-2. MAP estimation loop for single channel, discrete carrier passive arm filter realization.

in the linear tracking region, the mean-squared phase error can be related to the loop SNR by

$$\sigma_\phi^2 = \frac{1}{\rho S_L} \tag{8-9}$$

where

$$\rho = \frac{P_d}{N_0 B_L} \tag{8-10}$$

with B_L denoting the single-sided loop bandwidth and S_L the so-called "squaring loss," which reflects the additional penalty relative to the PLL loop SNR due to the squaring operation and is caused by the combination of signal × signal ($S \times S$), signal × noise ($S \times N$), and noise × noise ($N \times N$) distortions. The exact nature of the squaring loss depends heavily on the nonlinearity implemented in the in-phase arm (i.e., hyperbolic tangent function or its small and large argument approximations) and the type (active versus passive) of arm filters in both the I and Q arms.

From the standpoint of performance, the optimum behavior is obtained using a hyperbolic tangent nonlinearity and matched (to the pulse shape) arm filters. For this case, the squaring loss is given by [6]

$$S_L = \frac{\left(\overline{\tanh\left[2R_d - \sqrt{2R_d}X\right]}^X\right)^2}{\overline{\tanh^2\left[2R_d - \sqrt{2R_d}X\right]}^X}$$

$$= \exp\left(-R_d\right)\overline{\tanh\left(\sqrt{2R_d}X\right)\sinh\left(\sqrt{2R_d}X\right)}^X \tag{8-11}$$

where X is a (0,1) Gaussian random variable (RV), the overbar indicates statistical averaging with respect to the Gaussian probability distribution of X, and $R_d \triangleq P_dT/N_0$ is the data SNR. For the polarity-type Costas loop ($\tanh x \cong \operatorname{sgn} x$), Eq. (8-11) simplifies to

$$S_L = \operatorname{erf}^2\left(\sqrt{R_d}\right) \tag{8-12}$$

where $\operatorname{erf} x = (2/\sqrt{\pi})\int_0^x \exp\left(-y^2\right)dy$ is the error function, whereas for the conventional Costas loop ($\tanh x \cong x$), we obtain

$$S_L = \frac{2R_d}{1+2R_d} \tag{8-13}$$

As a compromise between Eqs. (8-12) and (8-13), the hyperbolic tangent nonlinearity is often approximated by a saturated amplifier, i.e.,

$$\tanh x \cong \begin{cases} x, & |x| \le 1 \\ \operatorname{sgn} x, & |x| > 1 \end{cases} \tag{8-14}$$

whose squaring loss can also be obtained in closed form as

$$S_L = \frac{\left\{\sqrt{\frac{R_d}{\pi}}\left[\exp\left(-A_1^2\right) - \exp\left(-A_2^2\right)\right] + \left(R_d + \frac{1}{2}\right)\operatorname{erf} A_1 + \left(R_d - \frac{1}{2}\right)\operatorname{erf} A_2\right\}^2}{1 - \sqrt{\frac{R_d}{\pi}}\left\{(1-2R_d)\exp\left(-A_1^2\right) + (1+2R_d)\exp\left(-A_2^2\right)\right\} + \left[R_d(1+2R_d) - \frac{1}{2}\right]\left[\operatorname{erf} A_1 + \operatorname{erf} A_2\right]};$$

$$A_1 \triangleq \frac{1+2R_d}{2\sqrt{R_d}}, \quad A_2 \triangleq \frac{1-2R_d}{2\sqrt{R_d}} \tag{8-15}$$

Figure 8-3 is a plot of the squaring losses in Eqs. (8-11), (8-12), (8-13), and (8-15) versus R_d in dB. We observe that, depending on the value of R_d, the polarity-type and conventional Costas loops trade performance in terms of which is superior, whereas for all values of R_d, the loop implemented with the hyperbolic tangent nonlinearity provides the best performance (minimum squaring loss) with the performance of the saturated amplifier nonlinearity virtually identical to it. Once again we remind the reader that the performances predicted by Eqs. (8-11), (8-12), (8-13), and (8-15) require the implementation of matched arm filters, which in turn require knowledge of the data rate/type and also symbol synchronization.

Before leaving the discussion of Costas loops with active arm filters, it is of interest to compare the performance (phase error variance) of the loop with the Cramer–Rao bound [7] on the variance of an unbiased estimator of the phase of a modulated BPSK carrier. The derivation of such a bound is given in Appendix 8-A, where it is shown that

$$\sigma_\phi^2 \geq \begin{cases} \dfrac{1}{K(2R_d)^2}, & R_d \text{ small} \\ \dfrac{1}{K(2R_d)}, & R_d \text{ large} \end{cases} \tag{8-16}$$

with, consistent with the notation used earlier in the chapter, K the number of bits in the observation. For the I-Q Costas loop at low SNR, we can rewrite Eq. (8-9) combined with Eqs. (8-10) and (8-13) as

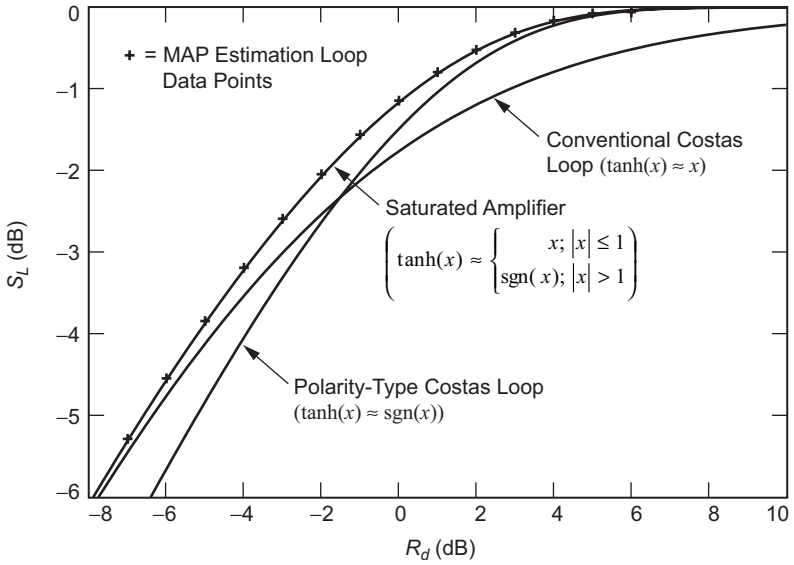


Fig. 8-3. A comparison of the squaring-loss performance of the MAP estimation loop with several practical implementations; BPSK.

$$\sigma_\phi^2 = \frac{N_0 B_L (1 + 2R_d)}{P_d (2R_d)} = \frac{2B_L T (1 + 2R_d)}{(2R_d)^2} \cong \frac{B_L (2KT)}{K (2R_d)^2} \tag{8-17}$$

Similarly, for high SNR we can rewrite Eq. (8-9) combined with Eqs. (8-10) and (8-12) as

$$\sigma_\phi^2 = \frac{N_0 B_L}{P_d \operatorname{erf}^2(\sqrt{R_d})} = \frac{2B_L T}{2R_d \operatorname{erf}^2(\sqrt{R_d})} \cong \frac{B_L (2KT)}{K (2R_d)} \tag{8-18}$$

Comparing Eqs. (8-17) and (8-18) with the Cramer–Rao bounds in Eq. (8-16), we observe that the performance of the I-Q Costas loop approaches these bounds at low and high SNR if we make the equivalence between the loop bandwidth and the noise bandwidth of an I&D of duration equal to the observation time, i.e., set $B_L = 1/2KT$. Thus, in conclusion, while for a fixed bandwidth and data rate the I&D Costas loop asymptotically behaves inverse linearly with SNR at high SNR, it has an asymptotic inverse square-law behavior with SNR at low SNR. In both cases, however, the behavior is inverse linear with the duration of the observation.

As intimated previously, it is possible to implement the arm filters of the Costas loop in passive form, thereby eliminating the need for symbol synchronization prior to obtaining carrier synchronization. Furthermore, as we shall see momentarily, in the absence of exact data rate information, the passive arm filter implementation is robust in that its performance is quite insensitive to a large variation of the data rate in one direction relative to the optimum choice of arm filter bandwidth. It is also possible to design the arm filters with different noise bandwidths ([8] suggests removing the quadrature arm filter completely), which has the advantage of improving the acquisition capability of the loop and also reducing its tendency to false lock but is accompanied by a penalty in tracking performance (as measured by squaring loss). For high-detection SNR, the additional squaring-loss penalty is quite small, and thus this technique could result in a significant overall performance advantage. For the current discussion, we shall assume that the two arm filters have identical designs.

Consider the Costas loop with LPFs having transfer function $G(s)$ (s is the Laplace transform operator) illustrated in Fig. 8-4. When operating in the linear tracking region, the mean-squared error is given by Eq. (8-9), where the squaring loss is now given by [9]

$$S_L = \frac{K_2^2}{K_4 + K_L \dfrac{B_i/\mathcal{R}}{2R_d}} \tag{8-19}$$

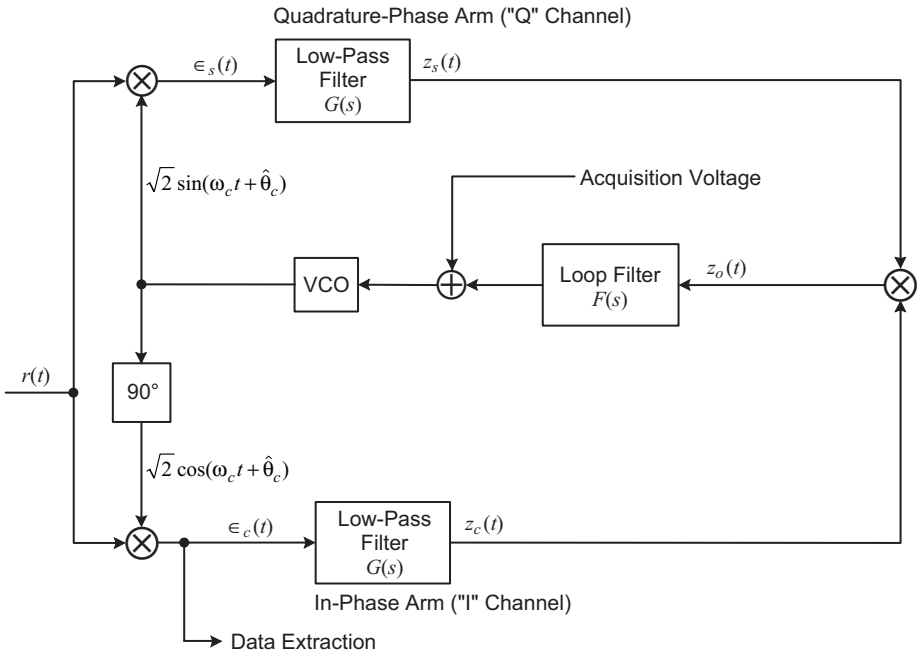


Fig. 8-4. The Costas loop with passive arm filters.

where

$$K_L = \frac{\int_{-\infty}^{\infty} |G(2\pi f)|^4 \, df}{\int_{-\infty}^{\infty} |G(2\pi f)|^2 \, df} \tag{8-20}$$

is a constant that depends only on the arm filter type,

$$K_l = \int_{-\infty}^{\infty} S_m(f) \, |G(2\pi f)|^l \, df, \quad l = 2, 4 \tag{8-21}$$

with

$$S_m(f) = \frac{1}{T} |P(j2\pi f)|^2 \tag{8-22}$$

the power spectral density of the modulation [$P(j2\pi f)$ is the Fourier transform of the pulse shape $p(t)$] and

$$B_i = \int_{-\infty}^{\infty} |G(j2\pi f)|^2 \, df \tag{8-23}$$

the two-sided noise bandwidth of the arm filters.[3] Typical values of the quantities in Eqs. (8-20) through (8-23) for the class of N-pole Butterworth filters and several data formats are given below:

Filter Transfer Function:

$$|G(2\pi f)|^2 = \frac{1}{1+\left(\dfrac{f}{f_{3\text{ dB}}}\right)^{2N}} \qquad B_i = \left(\frac{N}{\pi}\sin\frac{\pi}{2N}\right)^{-1} f_{3\text{ dB}} \tag{8-24}$$

Data Modulation Power Spectral Density:

$$\text{NRZ}: \qquad S_m(f) = T\frac{\sin^2\pi fT}{(\pi fT)^2} \tag{8-25}$$

$$\text{Manchester}: \qquad S_m(f) = T\frac{\sin^4\pi fT/2}{(\pi fT/2)^2} \tag{8-26}$$

$$\text{Square Wave}: \qquad S_m(f) = \frac{1}{4}\left(\frac{4}{\pi}\right)^2 \sum_{k=-\infty}^{\infty} \frac{1}{(2k-1)^2}\delta\left(f-\frac{2k-1}{2T}\right) \tag{8-27}$$

The square wave in Eq. (8-27) has period $2T$.

Evaluation of K_L:

$$K_L = \frac{2N-1}{2N} \tag{8-28}$$

Evaluation of K_2, K_4:

(a) Single-Pole Butterworth Filter ($N = 1$), NRZ Data:

[3] We assume that the arm filter transfer function is normalized such that $G(0) = 1$.

$$K_2 = 1 - \frac{1}{2B_i/\mathcal{R}}\left[1 - \exp\left(-2B_i/\mathcal{R}\right)\right]$$
$$K_4 = 1 - \frac{3 - (3 + 2B_i/\mathcal{R})\exp\left(-2B_i/\mathcal{R}\right)}{4B_i/\mathcal{R}} \tag{8-29}$$

(b) Two-Pole Butterworth Filter ($N = 2$), NRZ Data:

$$K_2 = 1 - \frac{1}{4B_i/\mathcal{R}}\left\{1 - \exp\left(\frac{-2B_i}{\mathcal{R}}\right)\left[\cos\left(\frac{2B_i}{\mathcal{R}}\right) - \sin\left(\frac{2B_i}{\mathcal{R}}\right)\right]\right\}$$

$$K_4 = 1 - \frac{5 - \left\{4\left(\frac{B_i}{\mathcal{R}}\right)\cos\left(\frac{2B_i}{\mathcal{R}}\right) + 5\left[\cos\left(\frac{2B_i}{\mathcal{R}}\right) - \sin\left(\frac{2B_i}{\mathcal{R}}\right)\right]\right\}\exp\left(\frac{-2B_i}{\mathcal{R}}\right)}{16B_i/\mathcal{R}}$$

(c) Single-Pole Butterworth Filter ($N = 1$), Manchester Data:

$$K_2 = 1 - \frac{1}{2B_i/\mathcal{R}}\left[3 - 4\exp\left(-B_i/\mathcal{R}\right) + \exp\left(-2B_i/\mathcal{R}\right)\right]$$
$$K_4 = 1 - \frac{9 - 4\left(3 + B_i/\mathcal{R}\right)\exp\left(-B_i/\mathcal{R}\right) + \left(3 + 2B_i/\mathcal{R}\right)\exp\left(-2B_i/\mathcal{R}\right)}{4B_i/\mathcal{R}} \tag{8-30}$$

(d) Two-Pole Butterworth Filter ($N = 2$), Manchester Data:

$$K_2 = 1 - \frac{1}{4B_i/\mathcal{R}}\left\{3 - 4\exp\left(\frac{-B_i}{\mathcal{R}}\right)\left[\cos\left(\frac{B_i}{\mathcal{R}}\right) - \sin\left(\frac{B_i}{\mathcal{R}}\right)\right] + \exp\left(\frac{-2B_i}{\mathcal{R}}\right)\left[\cos\left(\frac{2B_i}{\mathcal{R}}\right) - \sin\left(\frac{2B_i}{\mathcal{R}}\right)\right]\right\}$$

$$K_4 = 1 - \frac{15 - \left\{8\left(\frac{B_i}{\mathcal{R}}\right)\cos\left(\frac{B_i}{\mathcal{R}}\right) + 20\left[\cos\left(\frac{B_i}{\mathcal{R}}\right) - \sin\left(\frac{B_i}{\mathcal{R}}\right)\right]\right\}\exp\left(\frac{-B_i}{\mathcal{R}}\right)}{16B_i/\mathcal{R}} - \frac{\left\{4\left(\frac{B_i}{\mathcal{R}}\right)\cos\left(\frac{2B_i}{\mathcal{R}}\right) + 5\left[\cos\left(\frac{2B_i}{\mathcal{R}}\right) - \sin\left(\frac{2B_i}{\mathcal{R}}\right)\right]\right\}\exp\left(\frac{-2B_i}{\mathcal{R}}\right)}{16B_i/\mathcal{R}} \tag{8-31}$$

(e) Single-Pole Butterworth Filter ($N = 1$), Square Wave:

$$K_2 = 1 - \frac{1}{B_i/\mathcal{R}} \tanh\left(\frac{B_i}{\mathcal{R}}\right)$$

$$K_4 = 1 + \frac{1}{2} \operatorname{sech}^2\left(\frac{B_i}{\mathcal{R}}\right) - \frac{3}{2B_i/\mathcal{R}} \tanh\left(\frac{B_i}{\mathcal{R}}\right)$$

Finally, using a partial fraction expansion technique, closed-form expressions for the squaring loss, specifically, the parameters K_L, K_2, K_4, were derived [10] for a general class of passive arm filters whose transfer function is characterized by simple, but in general complex, poles.

The numerator of Eq. (8-19) reflects the $S \times S$ distortion whereas the two terms of the denominator reflect the $S \times N$ and $N \times N$ distortions, respectively. As the arm filter bandwidth narrows, the decrease in the $S \times S$ term dominates, whereas when the arm filter widens, the increase in the $S \times N$ term dominates. Thus, for a fixed filter type and data modulation format, K_L, K_2, and K_4 are only a function of the ratio of arm filter bandwidth to data rate $B_i/\mathcal{R}$, and thus a plot of S_L in dB versus $B_i/\mathcal{R}$ will reveal an optimum value in the sense of maximum S_L.[4] As an example, Figs. 8-5 and 8-6 illustrate such plots for one- and two-pole Butterworth arm filters and Manchester-coded data. We observe that, over a large range of detection SNRs, the squaring loss is extremely sensitive to values of $B_i/\mathcal{R}$ less than the optimum value, whereas it is rather insensitive to values of $B_i/\mathcal{R}$ greater than the optimum value. Thus, in the absence of exact information about the data rate, one can design the loop arm filters so that the optimum $B_i/\mathcal{R}$ ratio corresponds to the *maximum* expected data rate whereupon operation at data rates considerably less than the maximum would result in only a small squaring-loss penalty. We remind the reader that, with active arm filters of the I&D type, one does not have this design flexibility since the arm filter bandwidth of such a filter is fixed at $B_i = 1/T = \mathcal{R}$, and thus the squaring loss is also fixed at its value given by Eq. (8-13). Figure 8-7 is a comparison of the squaring-loss behavior for one- and two-pole Butterworth filters at two different detection SNRs. While the two-pole filter has a slightly better optimum squaring-loss performance, it is also more sensitive to data rate variation above the optimum value than is the one-pole filter. Thus, we see that the design of the arm filter is a trade-off between the optimum performance and sensitivity to data rate variation.

The next question is: How much do we sacrifice in performance by using a passive arm filter rather than the active one (matched filter)? Figure 8-8 illustrates a comparison between the optimum squaring-loss performances of the

[4] Note that since $S_L \leq 1$, maximizing S_L is equivalent to minimizing the squaring loss in dB.

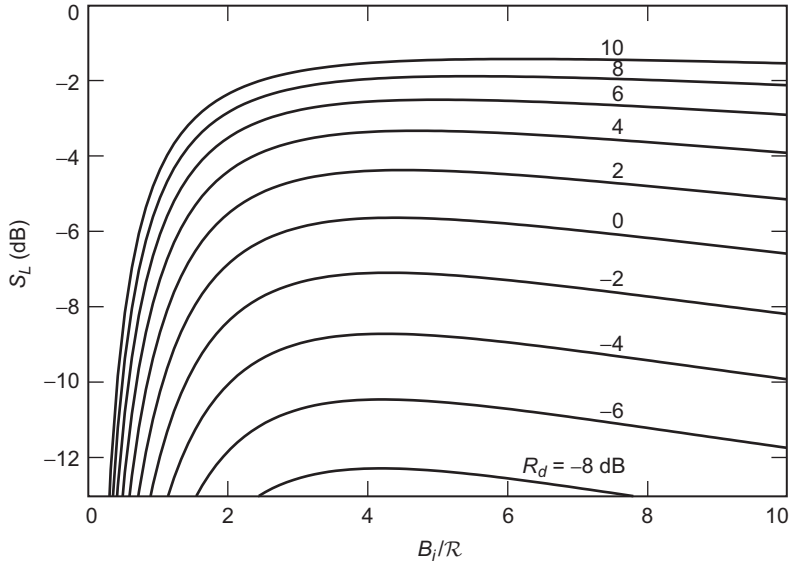


Fig. 8-5. Squaring-loss performance of Costas loop with single-pole Butterworth arm filters; Manchester-coded data.

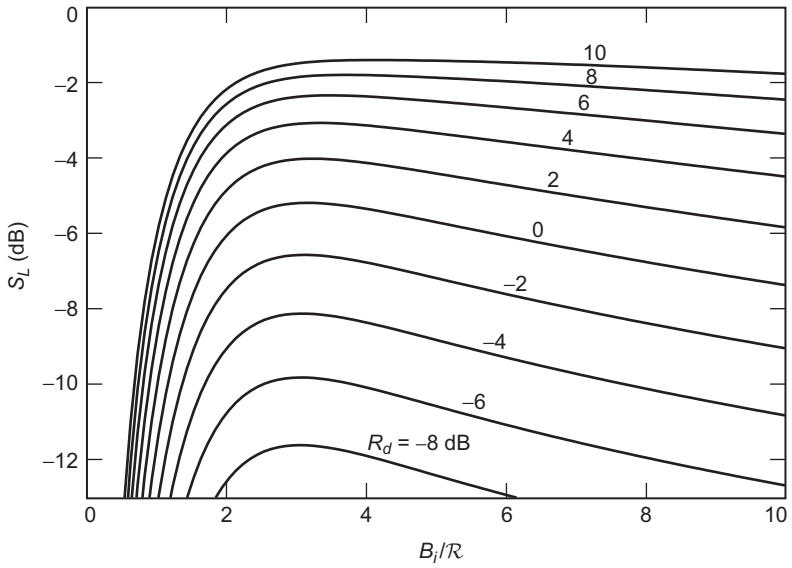


Fig. 8-6. Squaring-loss performance of Costas loop with two-pole Butterworth arm filters; Manchester-coded data.

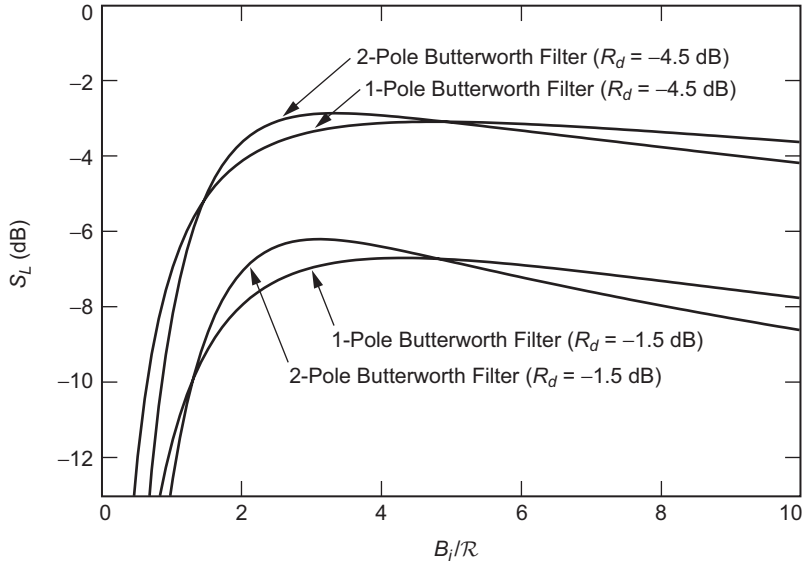


Fig. 8-7. A comparison of the squaring-loss performance of Costas loops with single- and two-pole arm filters at two different SNRs.

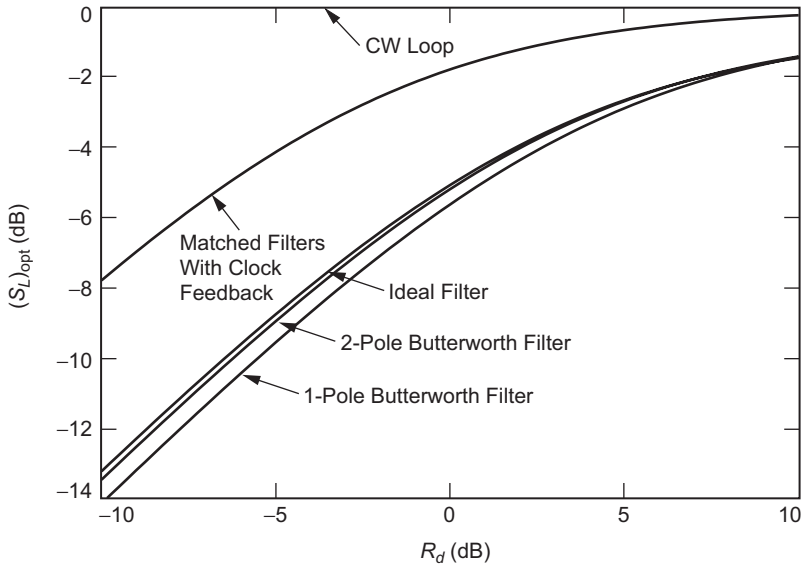


Fig. 8-8. A comparison of the squaring-loss performance of Costas loops with active and passive arm filters.

matched filter with perfect symbol synchronization, one- and two-pole Butterworth filters, and an ideal (brick wall) filter which corresponds to a Butterworth filter with $N \to \infty$. The curves are plotted for a Manchester data format. At high SNR, the optimum squaring-loss performance of the passive filters becomes independent of the number of filter poles, and at a detection SNR of 10 dB it is about 1 dB worse than that of the perfectly symbol-synchronized matched filter. We also note in passing that the performance of the matched-filter implementation is independent of the data format and is given by Eq. (8-13).

We now return to a point made earlier, namely, the ability of a Costas loop to successfully track a residual carrier BPSK signal, including the extreme case of a totally unmodulated carrier. When both a data-modulated and an unmodulated carrier component are simultaneously present at the input to a Costas loop, the two components tend to oppose each other at the error signal point in the loop. In fact, based on an analysis of a similar situation [11], it can be shown that there exists a critical modulation index at which the signal component of the loop error signal (loop S-curve) degenerates to zero, in which case the loop will not track at all. This critical modulation index is given by

$$\beta^* = \cot^{-1}\sqrt{K_2} \tag{8-32}$$

where K_2 is defined for a particular modulation format and arm filter type in Eq. (8-21). For a modulation index *greater* than β^*, the loop S-curve has the usual $\sin 2\phi$ characteristic and has stable lock points at $\phi = \pm n\pi,\ n = 0, 1, 2, \cdots$, which corresponds to the desired operation of the loop (assuming that one takes measures to resolve the normal 180-deg phase ambiguity, such as differential encoding of the input data). On the other hand, for a modulation index *less* than β^* (which of course includes the limiting case of $\beta = 0$, a totally unmodulated carrier), the loop S-curve has a $-\sin 2\phi$ characteristic and has stable lock points at $\phi = \pm(2n+1)\pi/2,\ n = 0, 1, 2, \cdots$ that, from a data detection standpoint, is an undesirable operating condition. Of course, if one knew the modulation index was in the region $\beta < \beta^*$, one could always insert a -1 gain in either the I or Q arm of the loop just prior to the I-Q multiplier, which would thereby invert the loop S-curve and reestablish the lock points at their desired location.

To quantify the degradation in performance in the presence of a residual carrier, it is straightforward to modify the results in [10] to show that under such conditions the squaring loss is given by

$$S_L = \frac{\left(K_2 \sin^2\beta - \cos^2\beta\right)^2}{\cos^2\beta + K_4 \sin^2\beta + \dfrac{1}{2}R_t\left(\sin^2 2\beta\right)\left(\dfrac{S_m(0)}{T}\right) + K_L\dfrac{B_i/R}{2R_t}} \tag{8-33}$$

where $R_t \triangleq P_t T/N_0$ is the total SNR. Since for Manchester coding $S_m(0) = 0$, Eq. (8-33) simplifies to

$$S_L = \frac{\left(K_2 \sin^2 \beta - \cos^2 \beta\right)^2}{\cos^2 \beta + K_4 \sin^2 \beta + K_L \dfrac{B_i/R}{2R_t}} \tag{8-34}$$

Next, for the discrete (unmodulated) carrier case wherein $\beta = 0$ ($P_t = P_c$), Eq. (8-33) becomes

$$S_L = \frac{1}{1 + K_L \dfrac{N_0 B_i}{2P_c}} \tag{8-35}$$

Note that, unlike the data-modulated case, there is no optimum arm filter bandwidth-to-data rate ratio since S_L is independent of the data rate. Furthermore, the squaring loss is smallest (S_L is maximum) for an unmodulated carrier. However, since, for any finite arm filter bandwidth, $S_L < 1$, then from Eq. (8-9), the phase error variance will always exceed that which arises from a PLL as alluded to previously.

When the data modulation is an NRZ-modulated square-wave subcarrier and as such the transmitted signal is characterized by Eq. (8-3), then the behavior of the Costas loop in the presence of a residual carrier is somewhat different from that described above. To illustrate this difference in the simplest way, we assume perfect subcarrier demodulation and once again active (I&D) arm filters. Under these circumstances, the I and Q I&D outputs would be given by

$$\begin{aligned} z_{ck} &= \int_{kT}^{(k+1)T} r(t) \left[\sqrt{2} \cos\left(\omega_c t + \hat{\theta}_c\right) Sq(\omega_{sc} t)\right] dt \\ &= \sqrt{P_c} \sin\phi \int_{kT}^{(k+1)T} Sq(\omega_{sc} t)\, dt + \sqrt{P_d} c_k \cos\phi \int_{kT}^{(k+1)T} Sq^2(\omega_{sc} t)\, dt + N_c \\ &= \sqrt{P_d} T\, c_k \cos\phi + N_c \end{aligned} \tag{8-36}$$

$$\begin{aligned} z_{sk} &= \int_{kT}^{(k+1)T} r(t) \left[-\sqrt{2} \sin\left(\omega_c t + \hat{\theta}_c\right) Sq(\omega_{sc} t)\right] dt \\ &= -\sqrt{P_c} \cos\phi \int_{kT}^{(k+1)T} Sq(\omega_{sc} t)\, dt + \sqrt{P_d} c_k \sin\phi \int_{kT}^{(k+1)T} Sq^2(\omega_{sc} t)\, dt + N_s \\ &= \sqrt{P_d} T\, c_k \sin\phi + N_s \end{aligned}$$

where N_c, N_s are again independent zero-mean Gaussian RVs with variance $N_0T/2$ and we have assumed a unit square-wave subcarrier with either an integer number of subcarrier cycles per bit or instead a large ratio of subcarrier frequency to bit rate. We observe that aside from the reduction of the power in the data signal from the total power P_t to $P_d = P_t \sin^2 \beta$, the I&D outputs in Eq. (8-36) are *identical* to what would be obtained for the same Costas loop operating in the conventional suppressed-carrier mode. Thus, we conclude (at least under the ideal assumptions made) that, in the case of an NRZ-modulated square-wave subcarrier, *the presence of a discrete carrier component does not degrade the performance of the loop other than to reduce the effective power in the data component by* $\sin^2 \beta$. Before leaving this subject, we also mention that since, as previously mentioned in Section 8-1, a Manchester-coded BPSK can be viewed as an NRZ data waveform modulated onto a square-wave subcarrier at the data rate prior to modulation onto the carrier, then, since the above arguments are independent of the subcarrier frequency, the same conclusion would also be true for this case when active (matched) filters are used in the I and Q arms of the Costas loop.

Finally, it is natural to ask whether there is an optimum passive arm filter type in the sense of minimizing the mean-squared phase error. In particular, one seeks a solution for $|G(2\pi f)|^2$ that minimizes

$$S_L^{-1} = \frac{P_d \int_{-\infty}^{\infty} S_m(f) |G(2\pi f)|^4 \, df + \dfrac{N_0}{2} \int_{-\infty}^{\infty} |G(2\pi f)|^2 \, df}{\left\{ \int_{-\infty}^{\infty} S_m(f) |G(2\pi f)|^2 \, df \right\}^2} \tag{8-37}$$

Using the method of Lagrange multipliers, it can be shown that the optimum arm filter magnitude-squared transfer function (which may or may not be physically realizable) is given by

$$|G_{opt}(2\pi f)|^2 = \lambda \frac{S_m(f)}{S_m(f) + \dfrac{N_0}{P_d}} \tag{8-38}$$

where λ is a Lagrange multiplier. Note that for low SNR ($P_d/N_0 \ll 1$), we obtain

$$G_{opt}(2\pi f) = \sqrt{\lambda \frac{P_d}{N_0}} \left[S_m(f) \right]^+ \tag{8-39}$$

where the "+" superscript refers to the part of $S_m(f)$ with poles in the left half-plane and as such represents the matched-filter solution.

8.4 Carrier Synchronization of Arbitrary Modulations

8.4.1 MPSK

In this section, we begin by continuing to consider the class of M-PSK modulations, where the value of $M = 2^m$ is now arbitrary. In an earlier section, we considered a carrier synchronization closed loop for BPSK that was motivated by the MAP estimation technique. Such an "optimum" loop was in the form of an I-Q structure with a hyperbolic tangent nonlinearity in its in-phase arm that resulted from using the gradient of the likelihood function as an error-control signal. Applying the MAP estimation technique for values of $M > 2$ [1,2], it can be shown that the derivative of the log-likelihood ratio has the form

$$\frac{d\Lambda(\theta_c)}{d\theta_c} \cong \sum_{k=0}^{K-1} \sum_{l=0}^{m-2} \frac{C_l V \tanh(C_l U) - S_l U \tanh(S_l V)}{1 + \sum\limits_{\substack{n=0 \\ n \neq l}}^{m-2} \dfrac{\cosh(C_n U)\cosh(S_n V)}{\cosh(C_l U)\cosh(S_l V)}} \tag{8-40}$$

where

$$\begin{aligned} C_l &\triangleq \cos\frac{(2l+1)\pi}{M} \\ S_l &\triangleq \sin\frac{(2l+1)\pi}{M} \end{aligned} \tag{8-41}$$

and

$$\begin{aligned} U &= \frac{2\sqrt{2P_d}}{N_0} \int_{kT}^{(k+1)T} r(t)\, p(t-kT) \sin(\omega_c t + \theta_c) dt \\ V &= \frac{2\sqrt{2P_d}}{N_0} \int_{kT}^{(k+1)T} r(t)\, p(t-kT) \cos(\omega_c t + \theta_c) dt \end{aligned} \tag{8-42}$$

Once again using the derivative of the log-likelihood function to motivate an error signal in a closed-loop implementation, the carrier synchronization loop that results is again an I-Q structure; however, the nonlinearity no longer resides only in the in-phase arm and, furthermore, as can be seen from Eq. (8-40), is consider-

ably more complicated than a simple hyperbolic tangent function.[5] An example of such a closed loop is illustrated in Fig. 8-9 for 8-PSK. We also have seen previously that, by approximating the nonlinearity for small and large arguments, one arrives at structures that are synonymous with well-known synchronization schemes for BPSK and approach the performance of the true MAP-motivated loop at low and high SNRs. Indeed, for $M > 2$, one can take the same approach and arrive at universal structures that lend themselves to simple implementation.

For large arguments we again use the approximation $\tanh x \cong \operatorname{sgn} x$ and in addition

$$\cosh x \cong \frac{1}{2}\exp(|x|) \tag{8-43}$$

Although not immediately obvious, these approximations lead to a closed-loop synchronizer that incorporates the optimum M-PSK symbol detector (MAP phase estimator) in its structure. An example of this is illustrated in Fig. 8-10 for 8-PSK and thus corresponds to the high SNR approximation of Fig. 8-9. For QPSK, the structure is somewhat simpler, involving hard-limiters (signum functions) in each of the I and Q arms as well as a crossover processing between the inputs and outputs of these nonlinearities to form the error signal [5] (see Fig. 8-11 for the passive arm filter implementation).

For small arguments, if one tries to use only the first term in the power series expansion of $\tanh x$, it can be shown [5,6] that for $M > 2$ the derivative of the log-likelihood function in Eq. (8-40) becomes equal to zero for all values of θ_c and thus cannot be used to motivate an error signal in a closed-loop configuration. In order to get a nonzero log-likelihood function, it can be shown that one must retain the first $M/2$ terms in the power series expansion. Thus, for example, for $M = 4$ and $M = 8$, we should use the approximations

$$\tanh x \cong \begin{cases} x - \dfrac{x^3}{3}, & M = 4 \\ x - \dfrac{x^3}{3} + \dfrac{2x^5}{15} - \dfrac{17x^7}{315}, & M = 8 \end{cases} \tag{8-44}$$

and in addition

[5] For $M = 4$, i.e., QPSK, the nonlinearity is still a hyperbolic tangent function that now resides in both the I and Q arms.

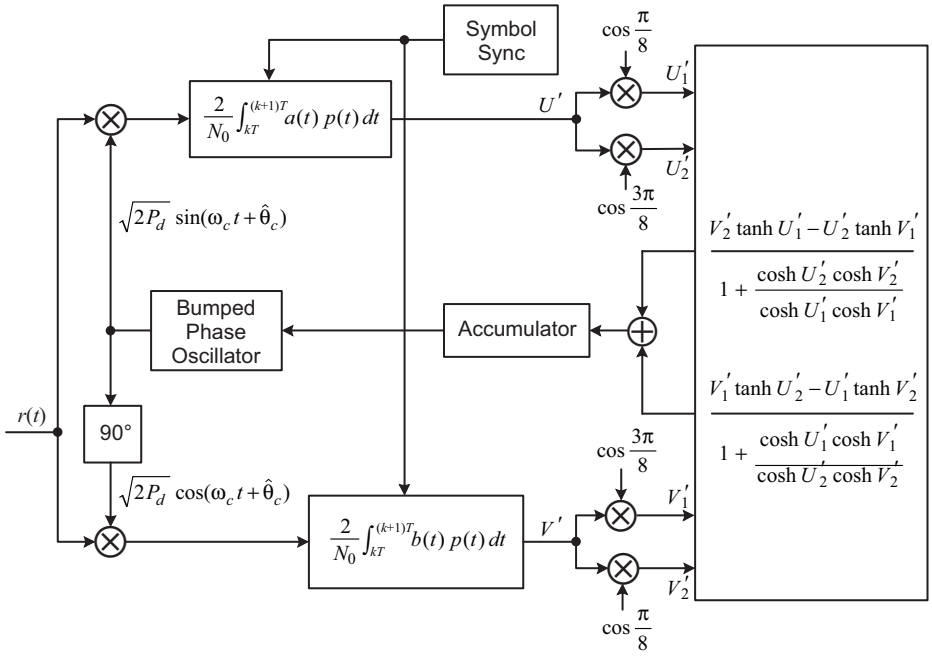


Fig. 8-9. A closed loop motivated by the MAP estimation of carrier phase for 8-PSK.

$$\cosh x \cong \begin{cases} 1+\dfrac{x^2}{2}, & M=4 \\ 1+\dfrac{x^2}{2!}+\dfrac{x^4}{4!}+\dfrac{x^6}{6!}, & M=8 \end{cases} \tag{8-45}$$

Applying these approximations and once again using passive arm filters, we obtain the single closed-loop structure illustrated in Fig. 8-12 capable of carrier synchronizing BPSK, QPSK, and 8-PSK. Several things are interesting about this structure. First of all, it is strictly of the I-Q type in that the loop error signal for all three modulations is derived from the I and Q arm filter outputs (i.e., U and V). The second and more interesting feature is that the error signal for the two higher-order modulations ($M = 4$ and $M = 8$) is derived from the multiplication of a *product* of two signals and a *difference of squares* of these same two signals. To see why this comes about, all one has to do is consider the following simple trigonometry.

For BPSK, the error signal is proportional to $\sin 2\phi$, which can be expressed as

$$\sin 2\phi = 2\sin\phi\cos\phi \tag{8-46}$$

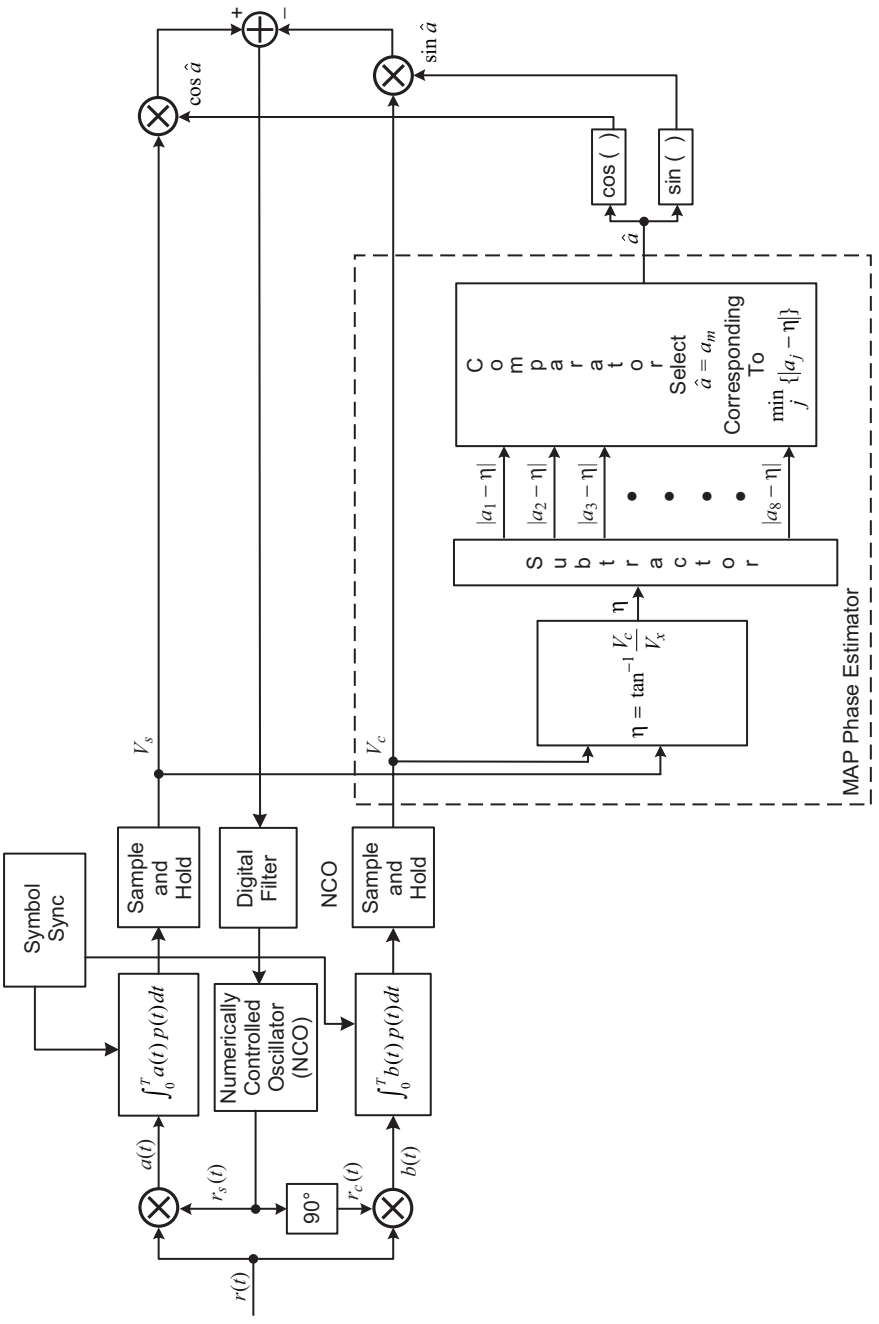


Fig. 8-10. High SNR approximation of the MAP estimation loop for 8-PSK.

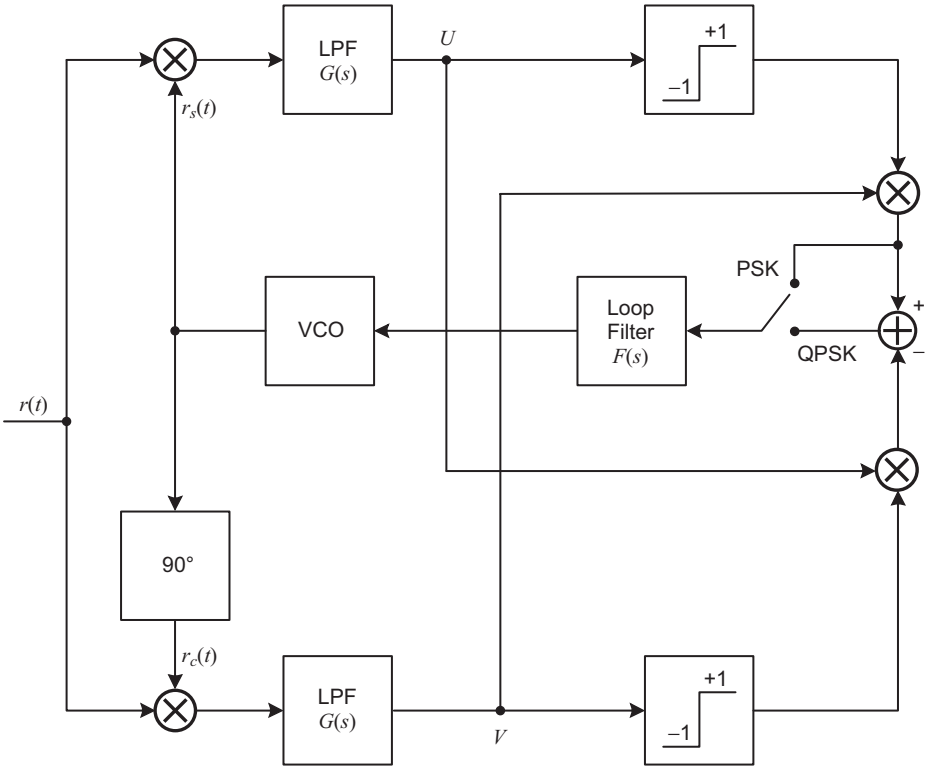


Fig. 8-11. High SNR approximation of the MAP estimation loop for BPSK and QPSK.

Thus, since U is proportional to $\sin\phi$ and V is proportional to $\cos\phi$, we see that the error signal is simply proportional to the product of U and V. For QPSK, the error signal is proportional to $\sin 4\phi$, which can be expressed as

$$\sin 4\phi = 2\sin 2\phi\cos 2\phi = 2 \underbrace{\sin 2\phi}_{\substack{\text{error signal} \\ \text{for BPSK}}} \underbrace{(\cos^2\phi - \sin^2\phi)}_{\substack{\text{lock detector signal} \\ \text{for BPSK}}} \tag{8-47}$$

Similarly, for 8-PSK, the error signal is proportional to $\sin 4\phi$ which can be expressed as

$$\sin 8\phi = 2\sin 4\phi\cos 4\phi = 2 \underbrace{\sin 4\phi}_{\substack{\text{error signal} \\ \text{for QPSK}}} \underbrace{(\cos^2 2\phi - \sin^2 2\phi)}_{\substack{\text{lock detector signal} \\ \text{for QPSK}}} \tag{8-48}$$

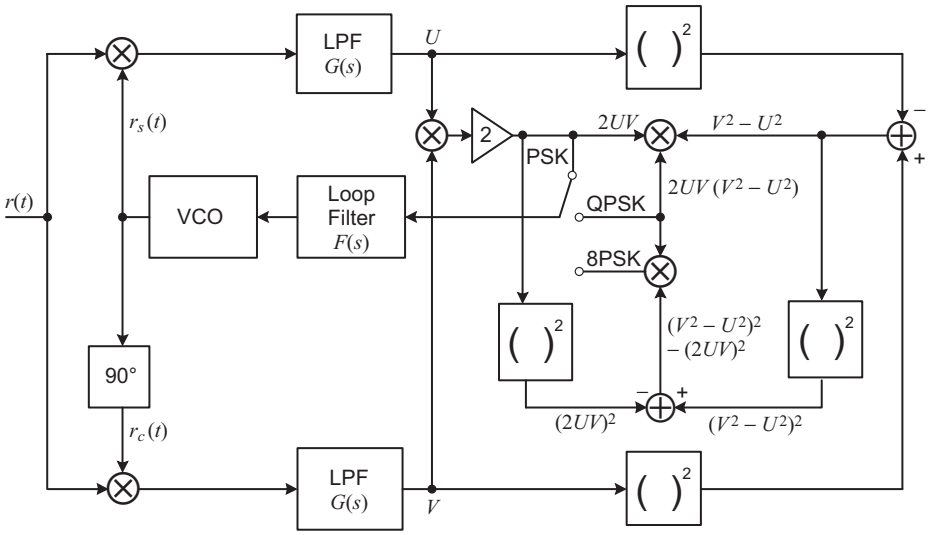


Fig. 8-12. A Costas-type loop capable of carrier tracking BPSK, QPSK, and 8-PSK; low SNR approximation of MAP estimation loop.

Examination of Fig. 8-12 clearly reveals that the error signals for the three modulations are formed in accordance with the relations in Eqs. (8-46) through (8-48). Thus, we conclude in general that the canonical structure for M-PSK has a front end (generation of the signals U and V) as in Fig. 8-12 and forms its error signal from the product of the error signal for $M/2$-PSK and the lock detector signal for $M/2$-PSK.

To use such a canonical configuration in an environment where the value of M is not known for certainty, one would proceed as follows. Since a BPSK loop cannot track QPSK, a QPSK cannot track 8-PSK, etc., one could start with the switch corresponding to the position of the lowest order modulation (BPSK) and sequentially move the switch to the positions of QPSK, 8-PSK, etc., until the loop locks. Another possibility would be to fix the switch in the position corresponding to the highest order loop and, provided that it would be capable of tracking all lower order modulations, accept the additional performance penalty incurred by using a higher order nonlinearity than necessary. To this end, as an example, we now examine the ability of a QPSK loop to track a BPSK signal. Since this issue appears not to be readily discussed in the literature, we shall be a bit more detailed here than we have been thus far in other parts of this chapter.

Consider the MAP estimation loop for QPSK illustrated in Fig. 8-13. The input to the loop is the BPSK signal $r(t) = s(t, \theta_c) + n(t)$, where

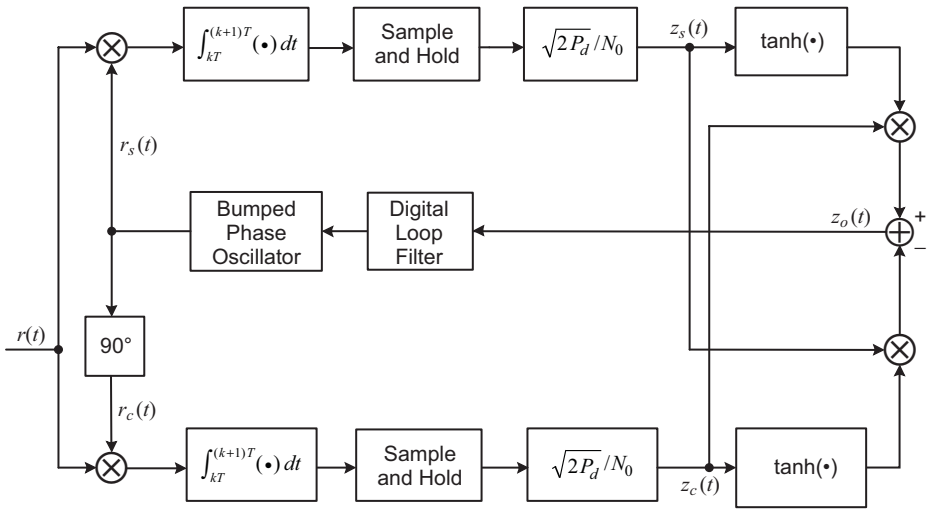


Fig. 8-13. The MAP estimation loop for carrier tracking QPSK with NRZ coding.

$$s\left(t,\theta_c\right)=\sqrt{2P_d}m\left(t\right)\sin\left(\omega_c t+\theta_c\right) \tag{8-49}$$

and the additive noise has the narrowband expansion

$$n\left(t\right)=\sqrt{2}\left[n_c\left(t\right)\cos\left(\omega_c t+\theta_c\right)-n_s\left(t\right)\sin\left(\omega_c t+\theta_c\right)\right] \tag{8-50}$$

where $n_c\left(t\right),n_s\left(t\right)$ are independent, low-pass Gaussian processes, each with single-sided PSD N_0 W/Hz and bandwidth $B_H<\omega_c/2\pi$. Assuming unit input I and Q phase detector (multiplier) gains and demodulation reference signals $r_c\left(t\right)=\sqrt{2}\cos\left(\omega_c t+\hat{\theta}_c\right)$ and $r_s\left(t\right)=\sqrt{2}\sin\left(\omega_c t+\hat{\theta}_c\right)$, then, after passing through the I and Q I&D filters of duration[6] T and amplification by $\sqrt{2P_d}/N_0$, the sample-and-hold outputs $z_c\left(t\right)$ and $z_s\left(t\right)$ are given by

[6] We remind the reader again that we are considering the case where the modulation bandwidth is held fixed and thus the I&D filters in all the configurations have a duration equal to the *symbol* time. Thus, while for the same *information* (bit) rate one would associate two BPSK bits with a QPSK symbol, for the same symbol rate, the I&D filters would correspond to a single bit interval for BPSK.

$$
\begin{aligned}
z_s(t) &\triangleq \frac{\sqrt{2P_d}}{N_0}\int_{kT}^{(k+1)T} r(t)r_s(t)dt \\
&= \sqrt{2}R_d c_k \cos\phi - \sqrt{R_d}X_1\cos\phi - \sqrt{R_d}X_2\sin\phi \\
z_c(t) &\triangleq \frac{\sqrt{2P_d}}{N_0}\int_{kT}^{(k+1)T} r(t)r_c(t)dt \\
&= \sqrt{2}R_d c_k \sin\phi - \sqrt{R_d}X_1\sin\phi + \sqrt{R_d}X_2\cos\phi
\end{aligned}
\tag{8-51}
$$

where $(k+1)\,T \le t \le (k+2)\,T$ and where X_1, X_2 are zero-mean, unit variance independent Gaussian RVs. Multiplying $z_c\,(t)$ by the nonlinearly processed $z_s\,(t)$ and vice versa gives the dynamic error signal

$$
z_o(t) = -z_s(t)\tanh z_c(t) + z_c(t)\tanh z_s(t) \tag{8-52}
$$

As in all analyses of this type, the tracking performance of a loop can, in its linear region of operation (small phase error), be determined by examining the equivalent signal and noise components of the $z_o(t)$ process, more specifically, the slope of the equivalent S-curve at $\phi = 0$ and the variance of the equivalent additive noise. This makes the usual assumption that the loop bandwidth is much less than the data bandwidth.

Since X_1 and X_2 are zero-mean random variables, then, from Eq. (8-52) together with Eq. (8-51), the signal component of $z_o(t)$ has a mean, i.e., the S-curve of the loop, given by

$$
\begin{aligned}
\eta\,(\phi) = & -4\sqrt{2}R_d\overline{(c_k\cos\phi)\tanh\left[\sqrt{2}R_d c_k\sin\phi\right.} \\
& \left.\overline{-\sqrt{R_d}X_1\sin\phi + \sqrt{R_d}X_2\cos\phi\right]}^{X_1,X_2,c_k} \\
& + 4\sqrt{2}R_d\overline{(c_k\sin\phi)\tanh\left[\sqrt{2}R_d c_k\cos\phi\right.} \\
& \left.\overline{-\sqrt{R_d}X_1\cos\phi - \sqrt{R_d}X_2\sin\phi\right]}^{X_1,X_2,c_k} \\
= & -4\sqrt{2}R_d\cos\phi\,\overline{\tanh\left(\sqrt{2}R_d\sin\phi - \sqrt{R_d}X_1\sin\phi + \sqrt{R_d}X_2\cos\phi\right)}^{X_1,X_2} \\
& + 4\sqrt{2}R_d\sin\phi\,\overline{\tanh\left(\sqrt{2}R_d\cos\phi - \sqrt{R_d}X_1\cos\phi - \sqrt{R_d}X_2\sin\phi\right)}^{X_1,X_2}
\end{aligned}
\tag{8-53}
$$

This S-curve is an odd function of ϕ and is periodic with period $\pi/2$, which implies a phase ambiguity for the loop of 90 deg. This ambiguity, which must be resolved for successful data detection, is the same as would be the case for the loop tracking a QPSK signal. Aside from the ambiguity itself, it can also be observed from Eq. (8-53) that the S-curve passes through zero midway between these potential lock points, namely, at $\phi = \pi/4$. Thus, in order to determine whether the loop will correctly lock at $\phi = 0$ (assuming resolution of the ambiguity) or incorrectly at $\phi = \pi/4$, one needs to show that the slope of the S-curve at the former phase error value is positive (which would imply that the slope at the latter value is negative).

The slope of the S-curve at $\phi = 0$ is obtained by differentiating Eq. (8-53) with respect to 4ϕ and evaluating the result at this same phase error value. Recognizing that $\overline{X \operatorname{sech}^2 \sqrt{R_d} X}^X = 0$, it is straightforward to show that

$$K_\eta = \frac{d\eta(\phi)}{d(4\phi)}\Big|_{\phi=0} = \sqrt{2} R_d \overline{\tanh\left(\sqrt{2} R_d - \sqrt{R_d} X\right)}^X - 2R_d^2 \overline{\operatorname{sech}^2\left(\sqrt{R_d} X\right)}^X \tag{8-54}$$

If we now make the low SNR approximation of the nonlinearities in Eq. (8-54) using only the first terms of their Taylor series expansions, i.e.,

$$\tanh x \cong x, \quad \operatorname{sech}^2 \cong 1 \tag{8-55}$$

then since the X's are zero mean, we immediately get

$$K_\eta = 0 \tag{8-56}$$

which implies that the loop would be unable to lock at all. Thus, as was the case in deriving the QPSK MAP estimation loop from maximum-likelihood considerations, we must include the next terms in the Taylor series expansions of the nonlinearities. That is, we apply

$$\tanh x \cong x - \frac{x^3}{3}, \quad \operatorname{sech}^2 x = \frac{d}{dx} \tanh x \cong 1 - x^2 \tag{8-57}$$

which results in the QPSK portion of the implementation in Fig. 8-12 but with I&D arm filters. When this is done, making use of the moments of a Gaussian RV, we obtain

$$\overline{\tanh\left(a-bX\right)}^{X} \cong a\left(1-\frac{a^2}{3}-b^2\right)$$
$$\overline{\operatorname{sech}^2 bX}^{X} \cong 1-b^2 \tag{8-58}$$

and thus

$$\overline{\tanh\left(a-bX\right)}^{X} - a\,\overline{\operatorname{sech}^2 bX}^{X} \cong -\frac{a^3}{3} \tag{8-59}$$

Using Eq. (8-59) in Eq. (8-54) gives

$$K_\eta = \sqrt{2}R_d\left(-\frac{1}{3}\left(\sqrt{2}R_d\right)^3\right) = -\frac{4}{3}R_d^4 \tag{8-60}$$

which is negative and thereby would require that the signs on the summer at the input to the accumulator in Fig. 8-13 be reversed, i.e., the polarity of the error signal flipped, in order for the loop to correctly lock at $\phi = 0$ (and equally well at the ambiguity phase error values $\phi = \pi/2, \pi, 3\pi/2$).

Proceeding now to an evaluation of the equivalent noise PSD, ignoring the self-noise of the signal component, the noise component of $z_o(t)$ (evaluated at $\phi = 0$) is

$$N_e(t) = \sqrt{R_d}X_1 \tanh\left(\sqrt{R_d}X_2\right) + \sqrt{R_d}X_2 \tanh\left[\sqrt{2}R_d c_k - \sqrt{R_d}X_1\right] \tag{8-61}$$

which has zero mean and variance

$$\sigma_{N_e}^2 = R_d\left\{\overline{\tanh^2\left(\sqrt{R_d}X\right)}^{X} + \overline{\tanh^2\left(\sqrt{2}R_d - \sqrt{R_d}X\right)}^{X}\right.$$
$$\left. + 2\overline{X\tanh\left(\sqrt{R_d}X\right)}^{X}\left[\overline{X\tanh\left(\sqrt{2}R_d - \sqrt{R_d}X\right)}^{X}\right]\right\} \tag{8-62}$$

Once again we apply the approximations of the nonlinearities in Eq. (8-57) to evaluate the variance in Eq. (8-62). In particular, the following results for each statistical average are obtained:

$$\begin{aligned}
\overline{\tanh^2\left(\sqrt{2}R_d - \sqrt{R_d}X\right)}^X &= \frac{8}{9}R_d^6 + \frac{20}{3}R_d^5 + \frac{22}{3}R_d^4 - \frac{19}{3}R_d^3 + R_d \\
\overline{\tanh^2\left(\sqrt{R_d}X\right)}^X &= R_d - 2R_d^2 + \frac{5}{3}R_d^3 \\
\overline{X\tanh\left(\sqrt{R_d}X\right)}^X &= \sqrt{R_d}\,(1 - R_d) \\
\overline{X\tanh\left(\sqrt{2}R_d - \sqrt{R_d}X\right)}^X &= -\sqrt{R_d}\left(1 - R_d - 2R_d^2\right)
\end{aligned} \tag{8-63}$$

Substituting the results of Eq. (8-63) into Eq. (8-62) results after some simplification in

$$\sigma_{N_e}^2 = R_d\left(\frac{8}{9}R_d^6 + \frac{20}{3}R_d^5 + \frac{10}{3}R_d^4 - \frac{8}{3}R_d^3 + 2R_d^2\right) \tag{8-64}$$

Because of the I&D arm filters in Fig. 8-13, the noise process of Eq. (8-61) is piecewise constant over intervals of T-seconds duration. Thus, as long as the loop bandwidth is much less than the data bandwidth, this process can be approximated, as has been done in the past, by a delta-correlated process with correlation function given by

$$R_{N_e}(\tau) \triangleq \overline{N_e(t)N_e(t+\tau)} = \begin{cases} \sigma_{N_e}^2\left[1 - \frac{|\tau|}{T}\right], & |\tau| \le T \\ 0; & |\tau| > T \end{cases} \tag{8-65}$$

with equivalent single-sided noise spectral density

$$N_0' \triangleq 2\int_{-\infty}^{\infty} R_{N_e}(\tau)\,d\tau = 2\sigma_{N_e}^2 T \tag{8-66}$$

As such, the linearized phase error variance is given by

$$\sigma_\phi^2 = N_0' B_L / K_\eta^2 = (\rho S_L)^{-1} \tag{8-67}$$

where ρ is the linear loop (PLL) SNR as defined in Eq. (8-10) and S_L is the "quadrupling loss" which reflects the penalty paid due to the signal and noise

cross-products present in $z_o(t)$. Substituting Eq. (8-66) in Eq. (8-67), the quadrupling loss can be identified as

$$S_L = \left(\frac{1}{2R_d}\right)\frac{K_\eta^2}{\sigma_{N_e}^2} = \frac{K_\eta^2/\left(2R_d^2\right)}{\sigma_{N_e}^2/R_d} \tag{8-68}$$

Finally, substituting Eqs. (8-60) and (8-64) in Eq. (8-68) gives the desired result:

$$S_L = \frac{\frac{8}{9}R_d^6}{\frac{8}{9}R_d^6 + \frac{20}{3}R_d^5 + \frac{10}{3}R_d^4 - \frac{8}{3}R_d^3 + 2R_d^2} = \frac{1}{1 + \frac{15}{2R_d} + \frac{15}{4R_d^2} - \frac{3}{R_d^3} + \frac{9}{4R_d^4}} \tag{8-69}$$

It is interesting to compare this loss to that which would result from the same loop tracking a QPSK signal. In particular, for the MAP estimation loop with a QPSK input, the squaring loss is given by [4]

$$S_L = \frac{\left[\overline{\tanh\left(R_d - \sqrt{R_d}X\right)}^X - R_d\,\overline{\mathrm{sech}^2\left(\sqrt{R_d}X\right)}^X\right]^2}{(1+R_d)\,\overline{\tanh^2\left(R_d - \sqrt{R_d}X\right)}^X - \left[\overline{X\tanh\left(R_d - \sqrt{R_d}X\right)}^X - \sqrt{R_d}\tanh\left(R_d - \sqrt{R_d}X\right)\right]} \tag{8-70}$$

which for the low SNR approximation loop (the QPSK portion of Fig. 8-12 with I&D arm filters) reduces, after considerable manipulation, to [6]

$$S_L = \frac{1}{1 + \frac{9}{2R_d} + \frac{6}{R_d^2} + \frac{3}{2R_d^3}} \tag{8-71}$$

Thus, from a comparison of Eqs. (8-69) and (8-71), we conclude that while the QPSK carrier tracking loop is capable of tracking a BPSK signal it does so with a different mean-squared tracking error performance than for a QPSK input signal. Furthermore, the quadrupling loss of Eq. (8-69) exceeds the squaring loss of Eq. (8-13) for all SNRs. The more important issue, however, is the means by which the 90-deg phase ambiguity must be resolved. Whereas for a BPSK loop tracking a BPSK signal one can easily resolve the associated 180-deg phase

ambiguity by differentially encoding the binary data, resolving the 90-deg phase ambiguity associated with the QPSK loop cannot be resolved solely by the same means. In the case of the latter, one must in addition detect the data from the outputs of *both* the I and Q channels and choose the one that has the higher reliability.

In view of the issues brought forth in the above example, it appears that the preferred solution for autonomous operation is not to use a single loop for all modulation orders but rather to first classify the modulation, i.e., determine its order and then place the switch in Fig. 8-12 in its appropriate position.

8.4.2 QAM and Unbalanced QPSK

It is straightforward to extend the notions described above to other modulations with a quadrature structure such as QAM and unbalanced quadrature phase-shift keying (UQPSK) [6]. For example, for square QAM with $M = K^2$ symbols described by

$$\begin{aligned} s(t,\theta_c) &= \sqrt{2}Am_I(t)\cos(\omega_c t+\theta_c) + \sqrt{2}Am_Q(t)\sin(\omega_c t+\theta_c) \\ A &= \sqrt{\frac{3}{2(M-1)}P_d} \end{aligned} \tag{8-72}$$

where $m_I(t), m_Q(t)$ are the quadrature data modulations of rate $1/T$ taking on values $\pm 1, \pm 3, \cdots, \pm\sqrt{M}-1$, the derivative of the log-likelihood function becomes

$$\begin{aligned} \frac{d\Lambda(\theta_c)}{d\theta_c} \cong & \sum_{k=0}^{K-1} \frac{\sum_{l=1}^{\sqrt{M}/2} \exp\left(-c_l^2 R_d\right) c_l V' \sinh(c_l U')}{\sum_{l=1}^{\sqrt{M}/2} \exp\left(-c_l^2 R_d\right) \cosh(c_l U')} \\ & - \sum_{k=0}^{K-1} \frac{\sum_{l=1}^{\sqrt{M}/2} \exp\left(-c_l^2 R_d\right) c_l U' \sinh(c_l V')}{\sum_{l=1}^{\sqrt{M}/2} \exp\left(-c_l^2 R_d\right) \cosh(c_l V')} \end{aligned} \tag{8-73}$$

where $c_l = 2l-1$, $U' = \sqrt{3/(M-1)}U, V' = \sqrt{3/(M-1)}V$, and, as before, $R_d = P_d T/N_0$ is the data SNR. A closed-loop carrier synchronizer motivated by this MAP estimation approach is illustrated in Fig. 8-14. Here again we

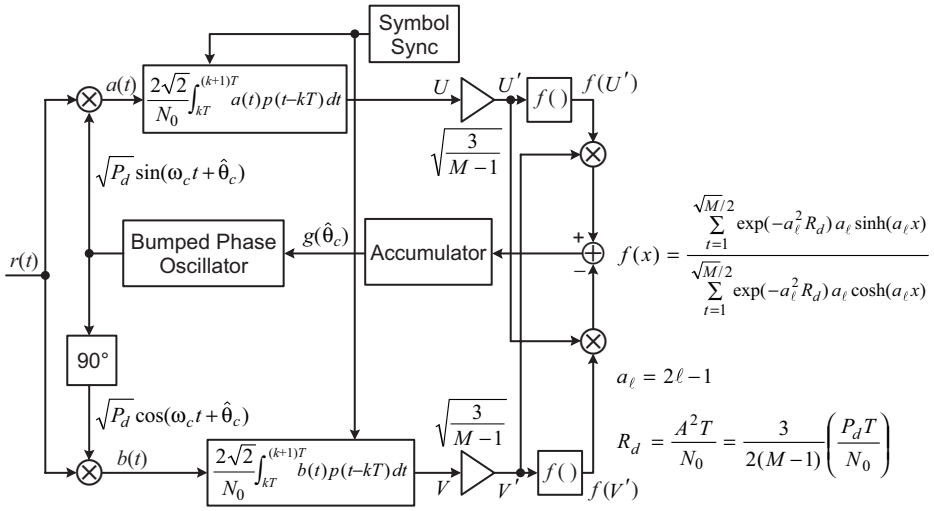


Fig. 8-14. The MAP estimation loop for square QAM.

notice the I-Q front end and crossover (between the I and Q arms) processing with the appropriate nonlinearity to establish the error signal. The passive arm filter implementation based on the large argument (high SNR) approximation of the nonlinearity is illustrated in Fig. 8-15 for $M = 16$. At low SNR, use of the approximations $\sinh \cong x$, $\cosh x \cong 1 + 0.5x^2$, and $\exp x \cong 1 - x$ in Eq. (8-73) results in the carrier synchronization loop of Fig. 8-16, which is identical to the QPSK portion of the canonical structure of Fig. 8-12. Thus, at low SNR, the multilevel nature of the QAM has no bearing on the closed-loop structure motivated by the MAP estimation approach, i.e., it is sufficient to use a QPSK loop.

For unbalanced QPSK, the transmitted signal is of the form

$$s(t,\theta_c) = \sqrt{2P_1}m_1(t)\cos(\omega_c t+\theta_c) + \sqrt{2P_2}m_2(t)\sin(\omega_c t+\theta_c) \qquad (8\text{-}74)$$

where P_1, P_2 are the average signal powers in the I and Q arms, respectively, and $m_I(t), m_Q(t)$ are the quadrature data modulations of rates $1/T_1$ and $1/T_2$ and, in general, different pulse shapes $p_1(t)$ and $p_2(t)$. For this modulation, the derivative of the log-likelihood function becomes

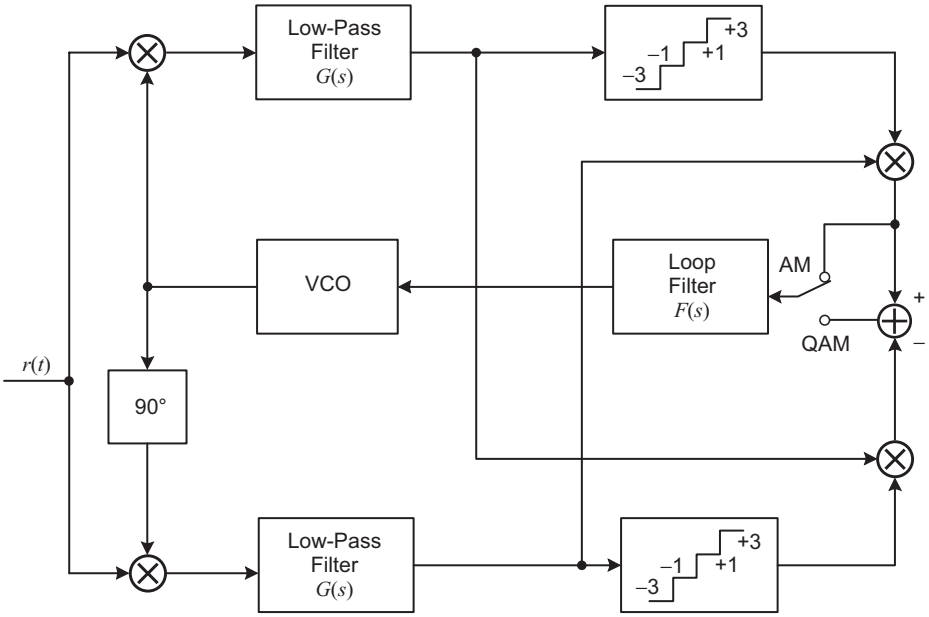


Fig. 8-15. High SNR approximation of the MAP estimation loop for amplitude modulation (AM) and QAM.

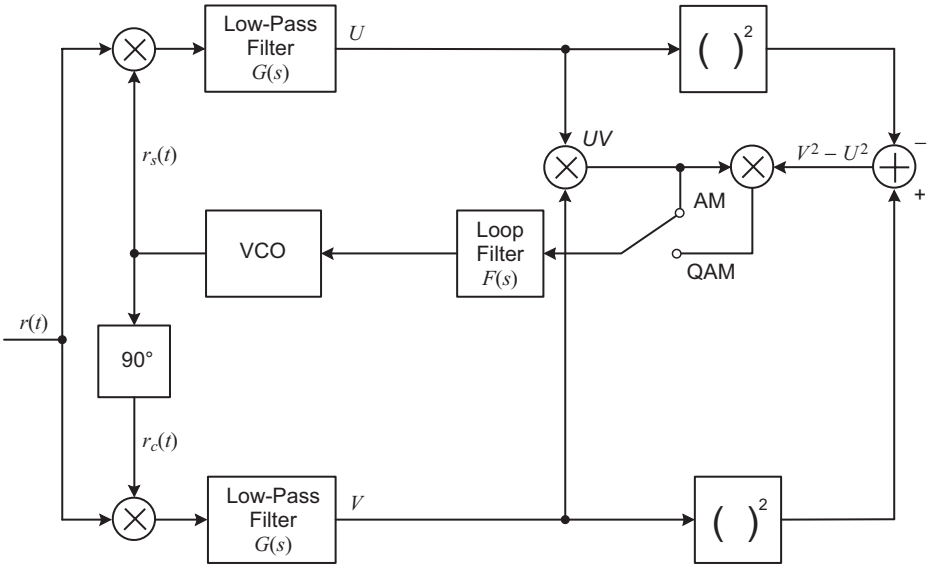


Fig. 8-16. Low SNR approximation for the MAP estimation loop for AM and QAM (the same as the low SNR approximation for PSK and QPSK).

$$\frac{d\Lambda(\theta_c)}{d\theta_c} \cong \sum_{k=0}^{K_2} \left(\frac{2\sqrt{2P_2}}{N_0} \int_{\delta_k}^{\delta_{k+1}} r(t) p_2(t-\delta_k) \cos(\omega_c t + \theta_c) dt \right)$$

$$\times \tanh \left\{ \frac{2\sqrt{2P_2}}{N_0} \int_{\delta_k}^{\delta_{k+1}} r(t) p_2(t-\delta_k) \sin(\omega_c t + \theta_c) dt \right\}$$

$$- \sum_{k=0}^{K_1} \left(\frac{2\sqrt{2P_1}}{N_0} \int_{\tau_k}^{\tau_{k+1}} r(t) p_1(t-\tau_k) \cos(\omega_c t + \theta_c) dt \right)$$

$$\times \tanh \left\{ \frac{2\sqrt{2P_1}}{N_0} \int_{\tau_k}^{\tau_{k+1}} r(t) p_1(t-\tau_k) \sin(\omega_c t + \theta_c) dt \right\} \qquad (8\text{-}75)$$

In Eq. (8-75), $\tau_k (k = 0, 1, \cdots, K_1)$ is the ordered set of time instants at which the modulation $m_1(t)$ may potentially have a symbol transition in the observation interval $(0 \leq t \leq T_o)$. Similarly, $\delta_k (k = 0, 1, \cdots, K_2)$ is the ordered set of time instants at which the modulation $m_2(t)$ may potentially have a symbol transition in the same observation interval. Note that, since we have not restricted T_1 or T_2 to be integer related, it is possible that the observation may not contain an integer number of symbol intervals of one of the two modulations. For this reason, we allow the summations in Eq. (8-75) to extend over $K + 1$ symbols. Furthermore, no restriction is placed on the relative synchronization between the taus and the deltas.

Figure 8-17 illustrates the MAP estimation closed loop that results from using Eq. (8-75) as an error signal. As before, one can use the approximations of the hyperbolic tangent nonlinearity as given in Eq. (8-14) to produce low and high SNR configurations. The difficulty with using the small argument approximation is that, as the ratio of rates and powers both approach unity, i.e., balanced QPSK, the two pairs of matched filters (or equivalently the two pairs of arm filters in the passive implementation) become identical, and thus the error signal at the input to the loop filter goes to zero for all phase errors. Thus, as was done previously in deriving the MAP estimation loop for balanced QPSK, one must consider the first two terms of the power series expansion of the hyperbolic tangent nonlinearity as in Eq. (8-44), which results in the configuration illustrated in Fig. 8-18 (assuming the passive arm filter implementation). Note that this two-channel Costas loop reduces (except for the 1/3 gain factor) to Fig. 8-16 when the transmitted signal becomes balanced QPSK. Thus, this configuration is capable of tracking unbalanced as well as balanced QPSK.

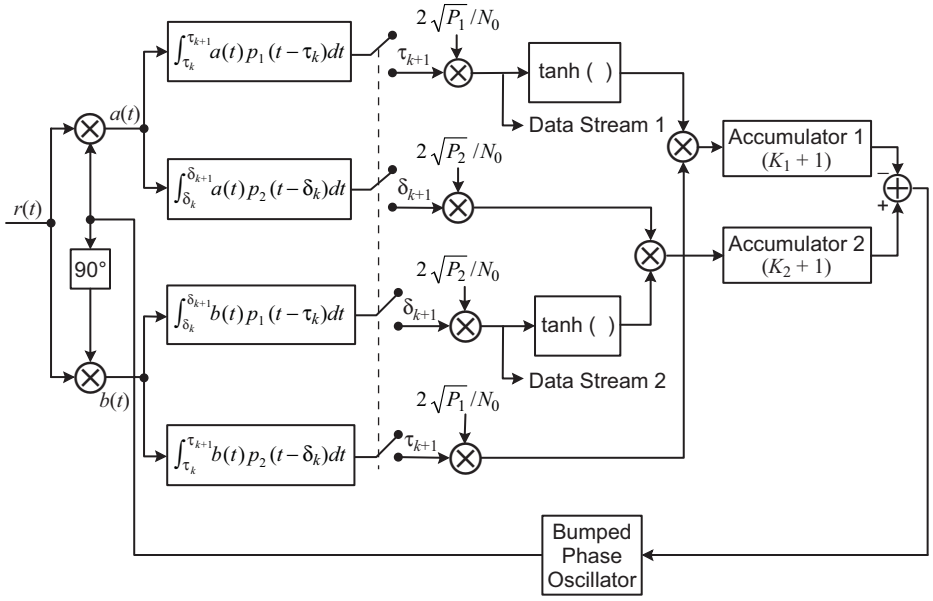


Fig. 8-17. A closed loop motivated by the MAP estimation of carrier phase for unbalanced QPSK.

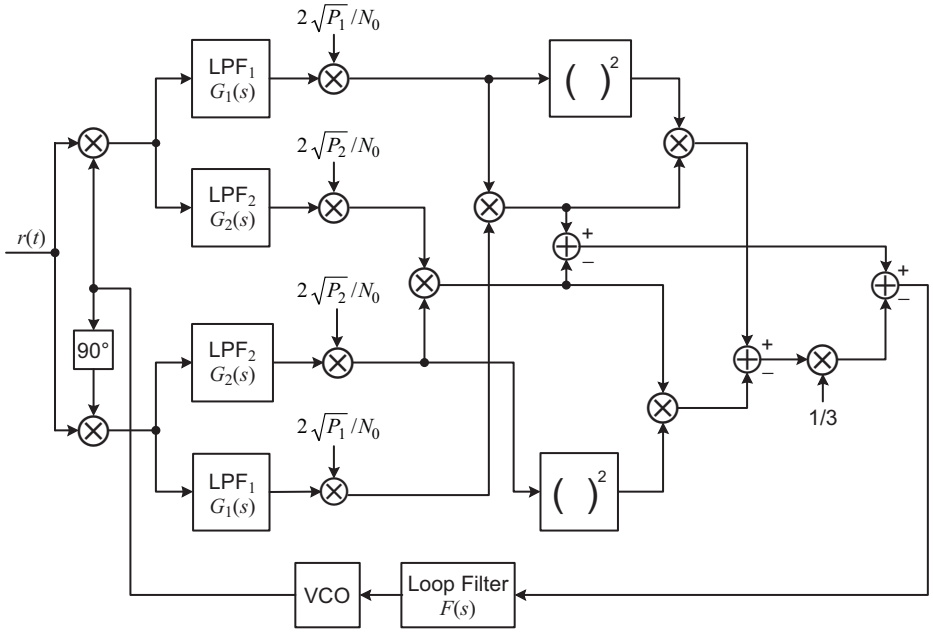


Fig. 8-18. Low SNR approximation for the MAP estimation loop for unbalanced QPSK.

We conclude by noting that, depending on the ratio of powers in the two channel of the UQPSK signal, it is possible to employ just the simple biphase or quadriphase Costas loops previously discussed, bearing in mind, however, that the performance of such would then be inferior to that of the loop in Fig. 8-18.

8.4.3 $\pi/4$ Differentially Encoded QPSK

As a final modulation form, we consider the case of $\pi/4$ differentially encoded QPSK in which the information phase symbols are chosen from the set $(\pm\pi/4, \pm 3\pi/4)$ and are differentially encoded prior to transmission. Denoting the information symbol in the nth symbol interval by $\Delta\phi_n$, then the actual transmitted symbol in the same interval is given by $\phi_n = \phi_{n-1} + \Delta\phi_n$, which, in view of the set used to define $\Delta\phi_n$ as given above, alternates between the allowable sets $(0, \pi/2, \pi, 3\pi/2)$ and $(\pm\pi/4, \pm 3\pi/4)$ in successive transmission intervals. Because of this $\pi/4$ rad rotation of the transmitted signaling constellation from symbol to symbol, the maximum instantaneous phase change between two successive symbols is $3\pi/4$ rad. This is to be compared to a maximum instantaneous phase change of π rad for the case where the signaling constellation does not rotate from symbol to symbol or, equivalently, the information symbols to be differentially encoded are chosen from the set $(0, \pi/2, \pi, 3\pi/2)$. Reducing the maximum phase jump from π to $3\pi/4$ reduces the envelope fluctuation in the signal, which is desirable on nonlinear channels to prevent spectral side lobes from being regenerated after having been filtered.

Applying the same MAP phase estimation approach as previously used to motivate a closed-loop structure, it can be shown that the appropriate carrier synchronizer for this so-called $\pi/4$ differentially encoded QPSK modulation is as illustrated in Fig. 8-19. Once again, if desired, one can apply the appropriate approximations to the hyperbolic tangent function to arrive at low and high SNR implementations of this generic structure.

References

[1] M. K. Simon, "Optimum Receiver Structures for Phase-Multiplexed Modulations," *IEEE Transactions on Communications*, vol. COM-26, pp. 865–872, June 1978.

[2] M. K. Simon, "Further Results on Optimum Receiver Structures for Digital Amplitude and Phase Modulated Signals," *International Conference on Communications (ICC'78) Record*, Toronto, Ontario, Canada, pp. 42.1.1–42.1.7, June 1978.

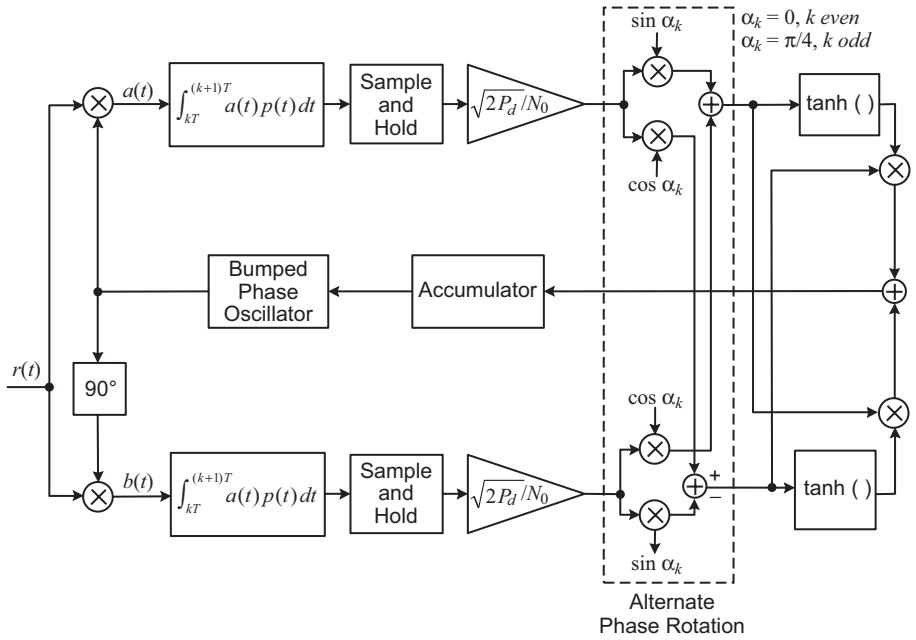


Fig. 8-19. MAP estimation loop for carrier synchronization of $\pi/4$ differentially encoded QPSK.

[3] P. W. Kinman, "TLM-21 DSN Telemetry System, Block-V Receiver," *DSMS Telecommunications Link Design Handbook*, 810-5, Rev. D, Jet Propulsion Laboratory, Pasadena, California, December 1996.

[4] "Proximity-1 Space Link Protocol," CCSDS 211.0-B-1, *Blue Book*, Issue 1, October 2002.

[5] M. K. Simon and S. Butman, "On the Receiver Structure for a Single-Channel, Phase-Coherent Communication System," *Space Program Summary 37-62*, vol. III, Jet Propulsion Laboratory, Pasadena, California, pp. 103–108, April 1970.

[6] M. K. Simon, "On the Optimality of the MAP Estimation Loop for Tracking BPSK and QPSK Signals," *IEEE Transactions on Communications*, vol. COM-27, pp. 158–165, January 1979.

[7] H. L. Van Trees, *Detection, Estimation, and Modulation Theory, Part III*, New York: John Wiley, 1968–1971.

[8] C. R. Cahn, "Improving Frequency Acquisition of a Costas Loop," *IEEE Transactions on Communications*, vol. COM-25, pp. 1453–1459, December 1977.

[9] M. K. Simon and W. C. Lindsey, "Optimum Performance of Suppressed Carrier Receivers with Costas Loop Tracking," *IEEE Transactions on Communications*, vol. COM-25, pp. 215–227, February 1977.

[10] M. K. Simon, "On the Calculation of Squaring Loss in Costas Loops with Arbitrary Arm Filters," *IEEE Transactions on Communications*, vol. COM-26, pp. 179–184, January 1978.

[11] M. K. Simon, "The Effects of Residual Carrier on Costas Loop Performance as Applied to the Shuttle S-Band Uplink," *IEEE Transactions on Communications*, vol. COM-26, pp. 1542–1548, November 1978.

Appendix 8-A

Cramer–Rao Bound on the Variance of the Error in Estimating the Carrier Phase of a BPSK Signal

As we have seen earlier in the chapter, MAP estimation of the carrier phase of a BPSK-modulated carrier can be formulated as the solution to a problem in which a suitably defined likelihood function is maximized with respect to the carrier phase parameter. As a check on the efficiency of the estimator so obtained, one often derives the Cramer–Rao lower bound [7] on the performance measure being considered. Here we derive the Cramer–Rao bound on the variance of the error in the MAP estimation of the phase of a BPSK signal. More often than not, what is typically done in the literature in such applications is to use the result obtained for an unmodulated carrier, which as we shall see is a proper thing to do only at high SNR.

Consider an observation over an interval $T_o = KT$ seconds of a BPSK-modulated carrier in AWGN, where K denotes the number of bits in the observation and T is the bit time (the reciprocal of the bit rate). The received signal in the kth bit time interval $kT \le t \le (k+1)T$ takes the form

$$r(t) = s(t,\theta_c) + n(t) = \sqrt{2P_d}c_k \cos(\omega_c t + \theta_c) + n(t) \tag{A-1}$$

where P_d is the received data power, ω_c is the radian carrier frequency, c_k is the kth bit taking on equiprobable ± 1 values, and $n(t)$ is the AWGN with single-sided power spectral density N_0 W/Hz. The likelihood function for the observation $r(t)$ conditioned on the data sequence $\mathbf{c} = (c_1, c_2, \cdots, c_K)$ and the carrier phase θ_c is well-known to be

$$\begin{aligned} p\big(r(t)\,|\theta_c, \mathbf{c}\,\big) &= C\exp\left\{-\frac{1}{N_0}\int_0^{T_o}\big[r(t) - s(t,\theta_c)\big]^2 dt\right\} \\ &= C\prod_{k=0}^{K-1}\exp\left\{-\frac{1}{N_0}\int_{kT}^{(k+1)T}\big[r(t) - s(t,\theta_c)\big]^2 dt\right\} \end{aligned} \tag{A-2}$$

To determine the Cramer–Rao bound, it is necessary to evaluate the parameter

$$\gamma \triangleq E\left\{\left(\frac{\partial \ln p\big(r\,(t)\,|\theta_c\,\big)}{\partial\theta_c}\right)^2\right\} \tag{A-3}$$

Thus, we must first average Eq. (A-2) over the bit sequence in order to arrive at $p\big(r(t)|\theta_c\big)$. Denoting $s_1(t,\theta_c)$ as the transmitted signal in the kth bit interval when $c_k = 1$ and likewise $s_{-1}\,(t,\theta_c)$ as the transmitted signal in the kth bit interval when $c_k = -1$, then

$$\begin{aligned} p\big(r(t)\,|\theta_c\big) &= C\prod_{k=0}^{K-1}\left[\frac{1}{2}\exp\left\{-\frac{1}{N_0}\int_{(k-1)T}^{kT}\big[r(t) - s_1\,(t,\theta_c)\big]^2 dt\right\}\right. \\ &\quad \left. + \frac{1}{2}\exp\left\{-\frac{1}{N_0}\int_{kT}^{(k+1)T}\big[r(t) - s_{-1}\,(t,\theta_c)\big]^2 dt\right\}\right] \\ &= C\prod_{k=0}^{K-1}\exp\left\{-\frac{1}{N_0}\int_{kT}^{(k+1)T} r^2(t)dt\right\}\exp(-R_d) \\ &\quad \times \cosh\left\{\frac{2}{N_0}\int_{kT}^{(k+1)T} r(t)s_1\,(t,\theta_c)\,dt\right\} \end{aligned} \tag{A-4}$$

where $R_d = P_d T/N_0$ is, as before, the data SNR. Taking the natural logarithm of Eq. (A-4) and differentiating with respect to θ_c gives

$$\begin{aligned}
\frac{\partial \ln p\big(r(t)\,|\theta_c\big)}{\partial \theta_c} &= \sum_{k=0}^{K-1} \frac{\partial}{\partial \theta_c} \ln \cosh \left\{ \frac{2}{N_0} \int_{kT}^{(k+1)T} r(t) s_1\left(t, \theta_c\right) dt \right\} \\
&= \sum_{k=0}^{K-1} \tanh \left\{ \frac{2}{N_0} \int_{kT}^{(k+1)T} r(t) s_1\left(t, \theta_c\right) dt \right\} \\
&\quad \times \left[\left\{ \frac{2}{N_0} \int_{kT}^{(k+1)T} r(t) \frac{\partial s_1\left(t, \theta_c\right)}{\partial \theta_c} dt \right\} \right] \\
&= -\sum_{k=0}^{K-1} \tanh \left\{ \frac{2\sqrt{2P_d}}{N_0} \int_{kT}^{(k+1)T} r(t) \cos\left(\omega_c t + \theta_c\right) dt \right\} \\
&\quad \times \left[\frac{2\sqrt{2P_d}}{N_0} \int_{kT}^{(k+1)T} r(t) \sin\left(\omega_c t + \theta_c\right) dt \right]
\end{aligned} \tag{A-5}$$

Consider first the evaluation of Eq. (A-3) together with Eq. (A-5) for the case of high SNR, where the tanh nonlinearity is approximated by the signum function. Letting

$$\begin{aligned}
I_k &= \frac{2\sqrt{2P_d}}{N_0} \int_{kT}^{(k+1)T} r(t) \cos\left(\omega_c t + \theta_c\right) dt \\
Q_k &= \frac{2\sqrt{2P_d}}{N_0} \int_{kT}^{(k+1)T} r(t) \sin\left(\omega_c t + \theta_c\right) dt
\end{aligned} \tag{A-6}$$

then for high SNR we have

$$\frac{\partial \ln p\big(y(t)\,|\theta_c\big)}{\partial \theta_c} = -\sum_{k=0}^{K-1} Q_k \operatorname{sgn} I_k \tag{A-7}$$

and because the I_k's and Q_k's are iid and independent of each other,

$$E\left\{\left(\frac{\partial \ln p\left(r(t)\,|\theta_c\right)}{\partial \theta_c}\right)^2\right\} = \sum_{k=0}^{K-1} E\left\{Q_k^2 \overbrace{\operatorname{sgn}^2 I_k}^{=1}\right\} + 2\sum_{\substack{k=0\\k\neq l}}^{K-1}\sum_{l=0}^{K-1} E\left\{Q_k \operatorname{sgn} I_k\right\} E\left\{Q_l \operatorname{sgn} I_l\right\} \tag{A-8}$$

Assuming that indeed $s_1(t,\theta_c)$ was transmitted in the kth interval, i.e., $r(t) = s_1(t,\theta_c) + n(t)$, then substituting Eq. (A-6) into Eq. (A-8) and carrying out the expectation over the noise gives, after simplification,

$$\begin{aligned} E\left\{Q_k^2\right\} &= 2R_d \\ E\left\{Q_k \operatorname{sgn} I_k\right\} &= 0 \end{aligned} \tag{A-9}$$

If instead one assumes that the transmitted signal was $s_{-1}(t,\theta_c)$ rather than $s_1(t,\theta_c)$, then one arrives at the identical result as Eq. (A-9). Thus, independent of the actual transmitted data sequence, we have

$$E\left\{\left(\frac{\partial \ln p\left(r(t)\,|\theta_c\right)}{\partial \theta_c}\right)^2\right\} = K\left(2R_d\right) \tag{A-10}$$

Finally, the Cramer–Rao bound on the variance of the unbiased estimation error $\phi \triangleq \theta_c - \hat{\theta}_c$ is given by

$$\sigma_\phi^2 \geq \left[E\left\{\left(\frac{\partial \ln p\left(r(t)\,|\theta_c\right)}{\partial \theta_c}\right)^2\right\}\right]^{-1} = \frac{1}{K\left(2R_d\right)} \tag{A-11}$$

For an unmodulated carrier of energy $E = P_d T_o$ over the observation, the Cramer–Rao bound on the variance of the estimation error is given by

$$\sigma_\phi^2 \geq \frac{1}{2E/N_0} \tag{A-12}$$

which in view of the relation $T_o = KT$ is identical to Eq. (A-8). Thus, as previously mentioned, we see that the Cramer–Rao bound for the modulated carrier is equivalent to that for the unmodulated carrier at high SNR.

For low SNR, one approximates the tanh nonlinearity by a linear function, i.e., $\tanh x = x$. Thus, the analogous relation to Eq. (A-7) is now

$$\frac{\partial \ln p\left(r(t)\,|\theta_c\right)}{\partial \theta_c} = -\sum_{k=1}^{K} Q_k I_k \tag{A-13}$$

Again because of the independence of the I_k's and Q_k's we get

$$E\left\{\left(\frac{\partial \ln p\left(r(t)\,|\theta_c\right)}{\partial \theta_c}\right)^2\right\} = \sum_{k=1}^{K} E\left\{Q_k^2 I_k^2\right\} + 2\sum_{\substack{k=1\\k\neq l}}^{K}\sum_{l=1}^{K} E\left\{Q_k I_k\right\} E\left\{Q_l I_l\right\} \tag{A-14}$$

with (after much simplification)

$$\begin{aligned} E\left\{Q_k^2 I_k^2\right\} &= \left(2R_d\right)^2\left(1+2R_d\right) \\ E\left\{Q_k I_k\right\} &= 0 \end{aligned} \tag{A-15}$$

Finally, substituting Eq. (A-15) into Eq. (A-14) gives the desired result, namely,

$$\sigma_\phi^2 \geq \frac{1}{K\left(2R_d\right)^2\left(1+2R_d\right)} \tag{A-16}$$

Note that at low SNR the bound approximately has an inverse square-law behavior with bit SNR as compared with the inverse linear behavior at high SNR.

It is important to emphasize that Eq. (A-16) is valid only when the denominator on the right-hand side of the equation is large. Thus, it is possible to apply the bound in Eq. (A-16) for small R_d provided that the number of bits in the observation, K, is sufficiently large.

Chapter 9
Modulation Classification

Jon Hamkins and Marvin K. Simon

Modulation classification is the process of deciding, based on observations of the received signal, what modulation is being used at the transmitter. It has long been an important component of noncooperative communications in which a listener desires to intercept an unknown signal from an adversary. It is also becoming increasingly important in cooperative communications, with the advent of the software-defined autonomous radio. Such a radio must configure itself, including what demodulator to use, based on the incoming signal.

In this chapter, we analyze the performance of optimum and sub-optimum modulation classifiers to discriminate M-ary phase-shift keying (M-PSK) from M'-ary phase-shift keying (M'-PSK). The measure of performance to be used is the *probability of misclassification*, i.e., the probability of deciding that M-PSK was transmitted when in fact M'-PSK was transmitted, or vice versa.

After dispensing with preliminaries in Section 9.1, we continue in Section 9.2 with a presentation of approximations to the maximum-likelihood (ML) classifier to discriminate between M-ary and M'-ary PSK transmitted on an additive white Gaussian noise (AWGN) channel and received noncoherently, partially coherently, or coherently, and when symbol timing is either known or unknown. A suboptimum classifier is shown to be ten times less complex than the ML classifier and has less than 0.1 dB of performance loss for symbol signal-to-noise ratios (SNRs) in the range $(-10,10)$ dB and any number of observed symbols. Other methods are shown to reduce complexity by a factor of 100 with less than 0.2 dB of performance loss. We also present a classifier that does not require an estimate of the symbol SNR, and in Section 9.3 we present a threshold optimization technique that improves the high-SNR performance of a previously published classifier. Complexity of the classifiers is discussed in Section 9.4. In

Section 9.5, we derive a classification error floor that exists for any classifier on any memoryless channel, even a noiseless one, by deriving a lower bound on the misclassification probability as a function of the number of observed samples. In Section 9.6, we present numerical results of each of the classifiers along with a summary comparison. In Section 9.7, we examine how symbol timing and modulation type may be jointly estimated. In Section 9.8, we show that, for the specific case of quadrature phase-shift keying (QPSK)/binary phase-shift keying (BPSK) classification, the error floor does not occur if $\pi/4$-QPSK modulation is used instead of QPSK. In Section 9.9, we follow the same ML approach as mentioned above for M-PSK (a special case of which is conventional QPSK) to derive the optimum and approximate classifiers for offset quadrature phase-shift keying (OQPSK) when received noncoherently over the AWGN. Examples are given for the special cases of OQPSK/BPSK and minimum-shift keying (MSK)/QPSK classifications. Finally, in Section 9.10, we discuss modulation classification in the presence of a carrier frequency offset.

9.1 Preliminaries

9.1.1 Signal Model

For ease of exposition, this chapter is limited to binary hypothesis testing in which each hypothesis occurs with equal a priori probability, although the extension to multiple hypotheses and unequal a priori probabilities can be done in the usual way [1]. Throughout, we assume $M < M'$ and each is a power of two, that the modulation type remains the same for N observed received symbols, and that each point of the constellation is transmitted with equal probability. The carrier phase, modulated data, and symbol timing are assumed unknown, while M, M', the symbol duration, signal power, noise variance, and carrier frequency[1] are assumed known.

As in Eq. (1-7), the complex baseband representation of the received M-PSK signal is

$$\tilde{r}(t) = \sqrt{2P_t} \sum_{n=-\infty}^{\infty} e^{j(\theta_n + \theta_c)} p(t - nT - \varepsilon T) + \tilde{n}(t) \tag{9-1}$$

where $2P_t$ is the known signal power of the complex baseband signal (the passband power is P_t); $\theta_n = [2q_n + (1 + (-1)^{M/2})/2]\pi/M$ is the data modulation for the nth M-PSK symbol, with independent and uniformly distributed

[1] Later on in the chapter, we shall consider modulation classification in the presence of a residual carrier frequency error that may exist after frequency correction.

$q_n \in \{0, 1, \cdots, M-1\}$; θ_c is the unknown carrier phase, uniform on $[0, 2\pi)$; $p(t)$ is a pulse shape satisfying $T^{-1}\int_0^T p^2(t)dt = 1$; T is the known symbol duration; ε is the unknown fractional symbol timing, uniform on [0,1); and $\tilde{n}(t)$ is a complex AWGN process with two-sided power-spectral density N_0 W/Hz per dimension (the passband process $n(t)$ has two-sided power spectral density (PSD) $N_0/2$ W/Hz).

The complex observables corresponding to the matched filter outputs at time instants $(n+1+\hat{\varepsilon})T,\ n = 0, 1, 2, \cdots, N-1$ are given by

$$\tilde{r}_n(\hat{\varepsilon}) = \frac{1}{T}\int_{(n+\hat{\varepsilon})T}^{(n+1+\hat{\varepsilon})T} \tilde{r}(t)\, p(t - nT - \hat{\varepsilon}T)\, dt \tag{9-2}$$

A sequence of N observables corresponding to N received symbols is denoted by $\tilde{\boldsymbol{r}}(\hat{\varepsilon}) = \big(\tilde{r}_1(\hat{\varepsilon}), \cdots, \tilde{r}_N(\varepsilon)\big)$. When the timing is known, the matched filter sets $\hat{\varepsilon} = \varepsilon$, which results in the observable

$$\tilde{r}_n = \tilde{r}_n(\varepsilon) = \sqrt{2P_t}e^{j(\theta_n+\theta_c)} + \tilde{n}_n, \quad n = 0, \cdots, N-1 \tag{9-3}$$

where we have dropped the symbol-timing argument. We will use the known-timing assumption throughout the remainder of the chapter, except in Section 9.7. In Eq. (9-3), $\tilde{n}_n = \tilde{n}_{n,R} + j\tilde{n}_{n,I}$ is a complex Gaussian random variable with mean zero, variance $\sigma^2 = N_0/T$ per dimension,[2] and independent components. Initially, we assume that P_t and σ^2 are known at the receiver, although we will drop that assumption later. For convenience, we denote the symbol SNR as

$$\gamma_s = \frac{E_s}{N_0} = \frac{P_t}{\sigma^2} = \frac{P_t T}{N_0} \tag{9-4}$$

9.1.2 Conditional-Likelihood Function

The multivariate Gaussian probability density function of a complex vector $\mathbf{x}$ with mean $\mathbf{m}_x$ has the form [2, Eq. 2.99]

$$p(\mathbf{x}) = \frac{1}{\pi^N|\mathbf{C}|}\exp\left[-(\mathbf{x}-\mathbf{m}_x)^{*T}\mathbf{C}^{-1}(\mathbf{x}-\mathbf{m}_x)\right] \tag{9-5}$$

[2] See Section 11.1 for a more detailed description of the noise variance in the discrete-time model.

where $\mathbf{C}$ is the covariance matrix. The covariance matrix for $\tilde{\boldsymbol{r}}$, given M, $\boldsymbol{\theta} = (\theta_0, \cdots, \theta_{N-1})$, and θ_c, is $\mathbf{C} = 2\sigma^2\mathbf{I}$, where $\mathbf{I}$ is the $N \times N$ identity matrix. Thus, the conditional probability density function of the complex baseband received signal $\tilde{\boldsymbol{r}}$, given P_t, N_0, M, $\boldsymbol{\theta}$, and θ_c, is

$$p(\tilde{\boldsymbol{r}}|M, \boldsymbol{\theta}, \theta_c) = \frac{1}{(2\pi\sigma^2)^N} \exp\left(-\frac{1}{2\sigma^2} \sum_{n=0}^{N-1} \left| \tilde{r}_n - \sqrt{2P_t} e^{j(\theta_n + \theta_c)} \right|^2 \right) \tag{9-6}$$

This may be rewritten as

$$p(\tilde{\boldsymbol{r}}|M, \boldsymbol{\theta}, \theta_c) = C \exp\left(-N\gamma_s + \mathrm{Re}\left\{ \frac{\sqrt{2P_t}}{\sigma^2} e^{-j\theta_c} \sum_{n=0}^{N-1} \tilde{r}_n e^{-j\theta_n} \right\} \right) \tag{9-7}$$

where C does not depend on M and, therefore, drops out of the ratios we are about to form. When Eq. (9-7) is averaged over $\boldsymbol{\theta}$, we obtain what we refer to as the conditional-likelihood function (CLF), i.e., the conditional probability density function of the received vector signal $\tilde{\boldsymbol{r}}$, given M and θ_c. This is given by [3, Eq. B.3b]

$$\mathrm{CLF}_M(\theta_c) = C \exp\left[-N\gamma_s + \sum_{n=0}^{N-1} \ln\left(\frac{2}{M} \sum_{q=0}^{\frac{M}{2}-1} \cosh\left[x_n(q; \theta_c) \right] \right) \right] \tag{9-8}$$

where $x_n(q; \theta_c) = (\sqrt{2P_t}/\sigma^2)\mathrm{Re}\left[\tilde{r}_n e^{-j(\theta_c + [2q + (1 + (-1)^{M/2})/2]\pi/M)} \right]$. We may rewrite Eq. (9-8) as

$$\mathrm{CLF}_M(\theta_c) = C e^{-N\gamma_s} \left(\frac{2}{M} \right)^N \prod_{n=0}^{N-1} \sum_{q=0}^{\frac{M}{2}-1} \cosh\left[x_n(q; \theta_c) \right] \tag{9-9}$$

9.2 Modulation Classifiers

9.2.1 ML Classifiers

The ML modulation classifier results in the minimum probability of classification error if the modulation types occur with equal a priori probability. It can be implemented by comparing the likelihood ratio (LR) of the N observed samples to a unity threshold.

9.2.1.1 ML Noncoherent Classifier Averaging over Data, then Carrier Phase. The CLF in Eq. (9-9) has already been averaged over the unknown data. The LR for the M and M' hypotheses is given by averaging over θ_c, which is uniform over $[0, 2\pi)$, and forming the ratio

$$\mathrm{LR} = \frac{\mathrm{LF}_M}{\mathrm{LF}_{M'}} = \frac{E_{\theta_c}\{\mathrm{CLF}_M(\theta_c)\}}{E_{\theta_c}\{\mathrm{CLF}_{M'}(\theta_c)\}} \tag{9-10}$$

To compute the expectations in Eq. (9-10), typically hundreds of function evaluations are needed. In doing this, $\sqrt{2P_t}/\sigma^2$ must be known to compute $x_n(\cdot;\cdot)$. Note that $\gamma_s = P_t/\sigma^2$ alone is not sufficient to determine $\sqrt{2P_t}/\sigma^2$.

9.2.1.2 ML Noncoherent Classifier Averaging over Carrier Phase, then Data. Alternatively, we may average Eq. (9-7) over the carrier phase first. To do this we rewrite Eq. (9-7) in the form

$$p\big(\tilde{\boldsymbol{r}}\,|M, \boldsymbol{\theta}, \theta_c\,\big)$$

$$= C\exp\left(-N\gamma_s + \mathrm{Re}\left\{\frac{\sqrt{2P_t}}{\sigma^2}\left|\sum_{n=0}^{N-1}\tilde{r}_n e^{-j\theta_n}\right| e^{-j\left[\theta_c - \arg\left(\sum_{n=0}^{N-1}\tilde{r}_n e^{-j\theta_n}\right)\right]}\right\}\right) \tag{9-11}$$

$$= C\exp\left\{-N\gamma_s + \frac{\sqrt{2P_t}}{\sigma^2}\left|\sum_{n=0}^{N-1}\tilde{r}_n e^{-j\theta_n}\right| \cos\left[\theta_c - \arg\left(\sum_{n=0}^{N-1}\tilde{r}_n e^{-j\theta_n}\right)\right]\right\} \tag{9-12}$$

Averaging over the uniform distribution of θ_c gives

$$p\left(\tilde{\boldsymbol{r}}\,|M, \boldsymbol{\theta}\right) = \exp\left(-N\gamma_s\right) I_0\left(\frac{\sqrt{2P_t}}{\sigma^2}\left|\sum_{n=0}^{N-1}\tilde{r}_n e^{-j\theta_n}\right|\right) \tag{9-13}$$

where $I_0\left(x\right)$ is the zero-order modified Bessel function of the first kind with argument x. Next we average over the data phase sequence to obtain

$$\mathrm{LF}_M = p\left(\tilde{\boldsymbol{r}}\,|M\,\right) = \underset{\boldsymbol{\theta}}{E}\left\{Ce^{-N\gamma_s} I_0\left(\frac{\sqrt{2P_t}}{\sigma^2}\left|\sum_{n=0}^{N-1}\tilde{r}_n e^{-j\theta_n}\right|\right)\right\} \tag{9-14}$$

This is feasible for small values of N, i.e., when the M^N values of $\boldsymbol{\theta}$ are relatively manageable.

9.2.1.3 ML Coherent/Partially Coherent Classification. In coherent reception, the carrier phase θ_c is known, and the expectation in Eq. (9-10) degenerates to an evaluation of the CLF at a single point:

$$\mathrm{LR} = \frac{\mathrm{CLF}_M(\theta_c)}{\mathrm{CLF}_{M'}(\theta_c)}$$

This is the optimum statistic for ML classification with coherent reception.

In partially coherent reception, only partial knowledge is available about the carrier phase and thus the randomness is not completely removed. We may account for this in the classifier by using a distribution on θ_c that is different from the uniform distribution assumed in noncoherent reception. For example, a phase-locked loop that tracks a residual carrier may produce an error in its carrier phase estimate that is Tikhonov distributed [4,5]. Other randomness, due to oscillator phase noise, intersymbol interference, or phase ambiguities, for example, may also introduce nonuniform randomness to θ_c.

9.2.2 Suboptimum Classifiers

9.2.2.1 Coarse Integral Approximation in the LR. One way to compute Eq. (9-10) is to accurately numerically evaluate the integrals in each of the numerator and denominator. One can see from Eq. (9-9) that the period of $\mathrm{CLF}_M(\theta)$ is $2\pi/M$. Thus, we can write the likelihood function (LF) for hypothesis $H \in \{M, M'\}$ as[3]

$$\mathrm{LF}_H = \frac{H}{2\pi}\int_0^{2\pi/H} \mathrm{CLF}_H(\theta)d\theta \cong \frac{1}{I}\sum_{i=1}^{I} \mathrm{CLF}_H\left(\frac{2i\pi}{IH}\right) \tag{9-15}$$

where the last approximation becomes an equality as $I \to \infty$ by the rectangular rule for integration. Standard integration algorithms [6] reduce computation by using a nonuniform partition of the interval, but even then, typically more than a hundred CLF evaluations are needed for an accurate integral evaluation. For the problem at hand, however, we needn't necessarily evaluate the integrals accurately. For example, for random realizations of $\tilde{\boldsymbol{r}}$, $\mathrm{CLF}_2(\theta_c)$ and $\mathrm{CLF}_4(\theta_c)$ often differ by more than an order of magnitude. If one of the CLFs is higher than the other for the entire range $\theta_c \in [0, 2\pi)$, then a single test (i.e., $I = 1$) comparing $\mathrm{CLF}_M(\theta)$ to $\mathrm{CLF}_{M'}(\theta)$ at any θ would produce the correct classification. While

[3] Note that Eq. (9-15) could have been written as $1/I\sum_{i=1}^{I}\mathrm{CLF}_H(2i\pi/(IM))]$, since $M < M'$, in which case the sample points of $\mathrm{CLF}_{M'}$ corresponding to the uniform quantization of θ_c would be the same as those of CLF_M.

this property does not hold with high enough probability to produce near-ML performance, we may set the tunable parameter I in Eq. (9-15) to values substantially smaller than required to obtain accurate integral evaluations, without significantly affecting classifier performance. This technique has also been used for MFSK classification [7].

9.2.2.2 The Generalized Likelihood Ratio Test. When $I = 1$ in the method above, the complexity is low, but the performance is poor. However, we can use $I = 1$ and still get near-ML performance by suitably choosing the single value of θ_c at which to evaluate CLF_M. In particular, one can use $\tilde{\boldsymbol{r}}$ to estimate θ_c in the M-PSK hypothesis, specifically using its Mth power to remove the modulation, averaging over the data symbols, and then normalizing the resulting angle:

$$\hat{\theta}_c^{(M)} = \frac{1}{M} \arg \sum_{n=0}^{N-1} \tilde{r}_n^M \tag{9-16}$$

where $\arg(z)$ denotes the angle of the complex quantity z. Equation (9-16) is the ML estimate of θ_c for low SNR, in the sense that it results from low-SNR approximations of the true ML estimate $\theta_{c_{\mathrm{ML}}}$ for M-PSK [8]. A decision based on the likelihood ratio $\mathrm{CLF}_M(\hat{\theta}_{c_{\mathrm{ML}}}^{(M)})/\mathrm{CLF}_{M'}(\hat{\theta}_{c_{\mathrm{ML}}}^{(M')})$ results from what is referred to as a *generalized likelihood ratio test (GLRT)*. A different GLRT arises when $\tilde{\boldsymbol{r}}$ is used to estimate $\boldsymbol{\theta}$ before classification [9]. We appropriately define the test based on $\mathrm{LR} = \mathrm{CLF}_M(\hat{\theta}_c^{(M)})/\mathrm{CLF}_{M'}(\hat{\theta}_c^{(M')})$ as a *quasi-generalized-likelihood ratio test (qGLRT)*, and it is a good approximation to Eq. (9-10) in the sense of being an approximation to coherent-reception classification. The carrier estimates in the two hypotheses are used with the same basic reasoning as in per-survivor processing, the method used to perform joint channel estimation and decoding. The phase ambiguities present in these estimates of θ_c, namely, a $2\pi/M$ ($2\pi/M'$) rad ambiguity for the M-PSK (M'-PSK) hypothesis, do not cause a problem because, as previously mentioned, the CLFs are also periodic with period $2\pi/M(2\pi/M')$.

9.2.2.3 Normalized Quasi-Log-Likelihood Ratio (nqLLR). All other things being equal, we would prefer a classifier that requires the least knowledge of channel parameters. If a classifier required only γ_s, for example, it would be preferred over one requiring P_t and σ^2 separately. Unfortunately, all of the classifiers discussed so far (ML, coarse integral approximate ML, qGLRT) require knowledge of both $\sqrt{2P_t}$ and σ^2 individually, and not simply their ratio.

We now discuss a classifier requiring knowledge of fewer parameters. We begin with the quasi-log-likelihood ratio (qLLR) approximation to the LLR [3,10]:[4]

$$\text{qLLR} = \left| \sum_{n=0}^{N-1} \tilde{r}_n^M \right| \tag{9-17}$$

This arises from Eq. (9-10) using the low SNR approximations $\cosh(x) \cong 1+x^2/2$ and $\ln(1+x) \cong x$, along with the approximation $\ln I_0(x) \cong x$ for large post-detection SNR. Although from its definition the qLLR itself does not require knowledge of P_t and σ^2, its use in making a classification decision requires such knowledge to compute the proper threshold (see Section 9.3). This requirement is avoided by using a normalized qLLR (nqLLR) metric:

$$\text{nqLLR} = \frac{\left| \sum_{n=0}^{N-1} \tilde{r}_n^M \right|}{\sum_{n=0}^{N-1} |\tilde{r}_n|^M} \tag{9-18}$$

Like the qLLR, this metric does not itself require knowledge of P_t, σ^2, or the ratio $\gamma_s = P_t/\sigma^2$; however, unlike the qLLR, the optimum threshold for it is also invariant to scale changes in $\tilde{\boldsymbol{r}}$—doubling $\tilde{\boldsymbol{r}}$ does not affect the nqLLR metric, for example—and it is fairly insensitive to variations of P_t or σ^2 as well [11,12].

9.3 Threshold Optimization

9.3.1 Suboptimality of Previously Derived Thresholds

A critical limitation of the qLLR metric is that it does not approximate the LLR metric precisely enough for the optimum LLR threshold (zero) to be used with success. Indeed, the qLLR metric is always nonnegative, meaning that with a zero threshold (optimum for LLR), it would produce the same decision regardless of the transmitted modulation!

One can optimize the threshold for the qLLR metric itself, instead of using the optimum threshold (zero) for the LLR metric. An approximate analytic solution was given in [10], using the assumptions that the real and imaginary parts of the sum in Eq. (9-17) are jointly Gaussian and have equal variance under

[4] Note that the qLLR does not depend on M' since, as shown in [10], for $M' > M$, the denominator of this approximate LLR is independent of M'.

the two hypotheses. These assumptions lead to the optimized threshold of [10, Eq. 37]:[5]

$$T = (2P_t)^{M/2} V_M I_0^{-1} \left[\exp\left(\frac{N}{2V_M} \right) \right] \tag{9-19}$$

where $V_M = \sum_{l=0}^{M} [(M!)^2 \gamma_s^{-l} / (2l![(M-l)!]^2])$, and $I_0^{-1}(x)$ denotes the inverse of the function $I_0(x)$. However, the equal-variance approximation breaks down at high SNR. For example, in BPSK/QPSK classification, the variance of the real part of the sum in Eq. (9-17) is $2P_tV_2$ when BPSK is sent and $2P_t(V_2+1)$ when QPSK is sent [10, Eq. (A.12)]. At high SNR, $V_2 \cong 1/2$, and thus the variance is nearly three times higher when QPSK is sent. A similar comparison shows the variance of the imaginary part is nearly twice as high when QPSK is sent. Therefore, although the threshold in Eq. (9-19) may be near-optimum at low SNR, at high SNR it will be too low.

The further approximation $I_0(x) \cong e^x$, valid for large x (high post-detection SNR), leads to the approximate threshold [10, Eq. 38]

$$T = \frac{N(2P_t)^{M/2}}{2} \tag{9-20}$$

This approximate threshold actually outperforms the "optimum" threshold in Eq. (9-19) for $\gamma_s > 4$ dB.

9.3.2 Empirical Threshold Optimization

The analytical derivation of appropriate thresholds for the qLLR in Eqs. (9-19) and (9-20) depended on the particular form of the metric, and involved some approximation. As an alternative, we present an empirical method to optimize the threshold that can be used for any classifier metric, including the qLLR and nqLLR metrics. Empirical threshold optimization has been briefly mentioned in M-PSK classification work [13] and in QAM classification [14]. However, the degree to which the analytic, approximate thresholds Eqs. (9-19) and (9-20) degrade performance has not been noted. Indeed, the fact that the "approximate" threshold Eq. (9-20) outperforms the "optimum" threshold Eq. (9-19) at high SNR is an indication that the assumptions in the derivation break down at high SNR.

---

[5] The original presentation in [10] used $\tilde{\boldsymbol{r}}/\sigma$ in place of $\tilde{\boldsymbol{r}}$ in the qLLR metric in Eq. (9-17), so that for each n, the noise variance of $\tilde{r}_n$ was unity in each dimension, not σ^2. To account for the difference, the thresholds we give in Eqs (9-19) and (9-20) are a factor $\sigma^M = (N_0/T)^{M/2}$ times what was in the original presentation in [10].

Empirical threshold determination consists of the following process:

(1) Generate a large number of received noisy M-PSK and M'-PSK samples according to Eq. (9-3).

(2) Group the samples into blocks of length N, and compute a set of sample classifier metrics (e.g., by Eq. (9-17) for the qLLR classifier).

(3) Sort the metrics for M-PSK and M'-PSK received symbols separately, in order of increasing value.

(4) For each observed metric of value α, in increasing order:

   (a) Count the number of observed M-PSK metrics having value less than α.

   (b) Count the number of observed M'-PSK metrics having value greater than α.

   (c) Compute the probability of misclassification when using threshold α, using the sum of the above counts.

(5) Report the threshold that minimizes the probability of misclassification.

This procedure may be implemented efficiently enough that a desktop machine can generate about 100 million samples and determine an empirically optimum threshold in less than an hour. The following additional points should be noted:

(1) Threshold optimization need be performed only once, offline. In a practical implementation, given N and γ_s, a table look-up may be used to determine the threshold.

(2) The optimum threshold for the LR metric and its approximation in Eq. (9-15) is zero and requires no empirical optimization.

(3) The optimum threshold for the qLLR metric for BPSK/QPSK classification is relatively flat over a broad region of γ_s, being near 0.6 for $\gamma_s > -5$ dB. This is a desirable characteristic, because γ_s may not be known exactly.

(4) The optimum threshold for the nqLLR metric is also quite flat, and ranges only from about 0.15 to 0.6 for -10 dB $< \gamma_s < 10$ dB. At high SNR, it is approximately equal to the optimum qLLR$/(2NP_t)$ threshold, because the qLLR$/(2NP_t)$ metric and the nqLLR metric are nearly identical in that region. Thus, the nqLLR and qLLR classifiers have similar performance, and the nqLLR has the advantage that neither P_t nor σ^2 need be known.

To see how the nqLLR classifier may be used without any knowledge of P_t or σ^2, consider the problem of BPSK/QPSK modulation classification. As per Eq. (9-20), for BPSK/QPSK classification, the qLLR metric can be compared to a threshold of NP_t. Since $\sum_{n=0}^{N-1} |\tilde{r}_n|^2$ is a good estimate of NP_t, the nqLLR classifier may be compared to 0.5, and gives performance about the same as the qLLR classifier. In fact, optimum thresholds for the nqLLR classifier were found to range only between 0.15 and 0.6 for $P_t \in (0, \infty)$. This compares to optimum thresholds between 0 and ∞ for the qLLR classifier.

9.4 Complexity

9.4.1 ML Classifier

It is suggested in [3,10] that the ML classifier is not practical. However, if computation of $E_{\theta_c}\{\mathrm{CLF}(\theta_c)\}$ requires only a constant number of evaluations of $\mathrm{CLF}(\theta_c)$, then Eq. (9-10) can be computed in $O(N)$ time. This is because, under either hypothesis (M-PSK or M'-PSK), $\mathrm{CLF}(\theta_c)$ is a product of N items, each of which can be computed in $O(1)$ time. In simulations with $N = 100$, $M = 2$, and $M' = 4$, it was found that 100 to 150 CLF evaluations were needed to compute the expectation, or about 100,000 hyperbolic cosine function evaluations overall.

Unfortunately, it is not true that a simple computation of $E_{\theta_c}\{\mathrm{CLF}(\theta_c)\}$ can be computed in $O(N)$ time, because the $\mathrm{CLF}(\theta_c)$ has first derivatives that are exponential in N, and thus the numerical computation of the integral requires finer partitions for higher values of N. This exponential relationship can be seen as follows. By elementary calculus, the first derivative of $\mathrm{CLF}_2(\theta_c)$ is

$$\mathrm{CLF}_2'(\theta_c) = Ce^{-N\gamma_s} \sum_{l=0}^{N-1} \left\{ \sinh\left[\frac{\sqrt{2P_t}}{\sigma^2} \mathrm{Re}\left(\tilde{r}_l e^{-j\theta_c}\right)\right] \frac{\sqrt{2P_t}}{\sigma^2} \mathrm{Im}\left(\tilde{r}_l e^{-j\theta_c}\right) \right.$$

$$\left. \times \prod_{n=0, n\neq l}^{N-1} \cosh\left[\frac{\sqrt{2P_t}}{\sigma^2} \mathrm{Re}\left(\tilde{r}_n e^{-j\theta_c}\right)\right] \right\} \tag{9-21}$$

If θ_c is small, the SNR is large, and $\tilde{\boldsymbol{r}} = \sqrt{2P_t} \cdot (1, \cdots, 1)$, then keeping terms of first order in θ_c in Eq. (9-21) yields

$$\mathrm{CLF}_2'(\theta_c) \cong -2CN\gamma_s\theta_c \left(\frac{e^{\gamma_s}}{2}\right)^N \tag{9-22}$$

Since $\mathrm{CLF}'(\theta_c)$ can be exponential in N, no numerical integration that partitions the domain of the integral into a number of intervals independent of N will produce a correct calculation as N increases. That is, for higher values of N, a finer partition of the domain is required and thus an increasing number of evaluations of the CLF. Therefore, computing the LR in this way requires more than $O(N)$ time.

9.4.2 Coarse Integral Approximate ML classifier

The approximate ML metric of Eq. (9-15) may be computed in $O(IN)$ time because it is the sum of I terms, each of which is a product of N terms, each of which takes $O(1)$ time to compute. If the integration range is partitioned in the same way under each hypothesis, each hyperbolic cosine evaluation used in the numerator of Eq. (9-10) is also used in the denominator, and thus, only $M'/2$ hyperbolic cosine evaluations are needed for each (i, n) pair. Also, we may write

$$x_n\left(q; \frac{2i\pi}{IM}\right) = \frac{\sqrt{2P_t}}{\sigma^2}\left[\mathrm{Re}[\tilde{r}_n]\cos\left(\alpha(q,i)\right) + \mathrm{Im}[r_n]\sin\left(\alpha(q,i)\right)\right]$$

where $\alpha(q,i) = 2\pi(i/I + q)/M$. The trigonometric terms for all $\alpha(\cdot,\cdot)$ may be precomputed and stored in a table, since they do not depend on $\tilde{\boldsymbol{r}}$. Thus, only addition, multiplication, and $M'/2$ hyperbolic cosine evaluations are required for each (i, n) pair.

9.4.3 qGLRT Classifier

The qGLRT estimator/classifier uses $I = 1$, which offers a speed advantage by a factor of 12 over the $I = 12$ case discussed above. It also requires the two phase estimates, however. These take $O(N)$ to compute and were observed to roughly cut the speed gain in half, i.e., 6 times faster than the $I = 12$ case.

9.4.4 qLLR and nqLLR

The qLLR and nqLLR classifiers also have $O(N)$ complexity, the lowest complexity of those considered in this chapter. They do not require any exponential, logarithmic, and trigonometric function evaluations—only multiplication, division, and magnitude operations. As a result, they were observed to offer speed increases of two orders of magnitude over ML classification.

The qLLR metric with threshold given by Eq. (9-19) or Eq. (9-20) requires knowledge of both P_t and σ^2. The approximate threshold of Eq. (9-20) performs worse but does not require knowledge of σ^2. The nqLLR classifier benefits from

knowledge of γ_s, but not from knowing P_t and σ^2 individually. Furthermore, among all classifiers considered, it is unique in providing acceptable performance without knowledge of either P_t or σ^2. If we set a fixed threshold of 0.4, performance is within approximately 1 dB of the ML classifier at $\gamma_s = 1$ and within 3.5 dB at very low γ_s.

9.5 Classification Error Floor

As we will see in Section 9.6, the ML classifier for $N = 10$ appears to exhibit a classification error floor at about 0.001. In this section, we analytically verify the validity of this observation by proving that an error floor of 2^{-N} occurs for BPSK/QPSK classification, which for $N = 10$ is about 9.8×10^{-4}. We also provide a more general error floor derivation for M-PSK/M'-PSK classification. The error floor can be explained by the probability that a randomly generated M'-PSK signal N-vector is consistent with an M-PSK transmission. The probability of this event can be obtained using the method of types [15]. This error floor exists for any M-PSK/M'-PSK classifier and gives an immediate lower bound on the number of observations N that must be made in order to achieve a given classification error rate, even when no channel impairments are present.

Theorem 9-1. *If N independent complex baseband symbols from an a priori equiprobable M-PSK or M'-PSK signal, $M < M'$, are observed at the output of a noncoherent noiseless channel, the minimum probability of modulation misclassification is $(M/M')^{N-1}/2$.*

Proof. Let $\tilde{r}_n$ be as in Eq. (9-3), with M-PSK and M'-PSK equiprobable and $\tilde{n}_n = 0$ for all $0 \le n \le N-1$. Since θ_c is uniformly distributed, $\tilde{r}_n$ is uniformly distributed on the circle of radius $\sqrt{2P_t}$, regardless of θ_n. Thus, when $N = 1$ the observed signals for M-PSK and M'-PSK are identically distributed and the minimum misclassification probability is 1/2.

Now suppose $N > 1$. Let $\boldsymbol{b} = (b_0, \cdots, b_{N-1})$, where $b_n = \theta_n + \theta_c$. From $(\tilde{r}_0, \cdots, \tilde{r}_{N-1})$, we may ascertain P_t and $\boldsymbol{b}$, and vice versa. Since P_t does not depend on the modulation order, $\boldsymbol{b}$ is a sufficient statistic for the optimum (minimum probability of misclassification) classifier. If there exists some n, $0 < n \le N-1$, such that $b_n \ne b_0 \bmod 2\pi/M$, then the signal cannot be M-PSK, and the optimum classifier decides that M'-PSK was sent, with no probability of error. Otherwise, for all $n = 0, \cdots, N-1$, $b_n = b_0 \bmod 2\pi/M$. Under the M-PSK hypothesis, $P(b_n = b_0 \bmod 2\pi/M | M\text{-PSK}) = 1$. Under the M'-PSK hypothesis, for $n > 1$, $P(b_n = b_0 \bmod 2\pi/M | M'\text{-PSK}) = M/M'$, since $b_n - b_0 \bmod 2\pi/M$ takes on the M'/M values in $\{0, 2\pi/M', 2 \cdot 2\pi/M', \cdots, ((M'/M) - 1) \cdot 2\pi/M')\}$ with equal probability. By the independence of the modulation symbols,

$$P(b_0 = \cdots = b_{N-1} \bmod 2\pi/M | M'\text{-PSK}) = \left(\frac{M}{M'}\right)^{N-1}$$

Thus, for all $N > 1$, the M-PSK-consistent event more probably arises from M-PSK than from M'-PSK, and the optimum classifier decides M-PSK. In summary, the optimum classifier is incorrect with probability

$$P_e = P(M'\text{-PSK})P_{e|M'\text{-PSK}} + P(M\text{-PSK})P_{e|M\text{-PSK}}$$

$$= \frac{1}{2} \cdot \left(\frac{M}{M'}\right)^{N-1} + \frac{1}{2} \cdot 0 = \frac{1}{2}\left(\frac{M}{M'}\right)^{N-1} \qquad \square$$

Corollary 9-1. *If N independent complex baseband symbols from an a priori equiprobable BPSK or QPSK signal are observed at the output of a noncoherent noiseless channel, the minimum probability of modulation misclassification is 2^{-N}, which constitutes an error floor.*

Corollary 9-2. *Achieving probability of BPSK/QPSK misclassification below 10^{-6} requires $N \geq 20$ observed samples.*

Proof. $2^{-20} \cong 9.54 \times 10^{-7}$. $\square$

9.6 Numerical Results

To illustrate the numerical performance of the proposed noncoherent classifiers, we consider the example of $M = 2$ and $M' = 4$, i.e., BPSK/QPSK classification, when the timing is known. The classifiers were each implemented in the C programming language.

The simulated performance of the ML BPSK/QPSK classifier is shown in Fig. 9-1, for various numbers of observed symbols N. The classification error floor of Theorem 9-1 is evident, and the high-SNR asymptote can be seen to be 2^{-N}.

The classifier based on coarse integral approximation with $I = 12$ is shown in Fig. 9-2 for various values of N. The coarse integral method has performance indistinguishable from that of ML classification, and reduces the complexity (number of CLF function-evaluations) by an order of magnitude. A C implementation on a Linux desktop computer required approximately 11 milliseconds to compute the ML classification metric, and 1 millisecond to compute the coarse integral approximation metric with $I = 12$.

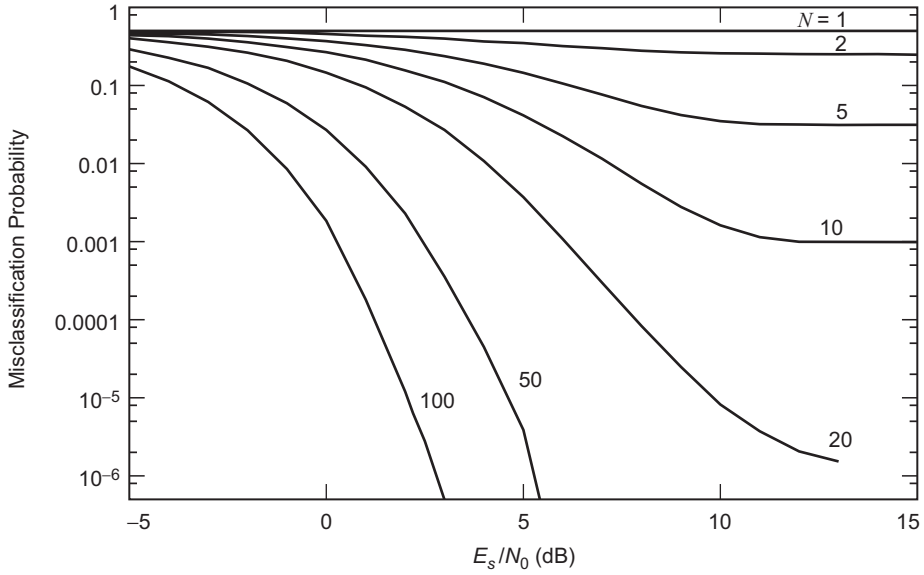


Fig. 9-1. ML BPSK/QPSK classification performance for various N.

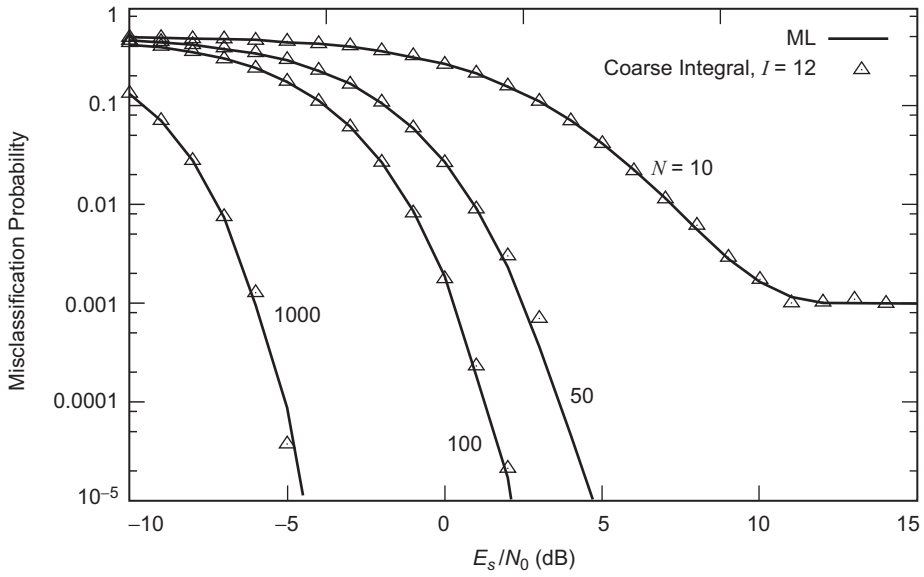


Fig. 9-2. The performance of the ML and coarse integral classifier with I = 12.

For smaller values of I, performance begins to degrade. When $I = 5$, speed increased approximately by a factor of 12/5 compared to $I = 12$, and performance was similar except at high SNR, where it degraded by about 0.4 dB, as can be seen in Fig. 9-3. When $I = 1$, the classifier fails, but by choosing the single CLF evaluation point using the qGLRT, the performance improves. At high SNR, it is virtually in agreement with the ML performance, because the estimates of θ_c are quite good. At low SNR, the performance degrades slightly. The qGLRT classifier was found to be 61 times faster than the ML classifier.

For comparison, the ML coherent classifier is also illustrated in Fig. 9-3. The performance is within 0.5 of the ML noncoherent classifier over a wide range of SNR.

The effect of threshold optimization is shown in Fig. 9-4. The qLLR metric used with either of the proposed thresholds in [10] results in a classifier whose gap from ML increases with increasing E_s/N_0. In fact, the analytically derived threshold is outperformed by its approximation when $E_s/N_0 > 4$ dB! The empirical optimization of the threshold reduces the gap from about 2.5 dB, at $E_s/N_0 = 4$ dB, to 1 dB.

The performance of the nqLLR classifier is also shown in Fig. 9-4. Although the normalization in the nqLLR was motivated by reducing the number of parameters that need to be estimated—the nqLLR doesn't require knowledge of the noise variance—serendipitously, the normalization also has a beneficial effect in the classification performance itself. It can be seen from Fig. 9-4 that at $\gamma_s = 1$ dB, the performance of nqLLR is 0.5 dB better than qLLR, and about 0.5 from the ML limit.

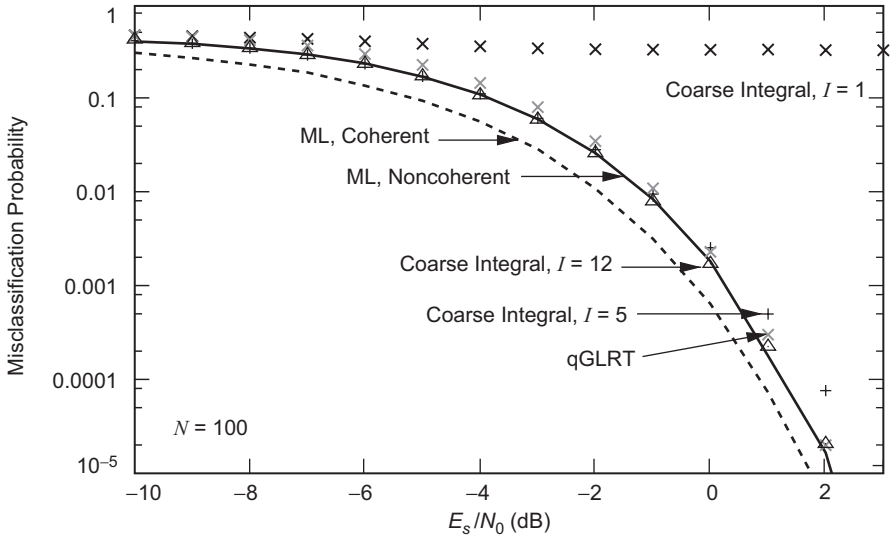


Fig. 9-3. Noncoherent classifier performance as a function of I.

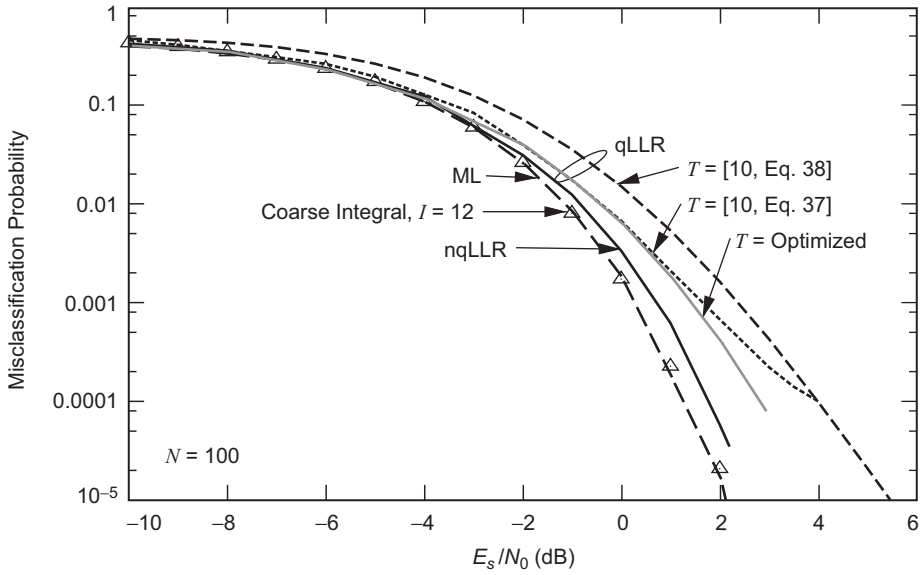


Fig. 9-4. Comparison of ML, coarse integral classifier with I = 12, nqLLR, and qLLR classifiers for various thresholds T.

The values of the empirically optimized thresholds for the qLLR and nqLLR are shown in Fig. 9-5. For low SNR, the empirically optimized threshold for the qLLR metric closely matches the analytic threshold Eq. (9-19), as expected, since the analytic threshold was derived with a low-SNR approximation. At higher SNRs, the optimum threshold is near 0.6, while both Eqs. (9-19) and (9-20) converge to 0.5. The optimum nqLLR threshold also converges to about 0.6, which is also expected because the nqLLR metric converges to the qLLR metric at high SNR.

To evaluate classifier performance with imperfect symbol timing, we consider the case of a fixed fractional symbol timing offset $\tau = (\hat{\varepsilon} - \varepsilon)$, where $\tau \in (-1/2, 1/2)$. When two adjacent symbols are the same, the matched filter output does not depend on the value of τ. On the other hand, when adjacent signals are antipodal, the effective SNR at the matched filter output is degraded by a factor $1 - 2|\tau|$. Thus, in the worst case, a classifier could be degraded by $10 \log_{10}(1 - 2|\tau|)$ dB in SNR. Figure 9-6 shows this upper bound for the ML BPSK/QPSK classifier when $N = 100$, as a function of τ.

Table 9-1 summarizes various attributes of the classifiers. The bold entries highlight favorable performance-complexity trade-offs. If performance within 0.1 dB of the ML classifier is desired, the coarse integral method can achieve it with a speed an order of magnitude higher than the ML classifier. For operation within 0.2 dB, the qGLRT estimator/classifier or nqLLR classifier may be used,

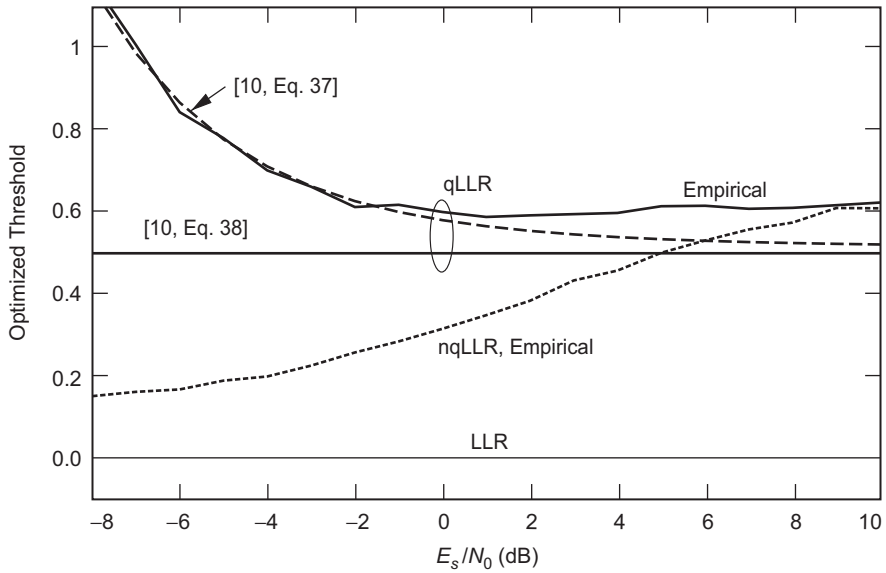


Fig. 9-5. Dependence of empirically optimized thresholds on SNR.

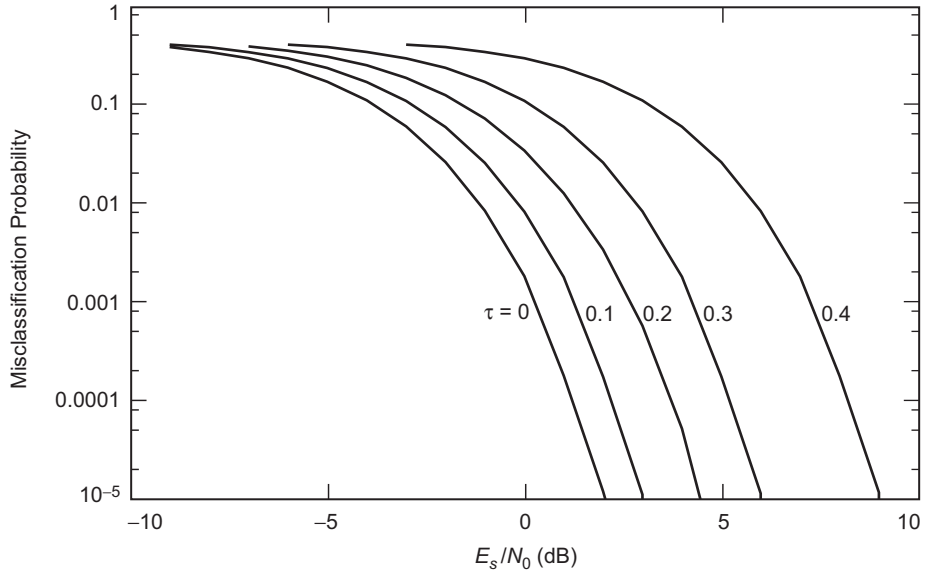


Fig. 9-6. Upper bound on ML BPSK/QPSK classifier performance for various fixed fractional symbol timing offsets τ.

Table 9-1. A summary comparison of BPSK/QPSK noncoherent classifiers, $N = 100$.

Classifier	Equation	Threshold	Needed parameters		dB gap to ML at $\gamma_s =$			Complexity[a]	
			γ_s	P_t	-5	-2	1	Ops	Speed-up
ML	(9-10)	1	√	√	0	0	0	100,000	1
Coarse	(9-15), $I = 12$	1	√	√	**0.0**	**0.0**	**0.1**	7,200	**11**
integral	(9-15), $I = 5$	1	√	√	0.0	0.1	0.4	3,000	26
approx.	(9-5), $I = 1$	1	√	√	4.0	5.5	8.5	600	132
to ML	(9-15), $I = 1$, qGLRT	1	√	√	0.8	0.4	**0.2**	600	**61**
qLLR	(9-17)	(9-19)	√	√	0.4	0.5	1.1	0	165
		(9-20)	—	√	1.7	1.3	1.7	0	165
		Empirical	√	√	0.0	0.5	1.0	0	165
nqLLR	(9-18)	Empirical	—	√	**0.0**	**0.2**	0.5	0	**99**
		0.4	—	—	3.5	2.0	1.0	0	99

[a] Ops is the number of exponential, logarithmic, trigonometric, or Bessel function evaluations required. Speed-up is the observed relative speed compared to the ML classifier.

depending on the SNR, for a speed-up factor of 61 to 99 over the ML classifier. If an SNR estimate is not available, the nqLLR metric can be used to operate within 0.5 dB of the ML classifier, at a speed-up factor of 99 over ML. The qLLR classifier is 165 times faster than ML, and can be used if losses larger than 1 dB can be tolerated.

All classifiers are subject to an error floor that is a function of the number of observed symbols, even on channels without impairments.

9.7 Unknown Symbol Timing

Thus far we have discussed exact and approximate likelihood-based noncoherent classifiers of M-PSK signals under the assumption of unknown (uniformly distributed) carrier phase, but perfectly known fractional symbol timing. In this section, we extend these classifiers to the case where the symbol timing is unknown and also uniformly distributed.

Following the same likelihood function approach as above, where we first average over the data symbol distribution, we eventually arrive at a CLF analogous to Eq. (9-8), namely,

$$\begin{aligned}\mathrm{CLF}_M(\theta_c,\varepsilon) &\triangleq p(\tilde{\boldsymbol{r}}\,|M,\theta_c,\varepsilon) \\ &= C\exp\left[-N\gamma_s+\sum_{n=0}^{N-1}\ln\left(\frac{2}{M}\sum_{q=0}^{M/2-1}\cosh\left[x_n(q;\theta_c,\varepsilon)\right]\right)\right] \end{aligned} \tag{9-23}$$

where ε denotes the unknown fractional symbol timing and

$$x_n(q;\theta_c,\varepsilon) \triangleq \frac{\sqrt{2P_t}}{\sigma^2}\mathrm{Re}\left\{\tilde{r}_n(\varepsilon)\,e^{-j(2\pi q/M+\theta_c)}\right\} \tag{9-24}$$

Thus, a comparison between BPSK and QPSK would be based on the LR

$$\begin{aligned}\mathrm{LR} &= \frac{\underset{\theta_c,\varepsilon}{E}\{\mathrm{CLF}_2(\theta_c,\varepsilon)\}}{\underset{\theta_c,\varepsilon}{E}\{\mathrm{CLF}_4(\theta_c,\varepsilon)\}} \\ &= \frac{\underset{\theta_c,\varepsilon}{E}\left\{\exp\left[\sum_{n=0}^{N-1}\ln\cosh\left[x_n(0;\theta_c,\varepsilon)\right]\right]\right\}}{\underset{\theta_c,\varepsilon}{E}\left\{\exp\left[\sum_{n=0}^{N-1}\ln\left(\frac{1}{2}\left[\cosh\left[x_n(0;\theta_c,\varepsilon)\right]+\cosh\left[x_n(1;\theta_c,\varepsilon)\right]\right]\right)\right]\right\}} \end{aligned} \tag{9-25}$$

where from Eq. (9-24)[6]

$$\begin{aligned} x_n(0;\theta_c,\varepsilon) &\triangleq \frac{\sqrt{2P_t}}{\sigma^2}\mathrm{Re}\left\{\tilde{r}_n(\varepsilon)\,e^{-j\theta_c}\right\} \\ x_n(1;\theta_c,\varepsilon) &\triangleq \frac{\sqrt{2P_t}}{\sigma^2}\mathrm{Im}\left\{\tilde{r}_n(\varepsilon)\,e^{-j\theta_c}\right\} \end{aligned} \tag{9-26}$$

It has previously been shown that an efficient way of evaluating the averages over the unknown parameters, in this case θ_c and ε, is to apply a rectangular

[6] We slightly abuse the notation $x_n(q;\theta_c,\epsilon)$ by not explicitly listing its dependence on M. In this case, $x_n(0;\theta_c,\epsilon)$ for BPSK is the same as it is for QPSK, and $x_n(1;\theta_c,\epsilon)$ does not exist for BPSK. For higher orders, one needs to be more careful.

numerical integration rule. Since the CLF of Eq. (9-23) is periodic in θ_c with period $2\pi/M$, the LR of Eq. (9-25) can be expressed as

$$\mathrm{LR} = \frac{\lim\limits_{I,J\to\infty} \frac{1}{I}\frac{1}{J}\sum\limits_{i=1}^{I}\sum\limits_{j=1}^{J} \mathrm{CLF}_2\left(\frac{i\pi}{I}, \frac{j}{J}\right)}{\lim\limits_{I,J\to\infty} \frac{1}{I}\frac{1}{J}\sum\limits_{i=1}^{I}\sum\limits_{j=1}^{J} \mathrm{CLF}_4\left(\frac{i\pi}{2I}, \frac{j}{J}\right)} \tag{9-27}$$

To reduce the complexity of Eq. (9-27) still further, it has also been suggested one use $I = 1$ (and now also $J = 1$), i.e., evaluate the CLFs at a single value of θ_c and ε, where these values are obtained as functions of the same set of observables used to form the CLFs themselves. Perhaps the best set of values to use are the ML estimates of these parameters. In principle, these ML estimates should be obtained *jointly* by simultaneously maximizing the LF (or equivalently its logarithm) with respect to both θ_c and ε. Specifically, for the numerator of the LF we would use

$$\begin{aligned}\hat{\theta}_{c2}, \hat{\varepsilon}_2 &= \underset{\theta_c,\varepsilon}{\operatorname{argmax}} \sum_{n=0}^{N-1} \ln\cosh\left[x_n\left(0;\theta_c,\varepsilon\right)\right] \\ &= \underset{\theta_c,\varepsilon}{\operatorname{argmax}} \sum_{n=0}^{N-1} \ln\cosh\left[\frac{\sqrt{2P_t}}{\sigma^2}\mathrm{Re}\left\{\tilde{r}_n\left(\varepsilon\right)e^{-j\theta_c}\right\}\right]\end{aligned} \tag{9-28}$$

while for the denominator of the LF we would use

$$\begin{aligned}\hat{\theta}_{c_4}, \hat{\varepsilon}_4 = \underset{\theta_c,\varepsilon}{\operatorname{argmax}} \sum_{n=0}^{N-1} \ln\Bigg\{&\frac{1}{2}\cosh\left[\frac{\sqrt{2P_t}}{\sigma^2}\mathrm{Re}\left\{\tilde{r}_n\left(\varepsilon\right)e^{-j\theta_c}\right\}\right] \\ &+ \frac{1}{2}\cosh\left[\frac{\sqrt{2P_t}}{\sigma^2}\mathrm{Im}\left\{\tilde{r}_n\left(\varepsilon\right)e^{-j\theta_c}\right\}\right]\Bigg\}\end{aligned} \tag{9-29}$$

Then, the low complexity LR to be used for classification would be given by

$$\mathrm{LR} = \frac{\mathrm{CLF}_2\left(\hat{\theta}_{c_2}, \hat{\varepsilon}_2\right)}{\mathrm{CLF}_4\left(\hat{\theta}_{c_4}, \hat{\varepsilon}_4\right)} \tag{9-30}$$

which when compared to a threshold again results in a GLRT. A derivation of the estimates $\hat{\theta}_{c_2}, \hat{\varepsilon}_2$ and $\hat{\theta}_{c_4}, \hat{\varepsilon}_4$ is contained in Appendix 9-A.

9.8 BPSK/π/4-QPSK Classification

As previously discussed in Section 9-5, for noncoherent BPSK/QPSK classification, the misclassification probability exhibits an error floor in the limit of large SNR. This occurs because, in the absence of noise, there is a finite probability that a sequence of N QPSK symbols could be identical (aside from a possible fixed phase rotation over the entire sequence) to a given sequence of N BPSK symbols, and thus the two modulations would be indistinguishable at the receiver, where the classification decision takes place. If instead of transmitting QPSK, one were to instead transmit $\pi/4$-QPSK [16] wherein the signal constellation is rotated back and forth by $\pi/4$ rad every other symbol, then it is no longer possible that a sequence of N QPSK symbols could be "identical" to a given sequence of N BPSK symbols. As such, there would no longer be an error floor in the misclassification probability performance.

In this section we investigate the ML and approximate ML algorithms appropriate to a classification decision between BPSK and $\pi/4$-QPSK.

9.8.1 ML Noncoherent Classifier Averaging over Data, then Carrier Phase

Analogous to the CLF for M-PSK in Eq. (9-9), one can derive the CLF of $\pi/4$-QPSK as

$$\begin{aligned}\mathrm{CLF}_{\pi/4-4}\left(\theta_c\right) = Ce^{-N\gamma_s}\left(\frac{1}{2}\right)^N \prod_{n=1,3,5,\cdots}^{N-1}\sum_{q=0}^{1}\cosh\left[x_n\left(q;\theta_c\right)\right] \\ \times \prod_{n=0,2,4,\cdots}^{N-2}\sum_{q=0}^{1}\cosh\left[y_n\left(q;\theta_c\right)\right] \end{aligned} \tag{9-31}$$

where

$$\begin{aligned} x_n\left(q;\theta_c\right) &= \frac{\sqrt{2P_t}}{\sigma^2}\mathrm{Re}\left\{\tilde{r}_n e^{-j(\theta_c+(2q+1)\pi/4)}\right\} \\ y_n\left(q;\theta_c\right) &= \frac{\sqrt{2P_t}}{\sigma^2}\mathrm{Re}\left\{\tilde{r}_n e^{-j(\theta_c+(2q+1)\pi/4+\pi/4)}\right\} \\ &= \frac{\sqrt{2P_t}}{\sigma^2}\mathrm{Re}\left\{\tilde{r}_n e^{-j\pi/4}e^{-j(\theta_c+(2q+1)\pi/4)}\right\} \end{aligned} \tag{9-32}$$

Since from Eq. (9-9) the CLF for BPSK would be

$$\mathrm{CLF}_2\left(\theta_c\right) = Ce^{-N\gamma_s}\prod_{n=0}^{N-1}\cosh\left[x_n\left(0;\theta_c\right)\right] \tag{9-33}$$

with $x_n\left(0;\theta_c\right) = \left(\sqrt{2P_t}/\sigma^2\right)\mathrm{Re}\left\{\tilde{r}_n e^{-j\theta_c}\right\}$, then averaging over the uniformly distributed carrier phase, the LR becomes

$$\mathrm{LR} = \frac{\underset{\theta_c}{E}\left\{\mathrm{CLF}_2\left(\theta_c\right)\right\}}{\underset{\theta_c}{E}\left\{\mathrm{CLF}_{\pi/4-4}\left(\theta_c\right)\right\}} \tag{9-34}$$

which can be evaluated numerically by the same approaches previously discussed in Section 9.2.1.

9.8.2 ML Noncoherent Classifier Averaging over Carrier Phase, then Data

Averaging first over the uniformly distributed carrier phase and then over the data, the unconditional LFs become [also see Eq. (9-14)]

$$\mathrm{LF}_2 = \underset{q}{E}\left\{Ce^{-N\gamma_s}I_0\left(\frac{\sqrt{2P_t}}{\sigma^2}\left|\sum_{n=0}^{N-1}\tilde{r}_n e^{-j\theta_n}\right|\right)\right\},$$

$$\theta_1,\theta_2,\cdots,\theta_N \in (0,\pi)$$

$$\mathrm{LF}_{\pi/4-4} = \underset{q}{E}\left\{Ce^{-N\gamma_s}I_0\left(\frac{\sqrt{2P_t}}{\sigma^2}\left|\sum_{n=0}^{N-1}\tilde{r}_n e^{-j\theta_n}\right|\right)\right\}, \tag{9-35}$$

$$\theta_1,\theta_3,\cdots,\theta_{N-1} \in \left(\frac{\pi}{4},\frac{3\pi}{4},\frac{5\pi}{4},\frac{7\pi}{4}\right),$$

$$\theta_2,\theta_4,\cdots,\theta_N \in \left(0,\frac{\pi}{2},\pi,\frac{3\pi}{2}\right)$$

and the corresponding LR becomes

$$\mathrm{LR} = \frac{\mathrm{LF}_2}{\mathrm{LF}_{\pi/4-4}} \tag{9-36}$$

9.8.3 Suboptimum Classifiers

9.8.3.1 The GLRT. As was the case in Section 9.2.2, finding the GLRT for the $\pi/4$-QPSK/BPSK classifier relies on finding the ML estimates of carrier phase for the two modulations under consideration. Following the approach taken in Appendix 9-A, the solution for this estimate corresponding to $\pi/4$-QPSK modulation is derived in Appendix 9-B with the result

$$\hat{\theta}_{c,\pi/4-4} = \frac{1}{4}\arg\left(\sum_{n=1}^{N}\left(\tilde{r}_n e^{-j\frac{\pi}{4}I_n}\right)^4\right) \tag{9-37}$$

where I_n is the indicator variable defined by

$$I_n = \begin{cases} 0, & n \text{ odd} \\ 1, & n \text{ even} \end{cases} \tag{9-38}$$

The form of $\hat{\theta}_{c,\pi/4-4}$ is intuitively satisfying in that in the odd symbol intervals the contribution of the observable $\tilde{r}_n$ to the sum is the same as that for the QPSK ML phase estimate (i.e., $\tilde{r}_n^4$) whereas in the even symbol intervals (where the transmitted phase is shifted by $\pi/4$ rad), the observable $\tilde{r}_n$ is phase derotated by $\pi/4$ (multiplication by $e^{-j\pi/4}$) before making the same contribution to the sum.

In reality, the ML carrier phase estimate given above could be deduced immediately from the result for conventional QPSK by recognizing that $\pi/4$-QPSK can be modeled in complex baseband form as a QPSK modulator followed by multiplication by $e^{j(\pi/4)I_n}$. Thus, the corresponding ML receiver for such a modulation would be one that first undoes this alternate phase rotation, i.e., first multiplies the received signal plus noise samples by $e^{-j(\pi/4)I_n}$, and then follows that with a conventional QPSK ML receiver. (Note that multiplication by $e^{-j(\pi/4)I_n}$ does not change the statistical nature of the received noise samples.) Equivalently then, the "observables" inputs to the conventional QPSK portion of the receiver are given by $\tilde{r}_n e^{-j(\pi/4)I_n}$ and thus ML parameter estimates for $\pi/4$-QPSK are obtained from those for conventional QPSK by replacing $\tilde{r}_n$ by $\tilde{r}_n e^{-j(\pi/4)I_n}$.

For the binary case, the ML estimate remains as before, namely [see Eq. (9-16)],

$$\hat{\theta}_{c2} = \frac{1}{2} \arg \left(\sum_{n=0}^{N-1} \tilde{r}_n^2 \right) \tag{9-39}$$

9.8.3.2 The qLLR and nqLLR Metrics. Analogous to the derivation of the qLLR metric for the BPSK/QPSK classification, one can show that the *identical* metric is appropriate for the BPSK/π/4-QPSK classification, namely,

$$\text{qLLR} = \left| \sum_{n=0}^{N-1} \tilde{r}_n^2 \right| \tag{9-40}$$

and likewise for the nqLLR. Of course, the misclassification probability performance for the BPSK/π/4-QPSK classification will be different than that previously found for the BPSK/QPSK classification.

9.9 Noncoherent Classification of Offset Quadrature Modulations

Offset quadrature modulations are a class of modulations in which the inphase (I) and quadrature (Q) modulations are misaligned in time with respect to one another by one-half of a symbol. Examples of such modulations are offset quadrature phase-shift keying (OQPSK) [alternatively called staggered QPSK (SQPSK)] for which the I and Q pulse streams have identical *rectangular* pulse shapes and minimum-shift keying (MSK) for which the I and Q pulse streams have identical *half-sinusoidal* pulse shapes.[7] The customary reason for using an offset form of quadrature modulation is that it reduces the maximum fluctuation of the instantaneous amplitude of the modulation from 180 deg to 90 deg since the I and Q modulations cannot change polarity at the same time instant. This has an advantage on nonlinear transmission channels where the instantaneous fluctuation of the modulation amplitude is related to the regeneration of spectral side lobes of the modulation after bandpass filtering and nonlinear amplification—the smaller the instantaneous amplitude fluctuation, the smaller the side-lobe regeneration and vice versa. On an ideal linear AWGN channel, there is no theoretical advantage of using an offset modulation relative to a conventional one; in fact, the two have identical error probability performance.

In this section, LFs for offset quadrature modulations are derived for use in the same noncoherent modulation classification applications as treated in the

[7] MSK is inherently a continuous phase frequency-shift keyed (CPFSK) modulation but can be represented as a precoded offset quadrature modulation where the precoder takes the form of a differential decoder [17, Chapter 10].

previous sections. Again, both optimum and suboptimum versions of the LFs are considered, and classification examples are presented based on discriminating OQPSK from BPSK and MSK from QPSK.

9.9.1 Channel Model and Conditional-Likelihood Function

For transmission of an offset quadrature modulation over an AWGN channel, the received signal can be written as

$$r(t) = \sqrt{P_t}\left(\sum_{n=-\infty}^{\infty} a_n p(t-nT)\right)\cos(\omega_c t+\theta) - \sqrt{P_t}\left(\sum_{n=-\infty}^{\infty} b_n p(t-(n+1/2)T)\right)\sin(\omega_c t+\theta)+n(t) \quad (9\text{-}41)$$

where $\{a_n\},\{b_n\}$ are independent, identically distributed (iid) binary (± 1) sequences, and as before, T is the symbol time, $p(t)$ is a unit power pulse shape of duration T seconds, P_t is the passband signal power, and $n(t)$ is an AWGN process with single-sided power spectral density N_0 W/Hz. Based on the above AWGN model, then for an observation of N data (symbol) intervals the CLF is given by

$$\begin{aligned} &p\big(r(t)|\{a_n\},\{b_n\},p(t),\theta_c\big) \\ &= \frac{1}{\sqrt{\pi N_0}}\exp\left\{-\frac{1}{N_0}\int_0^{NT}\left[r(t)-\sqrt{P_t}\left(\sum_{n=-\infty}^{\infty} a_n p(t-nT)\right)\right.\right. \\ &\quad \left.\left.\times\cos(\omega_c t+\theta_c)+\sqrt{P_t}\left(\sum_{n=-\infty}^{\infty} b_n p(t-(n+1/2)T)\right)\sin(\omega_c t+\theta_c)\right]^2 dt\right\} \\ &= C\exp(-N\gamma)\exp\left\{\frac{2\sqrt{P_t}}{N_0}\sum_{n=0}^{N-1} a_n\int_{nT}^{(n+1)T} r(t)\,p(t-nT)\cos\omega_c t dt\right\} \\ &\quad\times\exp\left\{-\frac{2\sqrt{P_t}}{N_0}\sum_{n=0}^{N-1} b_n\int_{(n+1/2)T}^{(n+3/2)T} r(t)\,p(t-(n+1/2)T)\sin\omega_c t dt\right\} \end{aligned} \quad (9\text{-}42)$$

where as before $\gamma_s \triangleq P_t T/N_0 = E_s/N_0$ is the symbol energy-to-noise ratio and C is a constant that will be independent of the classification to be made. Transforming the received signal of Eq. (9-41) to complex baseband via $r(t) = \mathrm{Re}\left\{\tilde{r}(t)\, e^{j\omega_c t}\right\}$ and defining the I and Q complex baseband observables (matched filter outputs) by

$$\begin{aligned} \tilde{r}_{In} &= \frac{1}{T}\int_{nT}^{(n+1)T} \tilde{r}(t)\, p(t-nT)\, dt \\ \tilde{r}_{Qn} &= \frac{1}{T}\int_{(n+1/2)T}^{(n+3/2)T} \tilde{r}(t)\, p\big(t-(n+1/2)\,T\big) dt \end{aligned} \tag{9-43}$$

then, letting $\sigma^2 = N_0/T$ denote the variance of these outputs, the CLF can be written in the equivalent form

$$\begin{aligned} p(\tilde{r}\,|\{a_n\},\{b_n\},\theta_c) &= C\exp(-N\gamma)\exp\left\{\frac{\sqrt{P_t}}{\sigma^2}\sum_{n=0}^{N-1} a_n \mathrm{Re}\left\{\tilde{r}_{In}e^{-j\theta_c}\right\}\right\} \\ &\quad\times\exp\left\{\frac{\sqrt{P_t}}{\sigma^2}\sum_{n=0}^{N-1} b_n \mathrm{Im}\left\{\tilde{r}_{Qn}e^{-j\theta_c}\right\}\right\} \\ &= C\exp\left(-N\gamma_s + \mathrm{Re}\left\{\frac{\sqrt{P_t}}{\sigma^2}e^{-j\theta_c}\sum_{n=0}^{N-1} a_n\tilde{r}_{In}\right\}\right. \\ &\quad\left. + \mathrm{Im}\left\{\frac{\sqrt{P_t}}{\sigma^2}e^{-j\theta_c}\sum_{n=0}^{N-1} b_n\tilde{r}_{Qn}\right\}\right) \end{aligned} \tag{9-44}$$

Note that for conventional (non-offset) QPSK we would have $\tilde{r}_{In} = \tilde{r}_{Qn} = \tilde{r}_n$, and thus letting $e^{j\theta_n} = (a_n + jb_n)/\sqrt{2}$, the CLF of Eq. (9-44) would become

$$p(\tilde{r}\,|\{a_n\},\{b_n\},\theta_c) = C\exp\left(-N\gamma_s + \mathrm{Re}\left\{\frac{\sqrt{2P_t}}{\sigma^2}e^{-j\theta_c}\sum_{n=0}^{N-1}\tilde{r}_n e^{-j\theta_n}\right\}\right) \tag{9-45}$$

which is consistent with Eq. (9-7).

Next we average over the I and Q data streams. Before doing so, however, we first manipulate the form of Eq. (9-44) as follows:

$$p\left(\tilde{r}\,|\{a_n\},\{b_n\},\theta_c\right) = C\exp\left(-N\gamma_s\right)\prod_{n=0}^{N-1}\exp\left[\mathrm{Re}\left\{\frac{\sqrt{P_t}}{\sigma^2}e^{-j\theta_c}a_n\tilde{r}_{In}\right\}\right]$$

$$\times\prod_{n=0}^{N-1}\exp\left[\mathrm{Im}\left\{\frac{\sqrt{P_t}}{\sigma^2}e^{-j\theta_c}b_n\tilde{r}_{Qn}\right\}\right]$$

$$= C\exp\left(-N\gamma_s\right)\exp\sum_{n=0}^{N-1}\ln\left(\exp\left[\mathrm{Re}\left\{\frac{\sqrt{P_t}}{\sigma^2}e^{-j\theta_c}a_n\tilde{r}_{In}\right\}\right]\right)$$

$$\times\exp\sum_{n=0}^{N-1}\ln\left(\exp\left[\mathrm{Im}\left\{\frac{\sqrt{P_t}}{\sigma^2}e^{-j\theta_c}b_n\tilde{r}_{Qn}\right\}\right]\right) \tag{9-46}$$

Now averaging over $\{a_n\}$ and $\{b_n\}$ gives

$$p\left(\tilde{r}\,|\theta_c\right) = C\exp\left(-N\gamma_s\right)\exp\sum_{n=0}^{N-1}\ln\cosh\left(\frac{\sqrt{P_t}}{\sigma^2}\mathrm{Re}\left\{\tilde{r}_{In}e^{-j\theta_c}\right\}\right)$$

$$\times\exp\sum_{n=0}^{N-1}\ln\cosh\left(\frac{\sqrt{P_t}}{\sigma^2}\mathrm{Im}\left\{\tilde{r}_{Qn}e^{-j\theta_c}\right\}\right)$$

$$= C\exp\left\{-N\gamma_s+\sum_{n=0}^{N-1}\left[\ln\cosh x_{In}\left(\theta_c\right)+\ln\cosh x_{Qn}\left(\theta_c\right)\right]\right\} \tag{9-47}$$

where

$$\begin{aligned} x_{In}\left(\theta_c\right) &\triangleq \frac{\sqrt{P_t}}{\sigma^2}\mathrm{Re}\left\{\tilde{r}_{In}e^{-j\theta_c}\right\} \\ x_{Qn}\left(\theta_c\right) &\triangleq \frac{\sqrt{P_t}}{\sigma^2}\mathrm{Im}\left\{\tilde{r}_{Qn}e^{-j\theta_c}\right\} \end{aligned} \tag{9-48}$$

Alternatively, defining

$$x_n(0;\theta_c) \triangleq x_{In}(\theta_c) - x_{Qn}(\theta_c)$$
$$x_n(1;\theta_c) \triangleq x_{In}(\theta_c) + x_{Qn}(\theta_c) \tag{9-49}$$

and using the trigonometric identity

$$\ln\left\{\frac{1}{2}\left[\cosh(X+Y)+\cosh(X-Y)\right]\right\} = \ln\cosh X + \ln\cosh Y \tag{9-50}$$

the CLF of Eq. (9-47) can be manipulated into the compact form

$$p(\tilde{r}\,|\theta_c) = C\exp\left\{-N\gamma_s + \sum_{n=0}^{N-1}\ln\left(\frac{1}{2}\sum_{q=0}^{1}\cosh x_n(q;\theta_c)\right)\right\}$$
$$= C\exp(-N\gamma_s)\left(\frac{1}{2}\right)^N\prod_{n=0}^{N-1}\sum_{q=0}^{1}\cosh x_n(q;\theta_c) \triangleq \mathrm{CLF}_4'(\theta_c) \tag{9-51}$$

9.9.2 Classification of OQPSK versus BPSK

As an example of noncoherent classification involving an offset quadrature modulation, we consider the case of classifying OQPSK versus BPSK. As noted in previous sections, the LFs that form the LR on which the classification is based can be arrived at in two ways as follows.

9.9.2.1 Averaging the LFs over the Data Sequences, then over the Carrier Phase. For OQPSK, the CLF obtained by averaging the LF over the I and Q data sequences has already been determined in Eq. (9-32). The corresponding result for BPSK was previously determined as [see Eq. (9-8)]

$$p(\tilde{r}\,|\theta_c) = C\exp(-N\gamma_s)\prod_{n=0}^{N-1}\cosh\left(\sqrt{2}x_n(0,\theta_c)\right) \triangleq \mathrm{CLF}_2(\theta_c) \tag{9-52}$$

where for BPSK, $x_{Qn}(\theta_c) = 0$ and, hence, $x_n(0;\theta_c) = x_{In}(\theta_c)$. Thus, averaging Eqs. (9-51) and (9-52) over θ_c, assumed to be uniformly distributed over the interval $(0, 2\pi)$, then the LR is computed as

$$\mathrm{LR} = \frac{\mathrm{LF}_4'}{\mathrm{LF}_2} = \frac{E_{\theta_c}\left\{\mathrm{CLF}_4'\left(\theta_c\right)\right\}}{E_{\theta_c}\left\{\mathrm{CLF}_2\left(\theta_c\right)\right\}} \tag{9-53}$$

Computing the LR from Eq. (9-53) must be performed numerically and is computationally intensive. Furthermore, in order to compute the LFs themselves, the parameters P_t and σ^2 must be evaluated (to allow computation of the $x_n\left(q;\theta_c\right)$'s).

9.9.2.2 Averaging the LFs over the Carrier Phase, then over the Data Sequences. Suppose instead we first average the CLF of Eq. (9-44) over the carrier phase. To see how to accomplish this, we rewrite Eq. (9-44) as follows:

$$\begin{aligned}
&p\big(\tilde{r}|\{a_n\},\{b_n\},\theta_c\big) \\
&\quad = C\exp\left(-N\gamma_s + \mathrm{Re}\left\{\frac{\sqrt{P_t}}{\sigma^2}\left|\sum_{n=0}^{N-1} a_n\tilde{r}_{In}\right| e^{-j\left[\theta_c - \arg\left(\sum_{n=0}^{N-1} a_n\tilde{r}_{In}\right)\right]}\right\}\right. \\
&\qquad \left. +\mathrm{Im}\left\{\frac{\sqrt{P_t}}{\sigma^2}\left|\sum_{n=0}^{N-1} b_n\tilde{r}_{Qn}\right| e^{-j\left[\theta_c - \arg\left(\sum_{n=0}^{N-1} b_n\tilde{r}_{Qn}\right)\right]}\right\}\right) \\
&\quad = C\exp\left(-N\gamma_s + \frac{\sqrt{P_t}}{\sigma^2}\left|\sum_{n=0}^{N-1} a_n\tilde{r}_{In}\right|\cos\left[\theta_c - \arg\left(\sum_{n=0}^{N-1} a_n\tilde{r}_{In}\right)\right]\right. \\
&\qquad \left. -\frac{\sqrt{P_t}}{\sigma^2}\left|\sum_{n=0}^{N-1} b_n\tilde{r}_{Qn}\right|\sin\left[\theta_c - \arg\left(\sum_{n=0}^{N-1} b_n\tilde{r}_{Qn}\right)\right]\right)
\end{aligned} \tag{9-54}$$

Applying the trigonometric identity

$$\begin{aligned}
&|X_1|\cos(\theta_c - \phi_1) - |X_2|\sin(\theta_c - \phi_2) \\
&\quad = \sqrt{|X_1|^2 + |X_2|^2 + 2\,|X_1|\,|X_2|\sin\left(\phi_2 - \phi_1\right)} \times \cos\left(\theta_c - \eta\right) \\
&\quad = |X_1 - jX_2|\cos\left(\theta_c - \eta\right); \\
&\eta = \tan^{-1}\frac{|X_1|\sin\phi_1 - |X_2|\cos\phi_2}{|X_1|\cos\phi_1 + |X_2|\sin\phi_2}
\end{aligned} \tag{9-55}$$

to Eq. (9-54), we get

$$p\big(\tilde{r}|\{a_n\},\{b_n\},\theta_c\big)$$

$$= C\exp\left\{-N\gamma_s + \frac{\sqrt{P_t}}{\sigma^2}\left|\sum_{n=0}^{N-1}\left(a_n\tilde{r}_{In} - jb_n\tilde{r}_{Qn}\right)\right|\cos\left(\theta_c - \eta\right)\right\} \quad (9\text{-}56)$$

which when averaged over the uniform distribution of θ_c gives the desired result:

$$p\big(\tilde{r}\,|\{a_n\}\,,\{b_n\}\,\big) = C\exp\left(-N\gamma_s\right)I_0\left(\frac{\sqrt{P_t}}{\sigma^2}\left|\sum_{n=0}^{N-1}\left(a_n\tilde{r}_{In} - jb_n\tilde{r}_{Qn}\right)\right|\right) \quad (9\text{-}57)$$

To check the consistency of this result with that for conventional QPSK, we proceed as follows. The previously derived result for the CLF of M-PSK is given by Eq. (9-8). For QPSK, we would have $e^{-j\theta_n} = \left(a_n - jb_n\right)/\sqrt{2}$, where a_n and b_n are as defined in Section 9.9.1. Thus,

$$\left|\sum_{n=0}^{N-1}\tilde{r}_n e^{-j\theta_n}\right| = \frac{1}{\sqrt{2}}\left|\sum_{n=0}^{N-1}\tilde{r}_n\left(a_n - jb_n\right)\right| \quad (9\text{-}58)$$

which when substituted in Eq. (9-8) agrees with Eq. (9-57) when $\tilde{r}_{In} = \tilde{r}_{Qn} = \tilde{r}_n$, as would be the case for conventional QPSK. Finally, the LF is obtained by averaging Eq. (9-57) over the data sequences $\{a_n\}$ and $\{b_n\}$, i.e.,

$$\mathrm{LF}_4' = E_{\{a_n\},\{b_n\}}\left\{C\exp\left(-N\gamma_s\right)I_0\left(\frac{\sqrt{P_t}}{\sigma^2}\left|\sum_{n=0}^{N-1}\left(a_n\tilde{r}_{In} - jb_n\tilde{r}_{Qn}\right)\right|\right)\right\} \quad (9\text{-}59)$$

which again must be done numerically and is feasible for small values of N.

9.9.3 Suboptimum (Simpler to Implement) Classifiers

In order to simplify the implementation of the ML classifiers, one must resort to approximations of the nonlinearities involved in their definitions in much the same way as was done for the conventional (non-offset) modulations. We start with the CLF averaged first over the data and then the carrier phase. Ignoring the factor $C\exp\left(-N\gamma_s\right)$ since in an LR test between two hypotheses it will cancel

out, then taking the natural logarithm of Eq. (9-51) gives the log-likelihood function (LLF)

$$\mathrm{LLF}_4' = \ln E_{\theta_c}\left\{\exp\left[\sum_{n=0}^{N-1}\ln\left(\frac{1}{2}\sum_{q=0}^{1}\cosh x_n\left(q;\theta_c\right)\right)\right]\right\} \tag{9-60}$$

Applying the approximations

$$\begin{aligned}\cosh x &\cong 1+\frac{x^2}{2}\\ \ln\left(1+x\right) &\cong x\end{aligned} \tag{9-61}$$

gives

$$\mathrm{LLF}_4' = \ln E_{\theta_c}\left\{\exp\left[\sum_{n=0}^{N-1}\frac{1}{2}\sum_{q=0}^{1}\frac{x_n^2\left(q;\theta_c\right)}{2}\right]\right\} \tag{9-62}$$

From Eq. (9-49), we find that

$$\begin{aligned}\sum_{q=0}^{1}\frac{x_n^2\left(q;\theta_c\right)}{2} &= \frac{1}{2}\left[\left(x_{In}\left(\theta_c\right)+x_{Qn}\left(\theta_c\right)\right)^2+\left(x_{In}\left(\theta_c\right)-x_{Qn}\left(\theta_c\right)\right)^2\right]\\ &= x_{In}^2\left(\theta_c\right)+x_{Qn}^2\left(\theta_c\right)\\ &= \frac{P_t}{\sigma^4}\left[\left(\mathrm{Re}\left\{\tilde{r}_{In}e^{-j\theta_c}\right\}\right)^2+\left(\mathrm{Im}\left\{\tilde{r}_{Qn}e^{-j\theta_c}\right\}\right)^2\right]\end{aligned} \tag{9-63}$$

Further, using the relations

$$\begin{aligned}\left(\mathrm{Re}\left\{\tilde{z}\right\}\right)^2 &= \frac{1}{2}\left|\tilde{z}\right|^2+\frac{1}{2}\mathrm{Re}\left\{\tilde{z}^2\right\}\\ \left(\mathrm{Im}\left\{\tilde{z}\right\}\right)^2 &= \frac{1}{2}\left|\tilde{z}\right|^2-\frac{1}{2}\mathrm{Re}\left\{\tilde{z}^2\right\}\end{aligned} \tag{9-64}$$

we obtain after some simplification

$$\sum_{q=0}^{1} \frac{x_n^2(q;\theta_c)}{2} = \frac{P_t}{2\sigma^4}\left[|\tilde{r}_{In}|^2 + |\tilde{r}_{Qn}|^2 + \left(\mathrm{Re}\left\{\tilde{r}_{In}^2 e^{-j2\theta_c}\right\}\right)^2 - \left(\mathrm{Re}\left\{\tilde{r}_{Qn}^2 e^{-j2\theta_c}\right\}\right)^2\right] \tag{9-65}$$

which when substituted in Eq. (9-62) gives

$$\begin{aligned}\mathrm{LLF}_4' = \ln E_{\theta_c}\Bigg\{ \exp\Bigg[\sum_{n=0}^{N-1} \frac{P_t}{4\sigma^4}\left[|\tilde{r}_{In}|^2 + |\tilde{r}_{Qn}|^2 + \left(\mathrm{Re}\left\{\tilde{r}_{In}^2 e^{-j2\theta_c}\right\}\right)^2\right] \\ - \left(\mathrm{Re}\left\{\tilde{r}_{Qn}^2 e^{-j2\theta_c}\right\}\right)^2\Bigg]\Bigg\}\end{aligned} \tag{9-66}$$

Noting that the first two terms of the summation in Eq. (9-66) do not depend on θ_c, then the LLF can be simplified to

$$\begin{aligned}\mathrm{LLF}_4' = \frac{P_t}{4\sigma^4}\sum_{n=0}^{N-1}\left[|\tilde{r}_{In}|^2 + |\tilde{r}_{Qn}|^2\right] \\ + \ln E_{\theta_c}\Bigg\{\exp\Bigg[\frac{P_t}{4\sigma^4}\mathrm{Re}\Bigg\{\left|\sum_{n=0}^{N-1}\tilde{r}_{In}^2\right| e^{-j\left(2\theta_c - \arg\left\{\sum_{n=0}^{N-1}\tilde{r}_{In}^2\right\}\right)} \\ - \left|\sum_{n=0}^{N-1}\tilde{r}_{Qn}^2\right| e^{-j\left(2\theta_c - \arg\left\{\sum_{n=0}^{N-1}\tilde{r}_{Qn}^2\right\}\right)}\Bigg\}\Bigg]\Bigg\}\end{aligned} \tag{9-67}$$

Using a relation analogous to Eq. (9-55), namely,

$$\begin{aligned}&|X_1|\cos(2\theta_c - \phi_1) - |X_2|\cos(2\theta_c - \phi_2) \\ &\quad = \sqrt{|X_1|^2 + |X_2|^2 - 2|X_1||X_2|\cos(\phi_2 - \phi_1)} \times \cos(2\theta_c - \eta) \\ &\quad = |X_1 - X_2|\cos(2\theta_c - \eta);\end{aligned} \tag{9-68}$$

$$\eta = \tan^{-1}\frac{|X_1|\sin\phi_1 - |X_2|\sin\phi_2}{|X_1|\cos\phi_1 - |X_2|\cos\phi_2}$$

the approximate LLF of Eq. (9-67) finally becomes

$$\mathrm{LLF}_4' = \frac{P_t}{4\sigma^4}\sum_{n=0}^{N-1}\left[|\tilde{r}_{In}|^2 + |\tilde{r}_{Qn}|^2\right] + \ln I_0\left(\frac{P_t}{4\sigma^4}\left|\sum_{n=0}^{N-1}\left(\tilde{r}_{In}^2 - \tilde{r}_{Qn}^2\right)\right|\right) \tag{9-69}$$

For BPSK (ignoring the same $C\exp(-N\gamma_s)$ factor), we obtain from Eq. (9-52) the LLF

$$\mathrm{LLF}_2 = \ln E_{\theta_c}\left\{\sum_{n=0}^{N-1}\ln\cosh\left(\sqrt{2}x_n(0,\theta_c)\right)\right\} \tag{9-70}$$

where again $x_n(0;\theta_c) = x_{In}(\theta_c)$ since in this case $x_{Qn}(\theta_c) = 0$. Making the same nonlinearity approximations as in Eq. (9-61), we obtain the approximate LLF

$$\begin{aligned}\mathrm{LLF}_2 &= \ln E_{\theta_c}\left\{\exp\left[\sum_{n=0}^{N-1}x_n^2(0;\theta_c)\right]\right\}\\ &= \ln E_{\theta_c}\left\{\exp\left[\frac{P_t}{\sigma^4}\sum_{n=0}^{N-1}\left(\mathrm{Re}\left\{\tilde{r}_{In}^2 e^{-j2\theta_c}\right\}\right)^2\right]\right\}\end{aligned} \tag{9-71}$$

Using Eq. (9-64), we again obtain after some simplification

$$\mathrm{LLF}_2 = \frac{P_t}{2\sigma^4}\sum_{n=0}^{N-1}|\tilde{r}_{In}|^2 + \ln I_0\left(\frac{P_t}{2\sigma^4}\left|\sum_{n=0}^{N-1}\tilde{r}_{In}^2\right|\right) \tag{9-72}$$

Finally, then qLLR$'$ is obtained as the difference of Eqs. (9-69) and (9-72), namely,

$$\begin{aligned}\mathrm{qLLR}' \triangleq \mathrm{LLF}_2 - \mathrm{LLF}_4' &= \frac{P_t}{4\sigma^4}\sum_{n=0}^{N-1}\left[|\tilde{r}_{In}|^2 - |\tilde{r}_{Qn}|^2\right] + \ln I_0\left(\frac{P_t}{2\sigma^4}\left|\sum_{n=0}^{N-1}\tilde{r}_{In}^2\right|\right)\\ &\quad - \ln I_0\left(\frac{P_t}{4\sigma^4}\left|\sum_{n=0}^{N-1}\left(\tilde{r}_{In}^2 - \tilde{r}_{Qn}^2\right)\right|\right)\end{aligned} \tag{9-73}$$

Note once again that for a classification between BPSK and conventional QPSK where $\tilde{r}_{In} = \tilde{r}_{Qn} = \tilde{r}_n$, Eq. (9-73) would simplify to

$$\text{qLLR}' \triangleq \text{LLF}_2 - \text{LLF}_4' = \ln I_0\left(\frac{P_t}{2\sigma^4}\left|\sum_{n=0}^{N-1}\tilde{r}_n^2\right|\right) = \ln I_0\left(\frac{\gamma}{2}\left|\sum_{n=0}^{N-1}\left(\frac{\tilde{r}_n}{\sigma}\right)^2\right|\right) \tag{9-74}$$

which is in agreement with Eq. (21) of [10].

To get to the final simplification, we now apply the approximation[8] $\ln I_0(x) \cong x \ \ (x \gg 1)$, resulting in

$$\text{qLLR}' = \frac{P_t}{4\sigma^4}\left[\sum_{n=0}^{N-1}\left[|\tilde{r}_{In}|^2 - |\tilde{r}_{Qn}|^2\right] + 2\left|\sum_{n=0}^{N-1}\tilde{r}_{In}^2\right| - \left|\sum_{n=0}^{N-1}\left(\tilde{r}_{In}^2 - \tilde{r}_{Qn}^2\right)\right|\right] \tag{9-75}$$

Since the true LR should be compared to unity threshold, the true LLR should be compared to a zero threshold. However, as was previously discussed in Section 9.3, it is not necessarily true that, due to the nature of the approximations, the qLLR should also be compared to a zero threshold. Rather, as in the conventional QPSK/BPSK classification case, the best threshold (to minimize the probability of misclassification) should be determined by numerical means. As such, one could equivalently use

$$\text{qLLR}' = \sum_{n=0}^{N-1}\left[|\tilde{r}_{In}|^2 - |\tilde{r}_{Qn}|^2\right] + 2\left|\sum_{n=0}^{N-1}\tilde{r}_{In}^2\right| - \left|\sum_{n=0}^{N-1}\left(\tilde{r}_{In}^2 - \tilde{r}_{Qn}^2\right)\right| \tag{9-76}$$

which by itself would be independent of P_t and σ^2 and adjust the threshold accordingly based on knowledge of these parameters for optimum misclassification probability performance. Alternatively, as was noted for the non-offset modulation classification case in Section 9.2.2, one could avoid this threshold dependence on the signal and noise parameters by using a normalized qLLR, namely,

---

[8] Note that the argument of the Bessel function in Eq. (9-72) is large, corresponding to the post-detection SNR, i.e., after the N symbols have been accumulated. This differs from the predetection SNR, γ_s, which can be assumed to be small in noisy environments.

$$\text{nqLLR}' = \frac{\sum_{n=0}^{N-1}\left[|\tilde{r}_{In}|^2 - |\tilde{r}_{Qn}|^2\right] + 2\left|\sum_{n=0}^{N-1}\tilde{r}_{In}^2\right| - \left|\sum_{n=0}^{N-1}\left(\tilde{r}_{In}^2 - \tilde{r}_{Qn}^2\right)\right|}{\sum_{n=0}^{N-1}\left[|\tilde{r}_{In}|^2 + |\tilde{r}_{Qn}|^2\right]} \tag{9-77}$$

that like the qLLR of Eq. (9-76) is itself independent of P_t and σ^2, but unlike the qLLR is invariant to scale changes in $\tilde{r}$ and is fairly insensitive to variations of P_t or σ^2 as well.

Finally, we conclude this section by noting that the previously discussed coarse integral approximation method applied to the individual CLFs in the LR (see Section 9.2.2) and likewise the GLRT method that replaces the integration of the CLF over θ_c by its evaluation at the ML value of θ_c can also be applied here in the offset modulation classification case.

9.9.4 Classification of MSK versus QPSK

Another example of classification of an offset quadrature modulation and a conventional modulation might be MSK versus QPSK. As previously noted, MSK has an offset quadrature representation in the form of a precoded OQPSK with identical half-sinusoidal pulse shapes on the I and Q channels. In order to maintain the power of the transmitted signal as P_t for both modulations, the mathematical description of the received signal corresponding to transmission of MSK is in the form of Eq. (9-41), now with

$$p(t) = \sqrt{2}\sin\frac{\pi t}{T}, \quad 0 \le t \le T \tag{9-78}$$

Thus, the observables for MSK are

$$\begin{aligned}
\tilde{r}_{In} &= \sqrt{2}\,\frac{1}{T}\int_{nT}^{(n+1)T}\tilde{r}(t)\sin\frac{\pi(t-nT)}{T}dt \\
\tilde{r}_{Qn} &= \sqrt{2}\,\frac{1}{T}\int_{(n+1/2)T}^{(n+3/2)T}\tilde{r}(t)\sin\frac{\pi\left(t-(n+1/2)\,T\right)}{T}dt \\
&= -\sqrt{2}\,\frac{1}{T}\int_{(n+1/2)T}^{(n+3/2)T}\tilde{r}(t)\cos\frac{\pi\left(t-nT\right)}{T}dt
\end{aligned} \tag{9-79}$$

whereas the observables for QPSK are

$$\tilde{r}_{In} = \tilde{r}_{Qn} = \frac{1}{T}\int_{nT}^{(n+1)T} \tilde{r}\,(t)\,dt \tag{9-80}$$

Furthermore, from Eq. (9-48), the conditional variables needed for the CLF of MSK are

$$\begin{aligned} x_{In}\,(\theta_c) &\triangleq \frac{\sqrt{P_t}}{\sigma^2}\,\mathrm{Re}\left\{\tilde{r}_{In}e^{-j\theta_c}\right\} \\ &= \frac{\sqrt{P_t}}{\sigma^2}\,\mathrm{Re}\left\{e^{-j\theta_c}\sqrt{2}\frac{1}{T}\int_{nT}^{(n+1)T}\tilde{r}\,(t)\sin\frac{\pi\,(t-nT)}{T}dt\right\} \\ x_{Qn}\,(\theta_c) &\triangleq \frac{\sqrt{P_t}}{\sigma^2}\,\mathrm{Im}\left\{\tilde{r}_{Qn}e^{-j\theta_c}\right\} \\ &= -\frac{\sqrt{P_t}}{\sigma^2}\,\mathrm{Im}\left\{e^{-j\theta_c}\sqrt{2}\frac{1}{T}\int_{(n+1/2)T}^{(n+3/2)T}\tilde{r}\,(t)\cos\frac{\pi\,(t-nT)}{T}dt\right\} \end{aligned} \tag{9-81}$$

whereas those needed for the CLF of QPSK are

$$\begin{aligned} x_{In}\,(\theta_c) &\triangleq \frac{\sqrt{P_t}}{\sigma^2}\,\mathrm{Re}\left\{\tilde{r}_{In}e^{-j\theta_c}\right\} = \frac{\sqrt{P_t}}{\sigma^2}\,\mathrm{Re}\left\{e^{-j\theta_c}\frac{1}{T}\int_{nT}^{(n+1)T}\tilde{r}\,(t)\,dt\right\} \\ x_{Qn}\,(\theta_c) &\triangleq \frac{\sqrt{P_t}}{\sigma^2}\,\mathrm{Im}\left\{\tilde{r}_{In}e^{-j\theta_c}\right\} = \frac{\sqrt{P_t}}{\sigma^2}\,\mathrm{Im}\left\{e^{-j\theta_c}\frac{1}{T}\int_{nT}^{(n+1)T}\tilde{r}\,(t)\,dt\right\} \end{aligned} \tag{9-82}$$

Since for QPSK the CLF, namely, $\mathrm{CLF}_4\,(\theta_c)$, also has the form of Eq. (9-51), then a classification of MSK versus QPSK would be based on the LR

$$\mathrm{LR} = \frac{\mathrm{LF}_4'}{\mathrm{LF}_4} = \frac{E_{\theta_c}\left\{\mathrm{CLF}_4'\,(\theta_c)\right\}}{E_{\theta_c}\left\{\mathrm{CLF}_4\,(\theta_c)\right\}} \tag{9-83}$$

where the $x_n\,(0;\theta_c)$ and $x_n\,(1;\theta_c)$ terms in Eq. (9-49) that define the CLFs in the numerator and denominator are appropriately expressed in terms of Eqs. (9-81) and (9-82), respectively.

9.10 Modulation Classification in the Presence of Residual Carrier Frequency Offset

Thus far in our discussions of modulation classification, the word "noncoherent" was used to mean that the carrier phase was completely unknown, i.e., uniformly distributed in the interval $[0, 2\pi)$, but at the same time the carrier frequency was assumed to be known exactly. Here we discuss the impact on modulation classification of imperfect knowledge of the carrier frequency, i.e., the presence of a fixed residual carrier frequency offset that may exist after frequency correction. In particular, we shall point out the degrading effect of this frequency error on the behavior of the previously derived modulation classifiers and then propose an ad hoc modification of the nqLLR to cope with the problem.

When a residual radian frequency error $\Delta\omega$ is present, the received signal of Eq. (9-1) is modified to

$$\tilde{r}(t) = \sqrt{2P_t} \sum_{n=-\infty}^{\infty} e^{j(\theta_n + \theta_c + \Delta\omega t)} p(t - nT - \varepsilon T) + \tilde{n}(t) \tag{9-84}$$

or, equivalently, under the assumption of perfect symbol timing, the observables of Eq. (9-3) become

$$\tilde{r}_n = \sqrt{2P_t} \left(\frac{\sin \dfrac{\Delta\omega T}{2}}{\dfrac{\Delta\omega T}{2}} \right) e^{j(\theta_n + \theta_c + n\Delta\omega T)} + \tilde{n}_n, \quad n = 0, \cdots, N-1 \tag{9-85}$$

Based on Eq. (9-85), we see that, aside from a $\sin x/x$ amplitude factor, in effect the constellations on which the likelihood function is based are discretely rotating by $\Delta\omega T$ rad during each symbol interval of the N-symbol observation time. So for example, if one attempts to use the nqLLR modulation classifier of Eq. (9-18) on the received signal in Eq. (9-84), the term $\tilde{r}_n^M$ in the numerator of Eq. (9-18) still removes the M-PSK modulation; however, its signal component now contains the phase factor $e^{jMn\Delta\omega T}$, whose argument changes linearly throughout the summation on n. Thus, even in the absence of noise, one is no longer summing a set of complex observables that are aligned in phase but rather summing a set of complex observables whose phase is uniformly rotating (by increments of $M\Delta\omega T$ rad) around the circle. In fact, if the residual frequency error-symbol time product is such that $M\Delta\omega T = 2\pi/N$, then in so far as the numerator of Eq. (9-18) is concerned, the constellation will have moved through a complete revolution during the observation, thereby confusing the classification of M-PSK versus M'-PSK.

To resolve this dilemma, we need to modify the classifier in such a way as to cancel out the effect of the frequency error in the terms being summed in the numerator of the LLR so that once again they are aligned in phase. An ad hoc solution can be obtained by replacing $\tilde{r}_n^M$ with $\tilde{r}_n^M \left(\tilde{r}_{n-1}^*\right)^M$. Since the frequency error has no effect on the terms in the denominator summation of Eq. (9-18), to maintain the needed normalization for independence of the metric on P_t and σ^2, we propose replacing $|\tilde{r}_n^M|$ by $|\tilde{r}_n^M \left(\tilde{r}_{n-1}^*\right)^M|$. Thus, in the presence of residual frequency error, the modification of Eq. (9-18) would become

$$\text{nqLLR} = \frac{\left|\sum_{n=1}^{N-1} \tilde{r}_n^M \left(\tilde{r}_{n-1}^*\right)^M\right|}{\sum_{n=1}^{N-1} \left|\tilde{r}_n^M \left(\tilde{r}_{n-1}^*\right)^M\right|} \tag{9-86}$$

Since, compared with Eq. (9-18), this metric applied to BPSK/QPSK classification now involves fourth-order (as opposed to second-order) signal $\times$ noise and noise $\times$ noise products, one anticipates a degradation in performance even in the case of zero residual frequency error. To compensate for this additional degradation, one would need to increase the length of the observable, i.e., increase N. It should also be pointed out that the optimized decision thresholds computed as in Section 9.3 will be different for the nqLLR of Eq. (9-86) than those for the nqLLR of Eq. (9-18). However, the procedure needed to perform the empirical threshold optimization would still follow the same steps as those discussed in Section 9.3.2.

References

[1] H. V. Poor, *An Introduction to Signal Detection and Estimation*, New York: Springer-Verlag, 1988.

[2] C. W. Therrien, *Discrete Random Signal and Statistical Signal Processing*, Englewood Cliffs, New Jersey: Prentice Hall, 1992.

[3] A. Polydoros and K. Kim, "On the Detection and Classification of Quadrature Digital Modulations in Broad-Band Noise," *IEEE Transactions on Communications*, vol. 38, pp. 1199–1211, August 1990.

[4] J. H. Yuen, *Deep Space Telecommunications Systems Engineering*, New York: Plenum Press, 1983.

[5] Y. Yang and S. Soliman, "A Suboptimal Algorithm for Modulation Classification," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 33, no. 1, pp. 38–45, 1997.

[6] W. H. Press, W. T. Vetterling, S. A. Teukolsky, and B. P. Flannery, *Numerical Recipes in C*, New York: Cambridge University Press, 1992.

[7] B. Beidas and C. Weber, "Asynchronous Classification of MFSK Signals Using the Higher Order Correlation Domain," *IEEE Transactions on Communications*, vol. 46, no. 4, pp. 480–493, 1998.

[8] U. Mengali and A. N. D'Andrea, *Synchronization Techniques for Digital Receivers*, New York: Plenum, 1997.

[9] N. E. Lay and A. Polydoros, "Modulation Classification of Signals in Unknown ISI Environments," *Military Communications Conference, 1995. MILCOM'95, Conference record, IEEE*, vol. 1, San Diego, California, pp. 170–174, November 1995.

[10] C.-Y. Huang and A. Polydoros, "Likelihood Methods for MPSK Modulation Classification," *IEEE Transactions on Communications*, vol. 43, pp. 1493–1504, February/March/April 1995.

[11] A. Swami and B. Sadler, "Hierarchical Digital Modulation Classification Using Cumulants," *IEEE Transactions on Communications*, vol. 48, no. 3, pp. 416–429, 2000.

[12] O. A. Dobre, Y. Bar-Ness, and W. Su, "Higher-Order Cyclic Cumulants for High Order Modulation Classification," *Military Communications Conference, 2003. MILCOM 2003, IEEE*, vol. 1, Boston, Massachusetts, pp. 112–117, 2003.

[13] P. Panagiotou, A. Anastasopoulos, and A. Polydoros, "Likelihood Ratio Tests for Modulation Classification," *MILCOM 2000, 21st Century Military Communications Conference Proceedings*, vol. 2, Los Angeles, California, pp. 670–674, 2000.

[14] C. Long, K. Chugg, and A. Polydoros, "Further Results in Likelihood Classification of QAM Signals," *Military Communications Conference, 1994. MILCOM '94. Conference Record, 1994 IEEE*, Fort Monmouth, New Jersey, pp. 57–61, 1994.

[15] T. M. Cover and J. A. Thomas, *Elements of Information Theory*, New York: Wiley, 1991.

[16] P. A. Baker, "Phase-Modulation Data Sets for Serial Transmission at 200 and 2400 Bits per Second, Part I," *AIEE Trans. on Commun. Electron.*, July 1962.

[17] M. K. Simon, S. M. Hinedi, and W. C. Lindsey, *Digital Communication Techniques: Signal Design and Detection*, Upper Saddle River, New Jersey: Prentice Hall, 1995.

Appendix 9-A
Parameter Estimation for the GLRT

To obtain the analytical form of the ML estimates of the unknown parameters in terms of the observables, one must make certain approximations to the nonlinear functions involved in Eqs. (9-28) and (9-29). In particular, for small arguments (e.g., low SNR), applying the approximations $\ln(1+x) \cong x$ and $\cosh(x) \cong 1 + x^2/2$ to Eq. (9-28) and letting $\tilde{r}_n(\varepsilon) = r_{I,n}(\varepsilon) + jr_{Q,n}(\varepsilon)$ gives

$$\begin{aligned}
\hat{\theta}_{c2}, \hat{\varepsilon}_2 &= \underset{\theta_c,\varepsilon}{\operatorname{argmax}} \sum_{n=0}^{N-1} \frac{1}{2} \left(\frac{\sqrt{2P_t}}{\sigma^2} \operatorname{Re}\left\{ \tilde{r}_n(\varepsilon) e^{-j\theta_c} \right\} \right)^2 \\
&= \underset{\theta_c,\varepsilon}{\operatorname{argmax}} \sum_{n=0}^{N-1} \frac{1}{2} \left(\frac{\sqrt{2P_t}}{\sigma^2} \right)^2 \left(r_{In}(\varepsilon)\cos\theta_c + r_{Qn}(\varepsilon)\sin\theta_c \right)^2 \\
&= \underset{\theta_c,\varepsilon}{\operatorname{argmax}} \sum_{n=0}^{N-1} \left[\frac{1}{2}\left(r_{In}^2(\varepsilon) + r_{Qn}^2(\varepsilon) \right) + \frac{1}{2}\left(r_{In}^2(\varepsilon) - r_{Qn}^2(\varepsilon) \right)\cos 2\theta_c \right. \\
&\quad \left. + r_{In}(\varepsilon) r_{Qn}(\varepsilon) \sin 2\theta_c \right] \\
&= \underset{\theta_c,\varepsilon}{\operatorname{argmax}} \sum_{n=0}^{N-1} \left[\frac{1}{2}\left(r_{In}^2(\varepsilon) - r_{Qn}^2(\varepsilon) \right)\cos 2\theta_c + r_{In}(\varepsilon) r_{Qn}(\varepsilon) \sin 2\theta_c \right] \\
&= \underset{\theta_c,\varepsilon}{\operatorname{argmax}} \frac{1}{2} \sum_{n=0}^{N-1} \operatorname{Re}\left\{ \tilde{r}_n^2(\varepsilon) e^{-j2\theta_c} \right\}
\end{aligned} \tag{A-1}$$

Equivalently, for a given ε, $\hat{\theta}_{c_2}$ is the solution of

$$\begin{aligned}
&\frac{d}{d\theta_c}\sum_{n=0}^{N-1}\left[\frac{1}{2}\left(r_{In}^2(\varepsilon)-r_{Qn}^2(\varepsilon)\right)\cos 2\theta_c + r_{In}(\varepsilon)\, r_{Qn}(\varepsilon)\sin 2\theta_c\right] \\
&\qquad = -\frac{1}{2}\sin 2\theta_c \sum_{n=0}^{N-1}\left(r_{In}^2(\varepsilon)-r_{Qn}^2(\varepsilon)\right) + \cos 2\theta_c \sum_{n=0}^{N-1} r_{In}(\varepsilon)\, r_{Qn}(\varepsilon) \\
&\qquad = \frac{1}{2}\sum_{n=0}^{N-1}\operatorname{Im}\left\{\tilde{r}_n^2(\varepsilon)\, e^{-j2\theta_c}\right\} = 0
\end{aligned} \tag{A-2}$$

or

$$\hat{\theta}_{c_2} = \tan^{-1}\frac{\sum\limits_{n=0}^{N-1} r_{In}(\varepsilon)\, r_{Qn}(\varepsilon)}{\frac{1}{2}\sum\limits_{n=0}^{N-1}\left(r_{In}^2(\varepsilon)-r_{Qn}^2(\varepsilon)\right)} \tag{A-3}$$

which can be written in the simpler form

$$\hat{\theta}_{c_2} = \frac{1}{2}\arg\left(\sum_{n=0}^{N-1}\tilde{r}_n^2(\varepsilon)\right) \tag{A-4}$$

This estimate of θ_c is a generalization of the ML estimate when $\varepsilon = 0$ as given in Eq. (9-16), for the special case of $H = 2$.

Similarly, applying the same approximations of the nonlinearities to Eq. (9-29) gives

$$\begin{aligned}
&\hat{\theta}_{c4},\hat{\varepsilon}_4 \\
&= \underset{\theta_c,\varepsilon}{\operatorname{argmax}}\sum_{n=0}^{N-1}\frac{1}{2}\left[\left(\frac{\sqrt{2P_t}}{\sigma^2}\operatorname{Re}\left\{\tilde{r}_n(\varepsilon)\, e^{-j\theta_c}\right\}\right)^2 + \left(\frac{\sqrt{2P_t}}{\sigma^2}\operatorname{Im}\left\{\tilde{r}_n(\varepsilon)\, e^{-j\theta_c}\right\}\right)^2\right] \\
&= \underset{\theta_c,\varepsilon}{\operatorname{argmax}}\sum_{n=0}^{N-1}\frac{1}{2}\left(\frac{\sqrt{2P_t}}{\sigma^2}\right)^2\left|\tilde{r}_n(\varepsilon)\right|^2
\end{aligned} \tag{A-5}$$

which unfortunately is independent of θ_c. Thus, we see that for QPSK we must take the next term in the approximation of the hyperbolic cosine function, i.e., we should use

$$\begin{aligned} \ln(1+x) &\cong x \\ \cosh x &\cong 1 + \frac{x^2}{2} + \frac{x^4}{24} \end{aligned} \tag{A-6}$$

When this is applied to Eq. (9-29), we obtain

$$\begin{aligned} &\hat{\theta}_{c4}, \varepsilon_4 \\ &= \underset{\theta_c,\varepsilon}{\operatorname{argmax}} \sum_{n=0}^{N-1} \frac{1}{48} \left[\left(\frac{\sqrt{2P_t}}{\sigma^2} \operatorname{Re}\left\{ \tilde{r}_n(\varepsilon) e^{-j\theta_c} \right\} \right)^4 + \left(\frac{\sqrt{2P_t}}{\sigma^2} \operatorname{Im}\left\{ \tilde{r}_n(\varepsilon) e^{-j\theta_c} \right\} \right)^4 \right] \\ &= \underset{\theta_c,\varepsilon}{\operatorname{argmax}} \sum_{n=0}^{N-1} \frac{1}{48} \left(\frac{\sqrt{2P_t}}{\sigma^2} \right)^4 \left\{ \left[\left(\operatorname{Re}\left\{ \tilde{r}_n(\varepsilon) e^{-j\theta_c} \right\} \right)^2 + \left(\operatorname{Im}\left\{ \tilde{r}_n(\varepsilon) e^{-j\theta_c} \right\} \right)^2 \right]^2 \right. \\ &\quad \left. -2 \left(\operatorname{Re}\left\{ \tilde{r}_n(\varepsilon) e^{-j\theta_c} \right\} \right)^2 \left(\operatorname{Im}\left\{ \tilde{r}_n(\varepsilon) e^{-j\theta_c} \right\} \right)^2 \right\} \\ &= \underset{\theta_c,\varepsilon}{\operatorname{argmin}} \sum_{n=0}^{N-1} \left(\operatorname{Re}\left\{ \tilde{r}_n(\varepsilon) e^{-j\theta_c} \right\} \right)^2 \left(\operatorname{Im}\left\{ \tilde{r}_n(\varepsilon) e^{-j\theta_c} \right\} \right)^2 \end{aligned} \tag{A-7}$$

After some manipulation it can be shown that, analogous to Eq. (A-4), the ML estimate of θ_c for a given ε is given by

$$\hat{\theta}_{c_4} = \frac{1}{4} \arg \left(\sum_{n=0}^{N-1} \tilde{r}_n^4(\varepsilon) \right) \tag{A-8}$$

that again is a generalization of Eq. (9-16) to nonzero $\hat{\varepsilon}$, for the special case of $H = 4$.

For the ML estimates of symbol timing, we return to the exact forms in Eqs. (9-28) and (9-29). For example, for BPSK and a given value of θ_c, differentiating Eq. (9-28) (with ε replaced by $\hat{\varepsilon}$) with respect to $\hat{\varepsilon}$ and equating the result to zero gives the following:

$$\begin{aligned} &\frac{d}{d\hat{\varepsilon}} \sum_{n=0}^{N-1} \ln\cosh \left[\frac{\sqrt{2P_t}}{\sigma^2} \operatorname{Re}\left\{ \tilde{r}_n(\hat{\varepsilon}) e^{-j\theta_c} \right\} \right] \\ &\quad = \sum_{n=0}^{N-1} \left(\tanh \left[\frac{\sqrt{2P_t}}{\sigma^2} \operatorname{Re}\left\{ \tilde{r}_n(\hat{\varepsilon}) e^{-j\theta_c} \right\} \right] \right) \left(\frac{d}{d\hat{\varepsilon}} \left[\frac{\sqrt{2P_t}}{\sigma^2} \operatorname{Re}\left\{ \tilde{r}_n(\hat{\varepsilon}) e^{-j\theta_c} \right\} \right] \right) \\ &\quad = \sum_{n=0}^{N-1} \operatorname{Re}\left\{ \tilde{r}_n'(\hat{\varepsilon}) e^{-j\theta_c} \right\} \left(\tanh \left[\frac{\sqrt{2P_t}}{\sigma^2} \operatorname{Re}\left\{ \tilde{r}_n(\hat{\varepsilon}) e^{-j\theta_c} \right\} \right] \right) = 0 \end{aligned} \tag{A-9}$$

where

$$\tilde{r}'_n(\hat{\varepsilon}) = \frac{2}{T}\int_{(n-\hat{\varepsilon})T}^{(n+1-\hat{\varepsilon})T} \tilde{r}(t)\,\frac{d}{d\hat{\varepsilon}}p(t-nT-\hat{\varepsilon}T)\,dt \tag{A-10}$$

$$= -\frac{2}{T}\int_{(n-\hat{\varepsilon})T}^{(n+1-\hat{\varepsilon})T} \tilde{r}(t)\,\frac{d}{dt}p(t-nT-\hat{\varepsilon}T)\,dt \tag{A-11}$$

Unfortunately Eq. (A-10) does not yield a closed-form solution for $\hat{\varepsilon}$. A similar situation takes place for QPSK, namely,

$$\frac{d}{d\hat{\varepsilon}}\sum_{n=0}^{N-1}\ln\left[\frac{1}{2}\cosh\left[\frac{\sqrt{2P_t}}{\sigma^2}\mathrm{Re}\left\{\tilde{r}_n(\hat{\varepsilon})e^{-j\theta_c}\right\}\right]+\frac{1}{2}\cosh\left[\frac{\sqrt{2P_t}}{\sigma^2}\mathrm{Im}\left\{\tilde{r}_n(\hat{\varepsilon})e^{-j\theta_c}\right\}\right]\right]$$

$$=\sum_{n=0}^{N-1}\frac{\sinh\left(\frac{\sqrt{2P_t}}{\sigma^2}\mathrm{Re}\left\{\tilde{r}_n(\hat{\varepsilon})e^{-j\theta_c}\right\}\right)\mathrm{Re}\left\{\tilde{r}'_n(\hat{\varepsilon})e^{-j\theta_c}\right\}+\sinh\left(\frac{\sqrt{2P_t}}{\sigma^2}\mathrm{Im}\left\{\tilde{r}_n(\hat{\varepsilon})e^{-j\theta_c}\right\}\right)\mathrm{Im}\left\{\tilde{r}'_n(\hat{\varepsilon})e^{-j\theta_c}\right\}}{\cosh\left[\frac{\sqrt{2P_t}}{\sigma^2}\mathrm{Re}\left\{\tilde{r}_n(\hat{\varepsilon})e^{-j\theta_c}\right\}\right]+\cosh\left[\frac{\sqrt{2P_t}}{\sigma^2}\mathrm{Im}\left\{\tilde{r}_n(\hat{\varepsilon})e^{-j\theta_c}\right\}\right]}$$

$$=0 \tag{A-12}$$

If one now applies the approximations

$$\begin{aligned}\tanh x &\cong x\\ \sinh x &\cong x\\ \cosh x &\cong 1\end{aligned} \tag{A-13}$$

then Eqs. (A-9) and (A-12) simplify respectively to

$$\sum_{n=0}^{N-1}\mathrm{Re}\left\{\tilde{r}'_n(\hat{\varepsilon})\,e^{-j\theta_c}\right\}\mathrm{Re}\left\{\tilde{r}_n(\hat{\varepsilon})\,e^{-j\theta_c}\right\}$$

$$=\sum_{n=0}^{N-1}\left[\mathrm{Re}\left\{\tilde{r}'_n(\hat{\varepsilon})\left(\tilde{r}_n(\hat{\varepsilon})\right)^*\right\}+\mathrm{Re}\left\{\tilde{r}'_n(\hat{\varepsilon})\,\tilde{r}_n(\hat{\varepsilon})\,e^{-2j\theta_c}\right\}\right]=0 \tag{A-14}$$

and

$$\sum_{n=0}^{N-1}\left[\mathrm{Re}\left\{\tilde{r}_n\left(\hat{\varepsilon}\right)e^{-j\theta_c}\right\}\mathrm{Re}\left\{\tilde{r}'_n\left(\hat{\varepsilon}\right)e^{-j\theta_c}\right\}+\mathrm{Im}\left\{\tilde{r}_n\left(\hat{\varepsilon}\right)e^{-j\theta_c}\right\}\mathrm{Im}\left\{\tilde{r}'_n\left(\hat{\varepsilon}\right)e^{-j\theta_c}\right\}\right]$$

$$=\sum_{n=0}^{N-1}\mathrm{Re}\left\{\tilde{r}_n\left(\hat{\varepsilon}\right)\left(\tilde{r}'_n\left(\hat{\varepsilon}\right)\right)^*\right\}=0 \tag{A-15}$$

both of which require numerical solution for their respective ML estimates $\hat{\varepsilon}_2$ and $\hat{\varepsilon}_4$. Note from Eq. (A-15) that the ML estimate of symbol timing for the QPSK hypothesis is independent of the carrier phase estimate.

Appendix 9-B

ML Estimation of Carrier Phase for π/4-QPSK Modulation

To obtain the estimator of carrier phase needed for the GLRT involving $\pi/4$-QPSK modulation, we need to find the solution of

$$\hat{\theta}_{c,\pi/4-4}=\arg\max_{\theta_c}\left[\sum_{n=1,3,5,\cdots}^{N-1}\ln\left(\frac{1}{2}\left[\cosh\left[x_n\left(0;\theta_c\right)\right]+\cosh\left[x_n\left(1;\theta_c\right)\right]\right]\right)\right.$$

$$\left.+\sum_{n=0,2,4,\cdots}^{N-2}\ln\left(\frac{1}{2}\left[\cosh\left[y_n\left(0;\theta_c\right)\right]+\cosh\left[y_n\left(1;\theta_c\right)\right]\right]\right)\right] \tag{B-1}$$

with $x_n\left(q;\theta_c\right)$ and $y_n\left(q;\theta_c\right)$ as defined in Eq. (9-32). Applying the small argument approximations $\ln\left(1+x\right)\cong x$, $\cosh x\cong 1+x^2/2+x^4/24$ and ignoring second-order terms (since they contribute terms that do not depend on θ_c) gives

$$\hat{\theta}_{c,\pi/4-4}$$

$$= \arg\max_{\theta_c} \left[\frac{1}{48} \sum_{\substack{n= \\ 1,3,5,\cdots}}^{N-1} \left[\left(\frac{\sqrt{2P_t}}{\sigma^2} \mathrm{Re}\left\{ \tilde{r}_n e^{-j\theta_c} \right\} \right)^4 + \left(\frac{\sqrt{2P_t}}{\sigma^2} \mathrm{Im}\left\{ \tilde{r}_n e^{-j\theta_c} \right\} \right)^4 \right] \right.$$

$$\left. + \frac{1}{48} \sum_{\substack{n= \\ 0,2,4,\cdots}}^{N-2} \left[\left(\frac{\sqrt{2P_t}}{\sigma^2} \mathrm{Re}\left\{ \tilde{r}_n e^{-j\pi/4} e^{-j\theta_c} \right\} \right)^4 + \left(\frac{\sqrt{2P_t}}{\sigma^2} \mathrm{Im}\left\{ \tilde{r}_n e^{-j\pi/4} e^{-j\theta_c} \right\} \right)^4 \right] \right]$$

$$= \arg\max_{\theta_c} \left[\frac{1}{48} \sum_{\substack{n= \\ 1,3,5,\cdots}}^{N-1} \left\{ \left[\left(\frac{\sqrt{2P_t}}{\sigma^2} \mathrm{Re}\left\{ \tilde{r}_n e^{-j\theta_c} \right\} \right)^2 + \left(\frac{\sqrt{2P_t}}{\sigma^2} \mathrm{Im}\left\{ \tilde{r}_n e^{-j\theta_c} \right\} \right)^2 \right]^2 \right. \right.$$

$$\left. -2 \left(\frac{\sqrt{2P_t}}{\sigma^2} \mathrm{Re}\left\{ \tilde{r}_n e^{-j\theta_c} \right\} \right)^2 \left(\frac{\sqrt{2P_t}}{\sigma^2} \mathrm{Im}\left\{ \tilde{r}_n e^{-j\theta_c} \right\} \right)^2 \right\}$$

$$+ \frac{1}{48} \sum_{\substack{n= \\ 0,2,4,\cdots}}^{N-2} \left\{ \left[\left(\frac{\sqrt{2P_t}}{\sigma^2} \mathrm{Re}\left\{ \tilde{r}_n e^{-j\pi/4} e^{-j\theta_c} \right\} \right)^2 + \left(\frac{\sqrt{2P_t}}{\sigma^2} \mathrm{Im}\left\{ \tilde{r}_n e^{-j\pi/4} e^{-j\theta_c} \right\} \right)^2 \right]^2 \right.$$

$$\left. \left. -2 \left(\frac{\sqrt{2P_t}}{\sigma^2} \mathrm{Re}\left\{ \tilde{r}_n e^{-j\pi/4} e^{-j\theta_c} \right\} \right)^2 \left(\frac{\sqrt{2P_t}}{\sigma^2} \mathrm{Im}\left\{ \tilde{r}_n e^{-j\pi/4} e^{-j\theta_c} \right\} \right)^2 \right\} \right] \qquad \text{(B-2)}$$

or equivalently

$$\hat{\theta}_{c,\pi/4-4} = \arg\min_{\theta_c} \left[\sum_{n=1,3,5,\cdots}^{N-1} \left(\mathrm{Re}\left\{ \tilde{r}_n e^{-j\theta_c} \right\} \right)^2 \left(\mathrm{Im}\left\{ \tilde{r}_n e^{-j\theta_c} \right\} \right)^2 \right.$$

$$\left. + \sum_{n=0,2,4,\cdots}^{N-2} \left(\mathrm{Re}\left\{ \tilde{r}_n e^{-j\pi/4} e^{-j\theta_c} \right\} \right)^2 \left(\mathrm{Im}\left\{ \tilde{r}_n e^{-j\pi/4} e^{-j\theta_c} \right\} \right)^2 \right] \qquad \text{(B-3)}$$

Letting $\tilde{r}_n = |\tilde{r}_n| \, e^{j\phi_n}$, the above becomes

$$\begin{aligned}
\hat{\theta}_{c,\pi/4-4} = & \arg\min_{\theta_c} \left[\sum_{n=1,3,5,\cdots}^{N-1} |\tilde{r}_n|^4 \cos^2(\phi_n - \theta_c) \sin^2(\phi_n - \theta_c) \right. \\
& \left. + \sum_{n=0,2,4,\cdots}^{N-2} |\tilde{r}_n|^4 \cos^2\left(\phi_n - \frac{\pi}{4} - \theta_c\right) \sin^2\left(\phi_n - \frac{\pi}{4} - \theta_c\right) \right] \\
= & \arg\min_{\theta_c} \left[\frac{1}{4} \sum_{n=1,3,5,\cdots}^{N-1} |\tilde{r}_n|^4 - \frac{1}{8} \sum_{n=1,3,5,\cdots}^{N-1} |\tilde{r}_n|^4 \cos\left[4(\phi_n - \theta_c)\right] \right. \\
& \left. - \frac{1}{8} \sum_{n=0,2,4,\cdots}^{N-2} |\tilde{r}_n|^4 \cos\left[4\left(\phi_n - \frac{\pi}{4} - \theta_c\right)\right] \right] \\
= & \arg\max_{\theta_c} \left[\sum_{n=1,3,5,\cdots}^{N-1} |\tilde{r}_n|^4 \cos\left[4(\phi_n - \theta_c)\right] \right. \\
& \left. + \sum_{n=0,2,4,\cdots}^{N-2} |\tilde{r}_n|^4 \cos\left[4\left(\phi_n - \frac{\pi}{4} - \theta_c\right)\right] \right] \qquad \text{(B-4)}
\end{aligned}$$

This can also be written in the form

$$\begin{aligned}
& \hat{\theta}_{c,\pi/4-4} \\
= & \arg\max_{\theta_c} \left[\mathrm{Re}\left\{ \sum_{n=1,3,5,\cdots}^{N-1} \tilde{r}_n^4 \right\} \mathrm{Re}\left\{e^{j4\theta_c}\right\} + \mathrm{Im}\left\{ \sum_{n=1,3,5,\cdots}^{N-1} \tilde{r}_n^4 \right\} \mathrm{Im}\left\{e^{j4\theta_c}\right\} \right. \\
& \left. + \mathrm{Re}\left\{ \sum_{n=0,2,4,\cdots}^{N-2} \tilde{r}_n^4 \right\} \mathrm{Re}\left\{e^{j4\left(\frac{\pi}{4}+\theta_c\right)}\right\} + \mathrm{Im}\left\{ \sum_{n=0,2,4,\cdots}^{N-2} \tilde{r}_n^4 \right\} \mathrm{Im}\left\{e^{j4\left(\frac{\pi}{4}+\theta_c\right)}\right\} \right] \\
= & \arg\max_{\theta_c} \left\{ \left| \sum_{n=1,3,5,\cdots}^{N-1} \tilde{r}_n^4 \right| \cos\left[\arg\left(\sum_{n=1,3,5,\cdots}^{N-1} \tilde{r}_n^4 \right) - 4\theta_c \right] \right. \\
& \left. + \left| \sum_{n=0,2,4,\cdots}^{N-2} \tilde{r}_n^4 \right| \cos\left[\arg\left(\sum_{n=0,2,4,\cdots}^{N-2} \tilde{r}_n^4 \right) - \pi - 4\theta_c \right] \right\} \\
= & \arg\max_{\theta_c} \left\{ \left| \sum_{n=1,3,5,\cdots}^{N-1} \tilde{r}_n^4 \right| \cos\left[\arg\left(\sum_{n=1,3,5,\cdots}^{N-1} \tilde{r}_n^4 \right) - 4\theta_c \right] \right. \\
& \left. - \left| \sum_{n=0,2,4,\cdots}^{N-2} \tilde{r}_n^4 \right| \cos\left[\arg\left(\sum_{n=0,2,4,\cdots}^{N-2} \tilde{r}_n^4 \right) - 4\theta_c \right] \right\} \qquad \text{(B-5)}
\end{aligned}$$

However, using the trigonometric identity

$$A\cos(a-x)-B\cos(b-x)=(A\cos a-B\cos b)\cos x+(A\sin a-B\sin b)\sin x$$

$$=\sqrt{(A\cos a-B\cos b)^2+(A\sin a-B\sin b)^2}\cos(x-\eta) \tag{B-6}$$

$$\eta=\tan^{-1}\frac{A\sin a-B\sin b}{A\cos a-B\cos b}$$

in Eq. (B-5) gives

$$\hat{\theta}_{c,\pi/4-4}=\arg\max_{\theta_c}\left[\cos(4\theta_c-\eta)\right] \tag{B-7}$$

where

$$\eta=\tan^{-1}\frac{\mathrm{Im}\left\{\sum\limits_{n=1,3,5,\cdots}^{N-1}\tilde{r}_n^4\right\}-\mathrm{Im}\left\{\sum\limits_{n=0,2,4,\cdots}^{N-2}\tilde{r}_n^4\right\}}{\mathrm{Re}\left\{\sum\limits_{n=1,3,5,\cdots}^{N-1}\tilde{r}_n^4\right\}-\mathrm{Re}\left\{\sum\limits_{n=0,2,4,\cdots}^{N-2}\tilde{r}_n^4\right\}} \tag{B-8}$$

Finally then,

$$\hat{\theta}_{c,\pi/4-4}=\frac{1}{4}\tan^{-1}\frac{\mathrm{Im}\left\{\sum\limits_{n=1,3,5,\cdots}^{N-1}\tilde{r}_n^4\right\}-\mathrm{Im}\left\{\sum\limits_{n=0,2,4,\cdots}^{N-2}\tilde{r}_n^4\right\}}{\mathrm{Re}\left\{\sum\limits_{n=1,3,5,\cdots}^{N-1}\tilde{r}_n^4\right\}-\mathrm{Re}\left\{\sum\limits_{n=0,2,4,\cdots}^{N-2}\tilde{r}_n^4\right\}} \tag{B-9}$$

$$=\frac{1}{4}\tan^{-1}\frac{\mathrm{Im}\left\{\sum\limits_{n=0}^{N-1}(-1)^{n-1}\tilde{r}_n^4\right\}}{\mathrm{Re}\left\{\sum\limits_{n=0}^{N-1}(-1)^{n-1}\tilde{r}_n^4\right\}}=\frac{1}{4}\arg\left(\sum_{n=0}^{N-1}(-1)^{n-1}\tilde{r}_n^4\right)$$

$$=\frac{1}{4}\arg\left(\sum_{n=0}^{N-1}\left(\tilde{r}_n e^{j(\pi/4)(n-1)}\right)^4\right)=\frac{1}{4}\arg\left(\sum_{n=0}^{N-1}\left(\tilde{r}_n e^{-j(\pi/4)I_n}\right)^4\right) \tag{B-10}$$

where I_n is the indicator variable defined by

$$I_n = \begin{cases} 0, & n \text{ odd} \\ 1, & n \text{ even} \end{cases} \tag{B-11}$$

Chapter 10
Symbol Synchronization

Marvin K. Simon

As we have seen in other chapters, the operation and performance of various receiver functions can be quite sensitive to knowledge of the timing (data transition epochs) of the received data symbols. Thus, the ability to accurately estimate this parameter and continuously update the estimate, i.e., perform symbol synchronization (sync), with little knowledge of other parameters is critical to successful operation of an autonomous receiver. Traditionally, symbol synchronization techniques have been developed assuming that the data symbols are binary, the modulation format, e.g., non-return to zero (NRZ) or Manchester data, is known a priori, and carrier synchronization is perfect. Thus, the symbol synchronization problem has been solved entirely at baseband, assuming perfect knowledge of the carrier phase and frequency.

Among the various symbol sync schemes that have been proposed in the literature, by far the most popular in terms of its application in binary communication systems is the data-transition tracking loop (DTTL) [1,2]. The scheme as originally proposed in the late 1960s is an in-phase–quadrature (I-Q) structure where the I arm produces a signal representing the polarity of a data transition (i.e., a comparison of hard (± 1) decisions on two successive symbols) and the Q arm output is a signal whose absolute value is proportional to the timing error between the received signal epoch and the receiver's estimate of it. The result of the product of the I and Q signals is an error signal that is proportional to this timing error, independent of the direction of the transition. Although originally introduced as an efficient symbol synchronization means for tracking an NRZ data signal received in additive white Gaussian noise (AWGN), it was later demonstrated (although not formally published) that the closed-loop DTTL structure can be obtained from a suitable interpretation of the maximum

a posteriori (MAP) open-loop estimate of symbol timing based on an observation of, say, N symbols at high symbol signal-to-noise ratio (SNR).

At the time of the DTTL's introduction, the binary communication systems in which the DTTL was employed were for the most part uncoded, and thus high symbol SNR was the region of primary interest. As time marched on, the design of communication systems became more and more power efficient through the application of error-correction coding, and as such a greater and greater demand was placed on the symbol synchronizer, which now had to operate in a low symbol[1] SNR region, with values based on today's coding technology perhaps as low as -8 dB. Since in this very low symbol SNR region, the DTTL scheme as originally proposed would no longer be the one motivated by MAP estimation theory, it is also likely that its tracking capability would be degraded in this region of operation. Despite this fact, the conventional DTTL appears to have continued to be used in coded communication applications.

Since autonomous receiver operation requires, in general, functioning over a wide range of SNRs, it is prudent to employ symbol-timing estimation and tracking schemes whose implementations can adapt themselves to this changing environment using the knowledge obtained from the SNR estimator. Furthermore, since as we have seen in a previous chapter, the SNR estimator itself requires knowledge of symbol timing, a means for obtaining a coarse estimate of this timing is essential.

In this chapter, we start out by considering the problem of obtaining symbol synchronization under the admittedly ideal assumption of perfect carrier synchronization. We refer to the class of schemes that results from solution of this problem as phase-coherent symbol synchronizers. In this context, we first review the MAP estimate of symbol timing based on an observation of a block of N symbols and then describe the means by which the conventional DTTL is motivated by this open-loop estimate. Next, we consider the appropriate modification of the DTTL so that it is motivated by the MAP estimate of symbol timing at low SNR; in particular, the I arm hard decisions are replaced by soft decisions whereupon, in the limiting case, the hard limiter is replaced by a linear device. As we shall show, such a loop will outperform the conventional DTTL at low SNR. We then consider the extension of the MAP-motivated closed-loop ideas to nonbinary modulations such as M-ary phase-shift keying (M-PSK) and quadrature amplitude modulation (QAM). Following this, we return to the open-loop MAP estimation of symbol sync and describe a sliding-window realization that provides sequential updates at the symbol (as opposed to the N-symbol block) rate and as such resembles the closed-loop techniques. Next, we investigate means of

[1] It is important to note here that, in a coded communication system, the symbol synchronizer precedes the decoder and thus performs its function on the *coded* symbols whose SNR is equal to the bit SNR times the code rate.

performing the symbol synchronization function in the absence of carrier phase information, i.e., so-called phase-noncoherent symbol synchronization. We show that a class of ad hoc symbol synchronizers previously proposed for solution of the phase-coherent symbol synchronization problem can be easily adapted to the noncoherent case. Finally, we propose a coarse symbol-timing estimator for use in the SNR estimation that is derived from the same statistics that are used to form the SNR estimate itself.

10.1 MAP-Motivated Closed-Loop Symbol Synchronization

Analogous to the maximum-likelihood (ML) approach taken in Chapter 9 on modulation classification, we first form the likelihood function (LF) of the received signal vector conditioned on the unknown parameters to be estimated. Specifically, for the case of M-PSK modulation with carrier phase and symbol timing as the unknown parameters, it was shown there that, after averaging over the data in a sequence of length-N symbols, the conditional-likelihood function (CLF) is given by

$$\mathrm{CLF}_M\left(\theta_c, \varepsilon\right) = C \exp\left[\sum_{n=0}^{N-1} \ln\left(\frac{2}{M} \sum_{q=0}^{M/2-1} \cosh\left[x_n\left(q; \theta_c, \varepsilon\right)\right]\right)\right] \tag{10-1}$$

where θ_c denotes the carrier phase, ε denotes the unknown fractional symbol timing, C is a constant independent of θ_c and ε, and

$$x_n\left(q; \theta_c, \varepsilon\right) \triangleq \frac{A}{\sigma^2} \mathrm{Re}\left\{\tilde{r}_n(\varepsilon) e^{-j\left(\left[2q + \left(1+(-1)^{M/2}\right)/2\right]\pi/M + \theta_c\right)}\right\} \tag{10-2}$$

with $A = \sqrt{2P}$ the signal amplitude (P is the transmitted power in the data)[2] and σ the standard deviation of the noise component (per dimension) of $\tilde{r}_n\left(\varepsilon\right)$. Also, in Eq. (10-2), the complex observables corresponding to the matched filter outputs at time instants $(n+\varepsilon)T,\ n = 0, 2, \cdots, N-1$ are given by

$$\tilde{r}_n\left(\varepsilon\right) = \frac{1}{T} \int_{(n+\varepsilon)T}^{(n+1+\varepsilon)T} \tilde{r}\left(t\right) p\left(t - nT - \varepsilon T\right) dt \tag{10-3}$$

[2] For simplicity of notation, we denote the data power by P rather than P_d since here we are not dealing with the power in the discrete carrier (if it exists) at all.

where $\tilde{r}(t)$ is the complex baseband received signal in the time interval $(n+\varepsilon)T \leq t \leq (n+1+\varepsilon)T$ and $p(t)$ is the pulse shape. Finally, the SNR at the complex output of the matched filter is given by $\gamma_s = A^2/(2\sigma^2) = E_s/N_0$, where $E_s = PT$ is the symbol energy and N_0 is the single-sided power spectral density of the additive noise.

For the purpose of finding the MAP estimate of symbol sync alone, we may assume perfect knowledge of the carrier phase, in which case, without any loss in generality, we can set $\theta_c = 0$. Under this assumption, the MAP estimate of symbol timing $\hat{\varepsilon}_{\text{MAP}}$ is given by

$$\hat{\varepsilon}_{\text{MAP}} = \underset{\hat{\varepsilon}}{\text{argmax}} \exp\left[\sum_{n=0}^{N-1} \ln\left(\frac{2}{M}\sum_{q=0}^{M/2-1} \cosh\left[x_n(q;\hat{\varepsilon})\right]\right)\right] \tag{10-4}$$

where now

$$\begin{aligned} & x_n(q;\varepsilon) \\ &= \frac{A}{\sigma^2}\,\text{Re}\left\{\tilde{r}_n(\varepsilon)\, e^{-j\left(\left[2q+\left(1+(-1)^{M/2}\right)/2\right]\pi/M\right)}\right\} \\ &= \frac{2\sqrt{P}}{N_0}\,\text{Re}\left\{e^{-j\left(\left[2q+\left(1+(-1)^{M/2}\right)/2\right]\pi/M\right)} \int_{(n+\varepsilon)T}^{(n+1+\varepsilon)T} \tilde{r}(t)\, p(t-nT-\varepsilon T)\, dt\right\} \end{aligned} \tag{10-5}$$

Note that the actual fractional symbol-timing offset ε is embedded in the received complex baseband signal $\tilde{r}(t)$, and thus the difference between $\hat{\varepsilon}_{\text{MAP}}$ and ε represents the normalized symbol-timing error.

As an alternative to Eq. (10-4), recognizing that the natural logarithm is a monotonic function of its argument, one can first take the natural logarithm of the CLF in Eq. (10-1), in which case the MAP estimate of symbol timing has the simpler form

$$\hat{\varepsilon}_{\text{MAP}} = \underset{\hat{\varepsilon}}{\text{argmax}} \left[\sum_{n=0}^{N-1} \ln\left(\frac{2}{M}\sum_{q=0}^{M/2-1} \cosh\left[x_n(q;\hat{\varepsilon})\right]\right)\right] \tag{10-6}$$

As is well-known in MAP-motivated closed-loop schemes, the argument can be made that, since the value of $\hat{\varepsilon}$ that maximizes the CLF is also the value at which the derivative of the CLF with respect to $\hat{\varepsilon}$ equates to zero, then one can use the CLF derivative itself as an error signal in a closed-loop symbol synchronization (tracking) configuration. As such, the MAP-motivated symbol synchronization loop would form

$$e = \frac{d}{d\hat{\varepsilon}} \left[\sum_{n=0}^{N-1} \ln \left(\frac{2}{M} \sum_{q=0}^{M/2-1} \cosh \left[x_n \left(q; \hat{\varepsilon} \right) \right] \right) \right]$$
$$= \sum_{n=0}^{N-1} \frac{\sum_{q=0}^{M/2-1} \sinh \left[x_n \left(q; \hat{\varepsilon} \right) \right] \frac{d}{d\hat{\varepsilon}} x_n \left(q; \hat{\varepsilon} \right)}{\sum_{q=0}^{M/2-1} \cosh \left[x_n \left(q; \hat{\varepsilon} \right) \right]} \tag{10-7}$$

as its error signal. Furthermore,

$$\begin{aligned} & x_n' \left(q; \hat{\varepsilon} \right) \\ & = \frac{A}{\sigma^2} \operatorname{Re} \left\{ \frac{d}{d\hat{\varepsilon}} \tilde{r}_n \left(\hat{\varepsilon} \right) e^{-j\left(\left[2q + \left(1 + (-1)^{M/2} \right)/2 \right] \pi / M \right)} \right\} \\ & = \frac{2\sqrt{P}}{N_0} \operatorname{Re} \left\{ e^{-j\left(\left[2q + \left(1 + (-1)^{M/2} \right)/2 \right] \pi / M \right)} \int_{(n+\hat{\varepsilon})T}^{(n+1+\hat{\varepsilon})T} \tilde{r} \left(t \right) \frac{dp \left(t - nT - \hat{\varepsilon} T \right)}{d\hat{\varepsilon}} dt \right\} \\ & = \frac{2\sqrt{P}}{N_0} \operatorname{Re} \left\{ -Te^{-j\left(\left[2q + \left(1 + (-1)^{M/2} \right)/2 \right] \pi / M \right)} \int_{(n+\hat{\varepsilon})T}^{(n+1+\hat{\varepsilon})T} \tilde{r} \left(t \right) p' \left(t - nT - \hat{\varepsilon} T \right) dt \right\} \end{aligned} \tag{10-8}$$

where the second equation follows from the Leibnitz rule, assuming $p(0) = p(T) = 0$. A closed-loop configuration that implements the expression in Eq. (10-7) as an error signal is referred to as a MAP estimation loop.

10.2 The DTTL as an Implementation of the MAP Estimation Loop for Binary NRZ Signals at High SNR

For binary signals ($M = 2$), the error signal of Eq. (10-7) simplifies to

$$e = \sum_{n=0}^{N-1} \tanh \left[x_n \left(0; \hat{\varepsilon} \right) \right] \frac{d}{d\hat{\varepsilon}} x_n \left(0; \hat{\varepsilon} \right) \tag{10-9}$$

where

$$\begin{aligned} x_n \left(0; \hat{\varepsilon} \right) &= \frac{2\sqrt{P}}{N_0} \int_{(n+\hat{\varepsilon})T}^{(n+1+\hat{\varepsilon})T} r \left(t \right) p \left(t - nT - \hat{\varepsilon} T \right) dt \\ x_n' \left(0; \hat{\varepsilon} \right) &= \frac{-2T\sqrt{P}}{N_0} \int_{(n+\hat{\varepsilon})T}^{(n+1+\hat{\varepsilon})T} r \left(t \right) p' \left(t - nT - \hat{\varepsilon} T \right) dt \end{aligned} \tag{10-10}$$

and $r(t)$ is now a real signal. A block diagram of a MAP estimation loop that uses e of Eq. (10-9) as an error signal to control a timing-pulse generator is illustrated in Fig. 10-1, where the shorthand notation $T_n(\hat{\varepsilon})$ has been introduced to represent the time interval $(n+\hat{\varepsilon})T \le t \le (n+1+\hat{\varepsilon})T$. In this figure, the accumulator represents the summation over N in Eq. (10-9). Thus, based on the above model, the loop would update itself in blocks of N symbols. In practice, however, one would replace this block-by-block accumulator by a digital filter that updates the loop every T seconds and whose impulse response is chosen to provide a desired dynamic response for the loop. The design of this filter and its associated closed-loop response characteristic are not dictated by the MAP estimation theory, which explains the use of the term "MAP-motivated" when describing the MAP estimation loop.

To go from the MAP estimation loop to the conventional DTTL, one needs to (1) approximate the hyperbolic tangent nonlinearity for large values of its argument, equivalently, at high SNR and (2) characterize, i.e., approximate, the derivative of the pulse shape required in Eq. (10-10). Specifically, for large values of its argument, one has the approximation

$$\tanh x \cong \operatorname{sgn} x \tag{10-11}$$

In theory, if $p(t)$ were a unit amplitude rectangular pulse shape, as would be the case for NRZ signals, then the derivative of $p(t)$ would be a positive delta function at the leading edge and a negative delta function at the trailing edge of the symbol interval. In practice, these unrealizable delta functions are replaced by a pair of narrow rectangular pulses whose width is treated as a design parameter. Denoting this pulse width by ξT, the above representation for two successive symbol intervals is shown in Fig. 10-2, where for simplicity of illustration we have assumed $\varepsilon = \hat{\varepsilon} = 0$. If we now group these pulses in pairs corresponding to

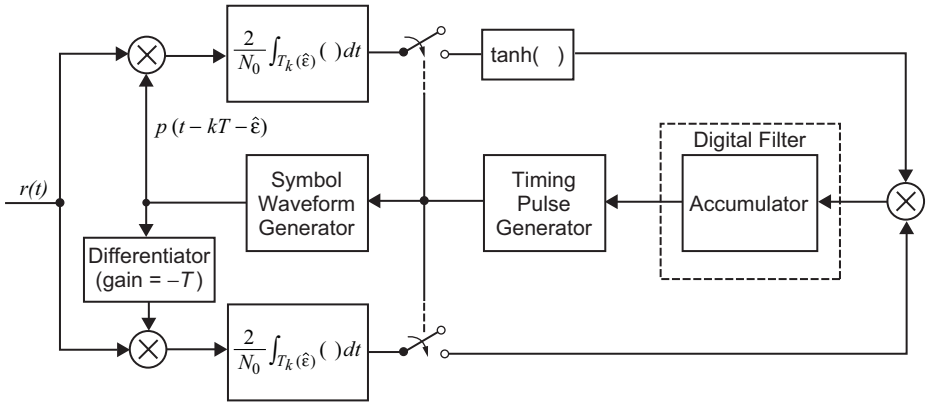


Fig. 10-1. A closed-loop symbol synchronizer motivated by the MAP estimation approach.

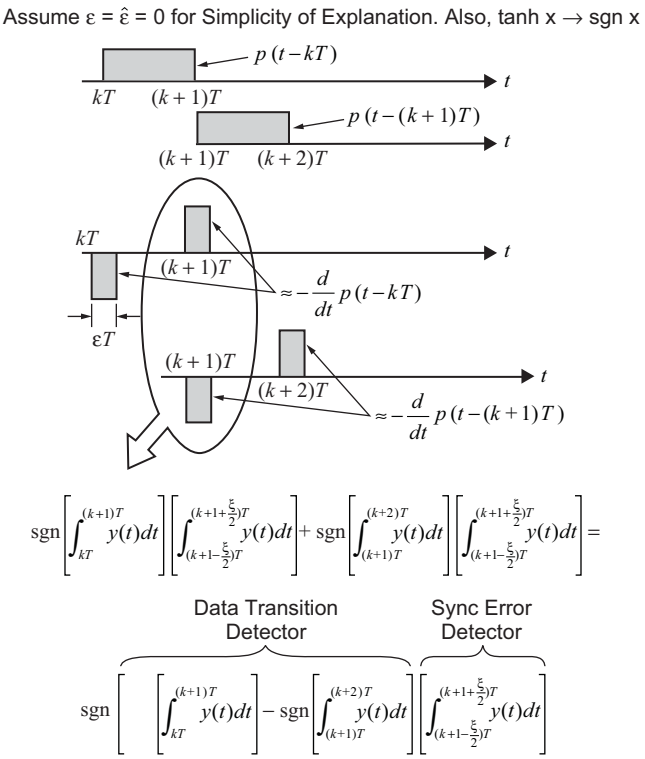


Fig. 10-2. Formation of the error signal from narrow-pulse approximation of the derivative of the pulse shape.

the trailing edge of one symbol and the leading edge of the next, then taking into account the approximation of the nonlinearity in Eq. (10-11), the contribution of the nth pair to the error signal in Eq. (10-9) would be expressed as

$$\begin{aligned} e_n = & \tanh\left(\frac{2\sqrt{P}}{N_0}\int_{nT}^{(n+1)T} r\,(t)\,p\,(t-nT)\,dt\right) \\ & \times \frac{2T\sqrt{P}}{N_0}\int_{nT}^{(n+1)T} r\,(t)\,p'\,(t-nT)\,dt \\ \cong & \left(\frac{2T\sqrt{P}}{N_0}\int_{(n+1-\xi/2)T}^{(n+1+\xi/2)T} r\,(t)\,dt\right)\left[\operatorname{sgn}\left(\frac{2\sqrt{P}}{N_0}\int_{nT}^{(n+1)T} r\,(t)\,dt\right)\right. \\ & \left.-\operatorname{sgn}\left(\frac{2\sqrt{P}}{N_0}\int_{(n+1)T}^{(n+2)T} r\,(t)\,dt\right)\right] \end{aligned} \tag{10-12}$$

The first factor in the final result of Eq. (10-12) represents an integration of width ξT *across* the data transition instant (often referred to as the window width of the synchronizer), whereas the second factor represents the difference of hard decisions on integrations *within* two successive symbol intervals. In the presence of a symbol-timing offset, when a data transition occurs, the first factor would provide a measure of the error between the actual symbol timing and the estimate of it produced by the loop. Thus, this factor is referred to as a sync error detector. The second factor is a measure of the *occurrence* of a transition in the data and thus is referred to as a data transition detector. Since the output of the sync error detector integrate-and-dump (I&D) occurs at time $(n + \xi/2)T$, where the data transition decision occurs at time $(n + 1)\,T$, one must delay the output of the former by an amount $(1 - \xi/2)T$ before the two can be multiplied. Based on the above assumptions and discussion, it is now clear that the MAP estimation loop migrates to the DTTL as illustrated in Fig. 10-3.

10.3 Conventional versus Linear Data Transition Tracking Loop

In the previous section, we observed that under high SNR conditions, where the nonlinearity is approximated as in Eq. (10-11), the I arm of the resulting symbol synchronizer becomes a detector of a transition in hard decisions made on successive symbols. In this section, we consider the synchronizer that results from approximating the nonlinearity for small values of its arguments, as would be appropriate at low SNR. It will be shown that, depending on the Q arm window width, there always exists a value of symbol SNR at which the

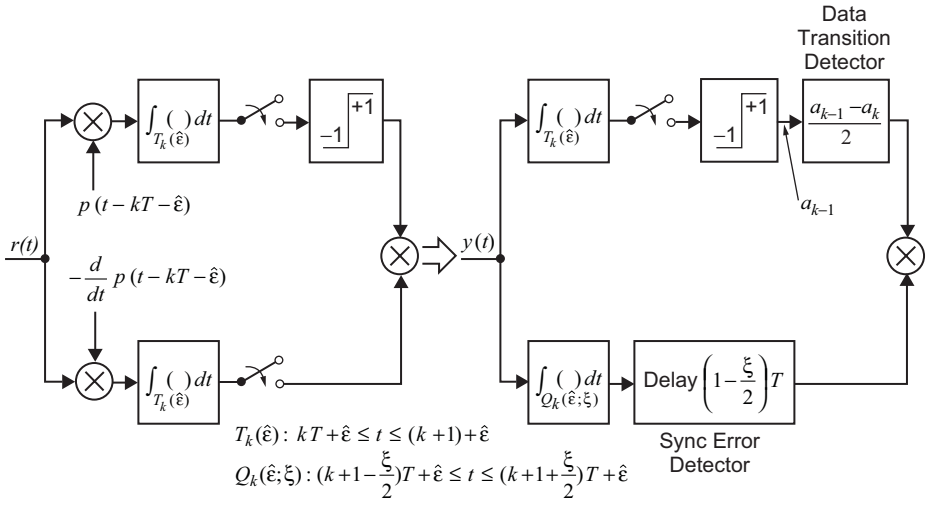


Fig. 10-3. Conventional DTTL derived from MAP estimation loop at high SNR.

linear data-transition tracking loop (LDTTL) outperforms the conventional DTTL with a hard decision I arm transition detector.

For sufficiently small values of its argument, the hyperbolic tangent nonlinearity can be approximated by

$$\tanh x \cong x \tag{10-13}$$

i.e., a linear function. Under this assumption, the appropriate MAP-motivated closed-loop synchronizer is illustrated in Fig. 10-4, and its performance is analyzed as follows. After perfect (known carrier phase) demodulation by the carrier reference $\sqrt{2}\cos(\omega_c t + \theta_c)$, the baseband signal input to the LDTTL is given by

$$\begin{aligned} r(t) &= s(t,\varepsilon) + n(t) \\ s(t,\varepsilon) &= \sqrt{P}\sum_{n=-\infty}^{\infty} d_n p(t - nT - \varepsilon T) \end{aligned} \tag{10-14}$$

where, consistent with the assumption of NRZ data, $p(t)$ is a unit amplitude rectangular pulse of duration T seconds and $\{d_n\}$ is an independent, identically distributed (iid) ± 1 sequence with d_n representing the polarity of the nth data symbol. The additive noise is a white Gaussian process with single-sided power spectral density N_0 W/Hz. The local clock produces a timing reference for the I and Q I&D filters that depends on its estimate $\hat{\varepsilon}$ of ε. Thus, the outputs of these filters corresponding to the nth symbol interval are respectively given by

$$\begin{aligned} y_{In} =& K_1 \int_{(n+\hat{\varepsilon})T}^{(n+1+\hat{\varepsilon})T} r(t)dt = \overbrace{K_1 \int_{(n+\hat{\varepsilon})T}^{(n+1+\hat{\varepsilon})T} s(t,\varepsilon)dt}^{c_n} \\ &+ \overbrace{K_1 \int_{(n+\hat{\varepsilon})T}^{(n+1+\hat{\varepsilon})T} n(t)dt}^{\nu_n} \\ y_{Qn} =& K_2 \int_{\left(n+1-\frac{\xi}{2}+\hat{\varepsilon}\right)T}^{\left(n+1+\frac{\xi}{2}+\hat{\varepsilon}\right)T} r(t)dt = \overbrace{K_2 \int_{\left(n+1-\frac{\xi}{2}+\hat{\varepsilon}\right)T}^{\left(n+1+\frac{\xi}{2}+\hat{\varepsilon}\right)T} s(t,\varepsilon)dt}^{b_n} \\ &+ \overbrace{K_2 \int_{\left(n+1-\frac{\xi}{2}+\hat{\varepsilon}\right)T}^{\left(n+1+\frac{\xi}{2}+\hat{\varepsilon}\right)T} n(t)dt}^{\mu_n} \end{aligned} \tag{10-15}$$

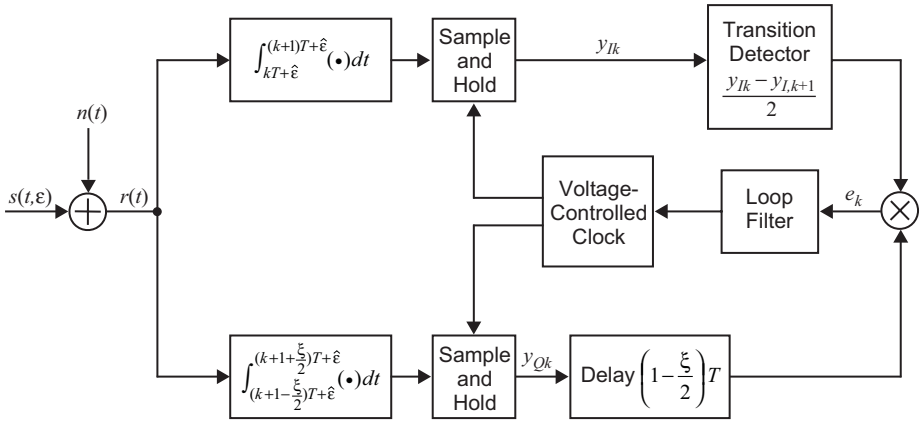


Fig. 10-4. The linear data transition tracking loop (LDTTL).

Since μ_n and ν_n are not independent, it is convenient, as was done in [1,2], to express them in terms of a new set of variables:

$$\begin{aligned}\nu_n &= N_n + M_n \\ \mu_n &= N'_{n+1} + M'_n\end{aligned} \tag{10-16}$$

where

$$\begin{aligned}N_n &= K_1 \int_{(n+\hat{\varepsilon})T}^{\left(n+\frac{1}{2}+\hat{\varepsilon}\right)T} n(t)dt, \quad M_n = K_1 \int_{\left(n+\frac{1}{2}+\hat{\varepsilon}\right)T}^{(n+1+\hat{\varepsilon})T} n(t)dt \\ N'_n &= K_2 \int_{(n+\hat{\varepsilon})T}^{\left(n+\frac{\xi}{2}+\hat{\varepsilon}\right)T} n(t)dt, \quad M'_n = K_2 \int_{\left(n+1-\frac{\xi}{2}+\hat{\varepsilon}\right)T}^{(n+1+\hat{\varepsilon})T} n(t)dt\end{aligned} \tag{10-17}$$

with the properties

N_k, M_n are mutually independent for all k, n

N'_k, M'_n are mutually independent for all k, n

N'_k, M_n and M'_k, N_n are mutually independent for all k, n

N'_k, N'_n and M_k, M_n are mutually independent for all $k \neq n$

Furthermore, all M_n, M'_n, N_n, N'_n, and their sums are Gaussian random variables with zero mean and variances

$$\sigma_{M_n}^2 = \sigma_{N_n}^2 = K_1^2 N_0 T/4$$
$$\sigma_{M_n'}^2 = \sigma_{N_n'}^2 = K_2^2 \xi N_0 T/4 \tag{10-18}$$

Taking the difference of two successive soft decisions y_{In} and $y_{I,n-1}$ and multiplying the average of the result by the quadrature I&D output (delayed by $(1-\xi/2)\,T$) gives the loop-error signal (prior to digital filtering),

$$e(t) = e_n = \left(b_n + M_n' + N_{n+1}'\right)\left[\frac{(c_n + M_n + N_n) - (c_{n+1} + M_{n+1} + N_{n+1}}{2}\right],$$
$$(n+2)T + \hat{\varepsilon} \le t \le (n+3) + T + \hat{\varepsilon} \tag{10-19}$$

which is a piecewise constant (over intervals of T seconds) random process. In Eq. (10-19),

$$b_n = \begin{cases} K_2\sqrt{P}T\left[d_n\left(\dfrac{\xi}{2}+\lambda\right) + d_{n+1}\left(\dfrac{\xi}{2}-\lambda\right)\right], & 0 \le \lambda \le \dfrac{\xi}{2} \\ K_2\sqrt{P}T d_n \xi, & \dfrac{\xi}{2} \le \lambda \le \dfrac{1}{2} \end{cases} \tag{10-20}$$
$$c_n = K_1\sqrt{P}T\left[d_{n-1}\lambda + d_n\left(1-\lambda\right)\right], \quad 0 \le \lambda \le \frac{1}{2}$$

where $\lambda \triangleq \varepsilon - \hat{\varepsilon},\ -1/2 \le \lambda \le 1/2$ denotes the normalized timing error.

10.3.1 The Loop S-Curve

The S-curve of the loop is by definition the statistical average of the error signal of Eq. (10-19) over the signal and noise probability distributions, i.e.,

$$g\left(\lambda\right) \triangleq E\left\{\left(b_n + M_n' + N_{n+1}'\right)\left[\frac{(c_n + M_n + N_n) - (c_{n+1} + M_{n+1} + N_{n+1})}{2}\right]\right\} \tag{10-21}$$

Substituting Eq. (10-20) into Eq. (10-21) and performing the necessary averaging over the noise and the data symbols d_{n-2}, d_{n-1}, and d_n gives the desired result, namely,

$$g_n(\lambda) \triangleq \frac{g(\lambda)}{K_1 K_2 P T^2} = \begin{cases} \lambda\left(1-\dfrac{\xi}{4}\right) - \dfrac{3}{2}\lambda^2, & 0 \le \lambda \le \dfrac{\xi}{2} \\ \dfrac{\xi}{2}(1-2\lambda), & \dfrac{\xi}{2} \le \lambda \le \dfrac{1}{2} \end{cases} \tag{10-22}$$

where the n subscript here stands for normalization. By comparison, the result corresponding to Eq. (10-22) for the DTTL is [1,2]

$$g_n(\lambda) \triangleq \frac{g(\lambda)}{K_2\sqrt{P}T} = \begin{cases} \lambda \operatorname{erf}\left(\sqrt{R_s}(1-2\lambda)\right) - \dfrac{1}{8}(\xi - 2\lambda) \\ \times \left[\operatorname{erf}\left(\sqrt{R_s}\right) - \operatorname{erf}\left(\sqrt{R_s}(1-2\lambda)\right)\right], & 0 \le \lambda \le \dfrac{\xi}{2} \\ \dfrac{\xi}{2}\operatorname{erf}\left(\sqrt{R_s}(1-2\lambda)\right), & \dfrac{\xi}{2} \le \lambda \le \dfrac{1}{2} \end{cases} \tag{10-23}$$

where $R_s \triangleq PT/N_0$ denotes the symbol SNR. Without belaboring the analysis, it is also straightforward to show that for $-(1/2) \le \lambda \le 0$, $g(\lambda) = -g(-\lambda)$, i.e., the S-curve is an odd function of the normalized timing error. Also note from Eq. (10-22) that the normalized S-curve for the LDTTL is independent of SNR, whereas that for the conventional DTTL [see Eq. (10-23)] is highly dependent on SNR. Figure 10-5 is an illustration of the S-curve in Eq. (10-22) for various values of window width ξ.

The slope of the normalized S-curve at the origin ($\lambda = 0$) will be of interest in computing the mean-squared timing-jitter performance. Taking the derivative of Eq. (10-22) with respect to λ and evaluating the result at $\lambda = 0$ gives for the LDTTL

$$K_g \triangleq \frac{dg(\lambda)}{d\lambda}\Big|_{\lambda=0} = K_1 K_2 P T^2 \left(1 - \frac{\xi}{4}\right) \tag{10-24}$$

whereas the corresponding result for the DTTL, based on the derivative of Eq. (10-23), is

$$K_g \triangleq \frac{dg(\lambda)}{d\lambda}\Big|_{\lambda=0} = K_2\sqrt{P}T\left[\operatorname{erf}\left(\sqrt{R_s}\right) - \frac{\xi}{2}\sqrt{\frac{R_s}{\pi}}\exp(-R_s)\right] \tag{10-25}$$

which clearly degrades with decreasing R_s.

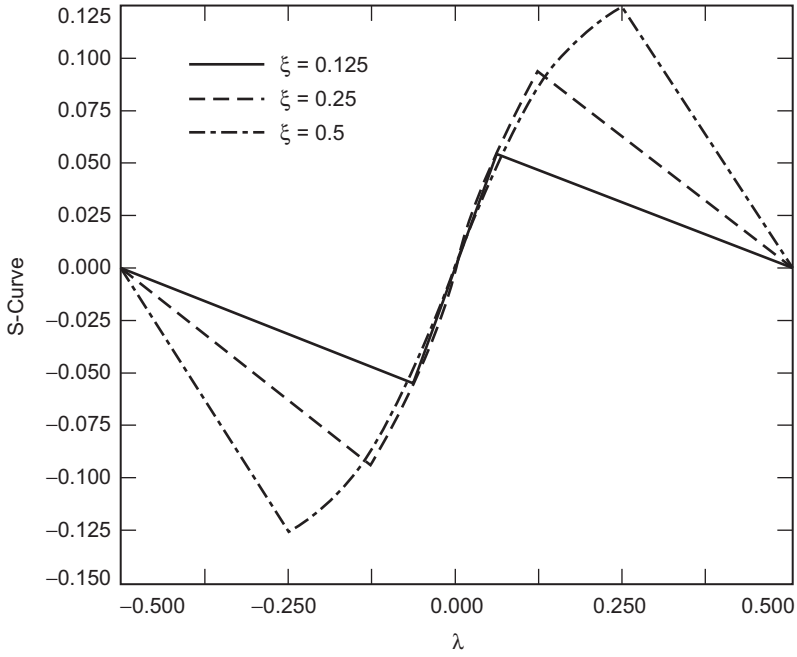


Fig. 10-5. Normalized S-curves for linear DTTL.

10.3.2 Noise Performance

The stochastic differential equation that characterizes the operation of the DTTL or the LDTTL is [1,2]

$$\dot{\lambda} = -KF(p)\left[g(\lambda) + n_\lambda(t)\right] \tag{10-26}$$

where K is the total loop gain, $F(p)$ is the transfer function of the loop filter with p denoting the Heaviside operator, and $n_\lambda(t)$ is the equivalent additive noise that characterizes the variation of the loop-error signal around its mean (the S-curve). Because of the I&D and sample-and-hold operations in the I and Q arms of the loops, $n_\lambda(t)$ is a piecewise (over intervals of T seconds) constant random process. In particular,

$$n_\lambda(t) = e_n - E\{e_n\} = e_n - g(\lambda), \quad (n+2+\hat{\varepsilon})T \le t \le (n+3+\hat{\varepsilon})T \tag{10-27}$$

with a covariance function that is piecewise linear between the sample values

$$
\begin{aligned}
R_n(\tau)\,|_{\tau=mT} &= E\{n_\lambda(t)\,n_\lambda(t+\tau)\}\,|_{\tau=mT} \\
&= E\{(e_n - E\{e_n\})\,(e_{n+m} - E\{e_{n+m}\})\} \\
&= E\{e_n e_{n+m}\} - g^2(\lambda) \triangleq R(m,\lambda), \quad m = 0, \pm 1, \pm 2, \cdots
\end{aligned} \tag{10-28}
$$

As is customary in the analysis of loops of this type, for loop bandwidths that are small compared to the reciprocal of the symbol time interval, $n_\lambda(t)$ can be approximated by a delta-correlated process with equivalent flat (with respect to frequency) power spectral density

$$
N_0' \triangleq 2\int_{-\infty}^{\infty} R_n(\tau)d\tau = 2T\left[R(0,\lambda) + 2\sum_{m=1}^{\infty} R(m,\lambda)\right] \tag{10-29}
$$

Furthermore, for large loop SNR,[3] it is customary to consider only the value of the equivalent power spectral density at $\lambda = 0$, namely,

$$
\begin{aligned}
N_0' &= 2T\left[R(0,0) + 2\sum_{m=1}^{\infty} R(m,0)\right] \\
&= 2T\left[E\left\{e_n^2\,|_{\lambda=0}\right\} + 2\sum_{m=1}^{\infty} E\{e_n e_{n+m}\,|_{\lambda=0}\}\right]
\end{aligned} \tag{10-30}
$$

With a good deal of effort, the following results can be obtained from Eq. (10-19):

$$
\begin{aligned}
&E\left\{e_n^2|_{\lambda=0}\right\} = \\
&\frac{1}{4}\Big[E\left\{b_n^2\,(c_{n+1}-c_n)^2\,|_{\lambda=0}\right\} + E\left\{b_n^2\right\} \\
&\times E\left\{(N_{n+1}+M_{n+1})^2 + (N_n+M_n)^2\right\} + E\left\{(c_{n+1}-c_n)^2\,|_{\lambda=0}\right\} \\
&\times E\left\{\left(N_{n+1}'+M_n'\right)^2\right\} + E\left\{\left(N_{n+1}'+M_n'\right)^2 (N_n+M_n-N_{n+1}-M_{n+1})^2\right\}\Big]
\end{aligned} \tag{10-31}
$$

[3] Note that this assumption does not require that the symbol SNR be large. Large loop SNR simply implies that the loop operates in the so-called linear region, i.e., where the mean-squared value of the timing error is small and the probability density function of the timing error is Gaussian distributed.

$$E\{e_n e_{n+1}|_{\lambda=0}\} = \frac{1}{4}\Big[E\{b_n b_{n+1}(c_{n+1}-c_n)(c_{n+2}-c_{n+1})|_{\lambda=0}\} - E\{b_n b_{n+1}|_{\lambda=0}\}E\left\{(N_{n+1}+M_{n+1})^2\right\}\Big] \quad (10\text{-}32)$$

$$E\{e_n e_{n+m}|_{\lambda=0}\} = 0, \quad m \neq 0,1 \quad (10\text{-}33)$$

Averaging Eqs. (10-31) through (10-33) over the signal (data sequence) and then using Eq. (10-18), we obtain the desired results, namely,

$$R(0,0) \triangleq E\{e_n^2|_{\lambda=0}\} = \left(K_1K_2PT^2\right)^2\left[\frac{\xi}{4R_s}\left(1+\frac{\xi}{2}+\frac{1}{2R_s}\right)\right] \quad (10\text{-}34)$$

$$R(1,0) \triangleq E\{e_n e_{n+1}|_{\lambda=0}\} = -\left(K_1K_2PT^2\right)^2\frac{\xi^2}{32R_s} \quad (10\text{-}35)$$

$$R(m,0) \triangleq E\{e_n e_{n+m}|_{\lambda=0}\} = 0, \quad m \neq 0,1 \quad (10\text{-}36)$$

Combining Eqs. (10-34) through (10-36), the equivalent power spectral density is then

$$N_0' = T\left(K_1K_2PT^2\right)^2\left[\frac{\xi}{2R_s}\left(1+\frac{\xi}{4}+\frac{1}{2R_s}\right)\right] \quad (10\text{-}37)$$

The equivalent quantity for the conventional DTTL can be obtained from the results in [1,2] to be

$$N_0' = T\left(K_2\sqrt{P}T\right)^2\left[\frac{\xi}{2R_s}\left[1+\frac{\xi R_s}{2}-\frac{\xi}{2}\left[\frac{1}{\sqrt{\pi}}\exp(-R_s)+\sqrt{R_s}\,\mathrm{erf}\sqrt{R_s}\right]^2\right]\right] \quad (10\text{-}38)$$

10.3.3 Mean-Squared Timing-Error Performance

The mean-squared timing jitter σ_λ^2 of either the LDTTL or the DTTL is readily computed for a first-order loop filter ($F(p) = 1$) and large loop SNR conditions. In particular, linearizing the S-curve to $g(\lambda) = K_g\lambda$ and denoting the single-sided loop bandwidth by B_L, we obtain

$$\sigma_\lambda^2 = \frac{N_0' B_L}{K_g^2} \tag{10-39}$$

where K_g is obtained from either Eq. (10-24) or Eq. (10-25) and N_0' from either Eq. (10-37) or Eq. (10-38). Making the appropriate substitutions in Eq. (10-39) gives the results

$$\sigma_\lambda^2 = \frac{\xi\left[1 + \frac{\xi}{4} + \frac{1}{2R_s}\right]}{2\rho\left(1 - \frac{\xi}{4}\right)^2} \text{ (LDTTL)}$$

$$\sigma_\lambda^2 = \frac{\xi\left[1 + \frac{\xi R_s}{2} - \frac{\xi}{2}\left[\frac{1}{\sqrt{\pi}}\exp\left(-R_s\right) + \sqrt{R_s}\,\text{erf}\,\sqrt{R_s}\right]^2\right]}{2\rho\left[\text{erf}\left(\sqrt{R_s}\right) - \frac{\xi}{2}\sqrt{\frac{R_s}{\pi}}\exp\left(-R_s\right)\right]^2} \text{ (DTTL)} \tag{10-40}$$

where $\rho \triangleq P/N_0 B_L$ is the so-called phase-locked loop SNR. Figure 10-6 is a plot of the ratio of $\sigma_\lambda^2\big|_{\text{LDTTL}}$ to $\sigma_\lambda^2\big|_{\text{DTTL}}$ in dB as a function of R_s in dB

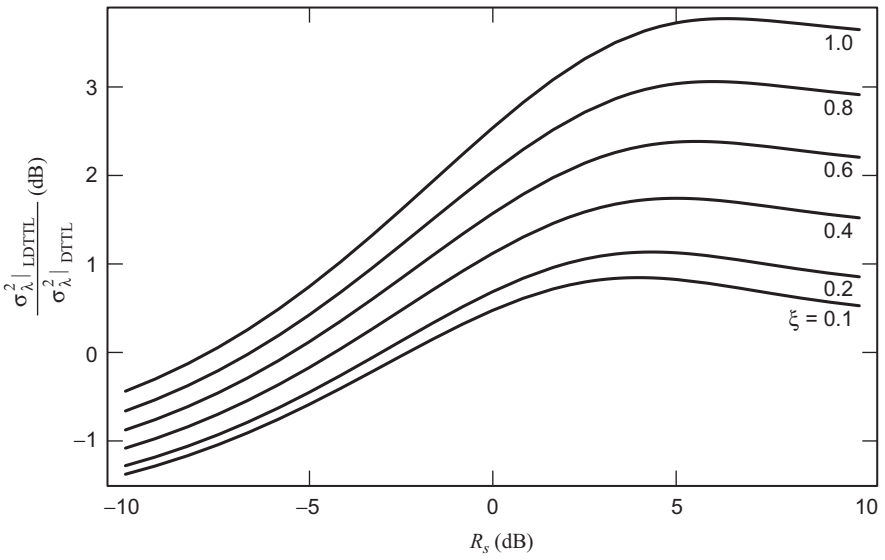


Fig. 10-6. Mean-squared jitter comparison of the nonlinear and linear DTTLs.

with quadrature arm normalized window width ξ as a parameter. The numerical results clearly illustrate the performance advantage of the LDTTL at low symbol SNRs. In fact, in the limit of sufficiently small SNR, the ratio of the variances approaches the limit

$$\lim_{R_s \to 0} \frac{\sigma_\lambda^2 \,|\, \text{LDTTL}}{\sigma_\lambda^2 \,|\, \text{DTTL}} = \frac{\left(2 - \frac{\xi}{2}\right)^2}{2\pi \left(1 - \frac{\xi}{4}\right)^2 \left(1 - \frac{\xi}{2\pi}\right)} \tag{10-41}$$

which for $\xi = 0$ (the theoretical value suggested by the MAP estimation of symbol sync approach) becomes

$$\lim_{\substack{R_s \to 0 \\ \xi \to 0}} \frac{\sigma_\lambda^2 \,|\, \text{LDTTL}}{\sigma_\lambda^2 \,|\, \text{DTTL}} = \frac{2}{\pi} \tag{10-42}$$

The fact that this ratio approaches a finite limit is not surprising in view of a similar behavior for other synchronization loops motivated by the MAP estimation approach. For example, when comparing the conventional Costas loop (motivated by the low SNR approximation to the MAP estimation of carrier phase) to the polarity-type Costas loop (motivated by the high SNR approximation to the MAP estimation of carrier phase), the ratio of variances of the phase error is given by (see Chapter 8)

$$\frac{\sigma_\phi^2 \,|_{\text{Conventional}}}{\sigma_\phi^2 \,|_{\text{Polarity-Type}}} = \frac{\text{erf}^2\left(\sqrt{R_s}\right)}{2R_s/\left(1 + 2R_s\right)} \tag{10-43}$$

which for sufficiently small SNR becomes

$$\lim_{R_s \to 0} \frac{\sigma_\phi^2 \,|_{\text{Conventional}}}{\sigma_\phi^2 \,|_{\text{Polarity-Type}}} = \frac{2}{\pi} \tag{10-44}$$

For large symbol SNR, the ratio of the variances in Eq. (10-40) approaches

$$\lim_{R_s \to \infty} \frac{\sigma_\lambda^2 \,|\, \text{LDTTL}}{\sigma_\lambda^2 \,|\, \text{DTTL}} = \frac{1 + \frac{\xi}{4}}{\left(1 - \frac{\xi}{4}\right)^2} \tag{10-45}$$

which for small window widths results in a small penalty for removing the I arm hard limiter.

10.4 Simplified MAP-Motivated Closed-Loop Symbol Synchronizers for *M*-PSK

In Section 10.1, we derived the form of the error signal [see Eq. (10-7)] for a MAP-motivated closed-loop symbol synchronizer of M-PSK, which is somewhat complicated at best. Applying a large argument (high SNR) approximation to the nonlinearities in the numerator and denominator of the expression in Eq. (10-7), namely, $\sinh x \cong (e^x/2)\operatorname{sgn} x$, $\cosh x \cong e^x/2$, unfortunately does not simplify matters because of the summation over the index (q) resulting from averaging over the signal constellation. In problems of this nature, it is common to approximate the summation by its largest term. In this particular case, it is most convenient to make this approximation in the CLF of Eq. (10-1) prior to taking its derivative to form the error signal in the MAP estimation loop. When this is done, we obtain for the MAP estimate (again setting $\theta_c = 0$)

$$\hat{\varepsilon}_{\text{MAP}} \cong \underset{\hat{\varepsilon}}{\operatorname{argmax}} \left[\sum_{n=0}^{N-1} \ln \left(\frac{2}{M} \max_{q} \left\{ \cosh \left[x_n \left(q; \hat{\varepsilon} \right) \right] \right\} \right) \right] \tag{10-46}$$

or, equivalently, because of the monotonicity of the hyperbolic cosine function,

$$\hat{\varepsilon}_{\text{MAP}} \cong \underset{\hat{\varepsilon}}{\operatorname{argmax}} \left[\sum_{n=0}^{N-1} \ln \left(\frac{2}{M} \cosh x_n \left(q_{\max}; \hat{\varepsilon} \right) \right) \right] \tag{10-47}$$

where

$$q_{\max} \triangleq \max_{q} \left\{ \left| x_n \left(q; \hat{\varepsilon} \right) \right| \right\} \tag{10-48}$$

Now differentiating Eq. (10-47) with respect to $\hat{\varepsilon}$, we obtain an expression for the error signal in a MAP-motivated symbol synchronizer for M-PSK at high SNR, namely,

$$e = \frac{d}{d\hat{\varepsilon}} \left[\sum_{n=0}^{N-1} \ln \left(\frac{2}{M} \cosh x_n \left(q_{\max}; \hat{\varepsilon} \right) \right) \right] = \sum_{n=0}^{N-1} \tanh \left[x_n \left(q_{\max}; \hat{\varepsilon} \right) \right] \frac{d}{d\hat{\varepsilon}} x_n \left(q_{\max}; \hat{\varepsilon} \right) \tag{10-49}$$

Note the similarity of Eq. (10-49) to Eq. (10-9). In fact, for binary phase-shift keying (BPSK), the only value of q in the sum is $q = 0$, and thus for this case $q_{\max} = 0$, which establishes the equivalence between Eq. (10-49) and Eq. (10-9). Because of this similarity, one can immediately apply the same small and large argument approximations to the hyperbolic tangent nonlinearity and, analogous to Figs. 10-3 and 10-4, arrive at DTTL-like implementations that are illustrated in Figs. 10-7 and 10-8.

10.5 MAP Sliding-Window Estimation of Symbol Timing

As discussed in Section 10.1, open-loop MAP estimation of the symbol epoch involves finding the conditional (on the symbol timing) likelihood function of the received signal based on a single observation of the received signal over a block of symbols. Furthermore, since the unknown symbol epoch is assumed to be uniformly distributed over the symbol interval, the MAP estimate is equivalent to the ML estimate. We have also seen that the traditional closed-loop estimation scheme motivated by the MAP estimation approach employs an error signal derived from the derivative of the CLF that can be updated at intervals corresponding to the symbol time. Since for rectangular pulses, e.g., an NRZ data stream, the derivative of the CLF, which is related to the derivative of the pulse shape, is undefined, closed-loop structures motivated by the MAP estimation approach strictly speaking do not exist. Nevertheless, with suitable approximations of the derivative of the pulse shape, such a closed loop, e.g., the DTTL, will in fact provide symbol sync for an NRZ data stream with rectangular pulses; however, it does so with a degradation in performance relative to that which can be provided by the MAP or minimum mean-squared (MMS) open-loop estimators. On the other hand, the closed-loop approach provides a continuous updating (tracking) of the symbol timing (once per bit interval) that is desirable in the presence of channel dynamics, whereas the open approach usually is regarded as either a one-shot estimator, i.e., compute the MAP or MMS estimate based on a single observed long block of data, or a block-by-block estimator where the single shot is sequentially repeated over and over.

What is important to observe is that the open-loop estimation techniques can be modified to provide sequential updates at the symbol rate to the symbol-timing epoch estimates and as such resemble the closed-loop techniques with, however, improved performance. It is this issue that we wish to discuss here, namely, a simple sequential digital implementation of the MAP estimation of symbol epoch that can track the dynamics in this parameter yet provide a performance approaching that of the true optimum MAP estimation technique.

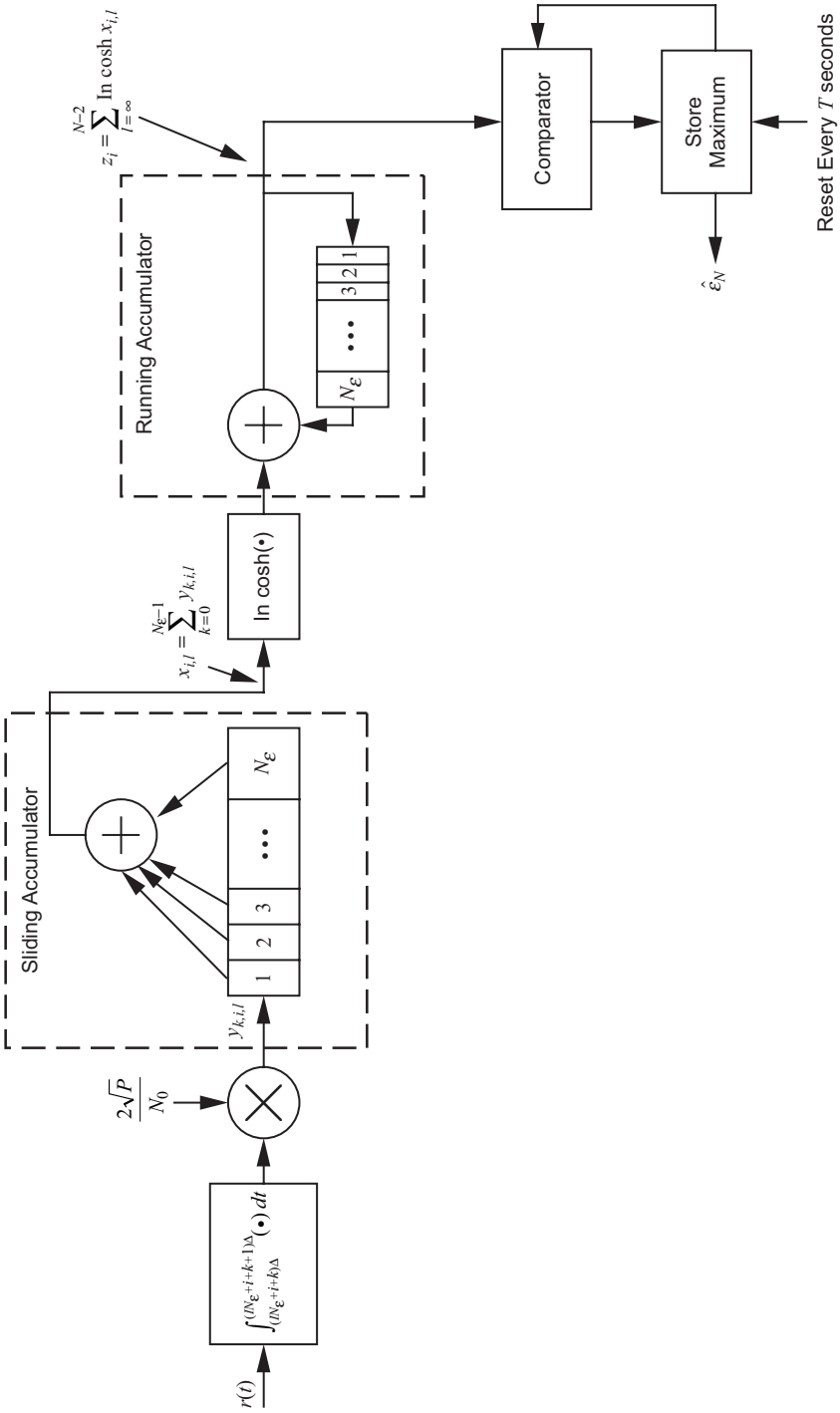


Fig. 10-7. A maximum-likelihood sliding-window estimator of timing for an NRZ data stream.

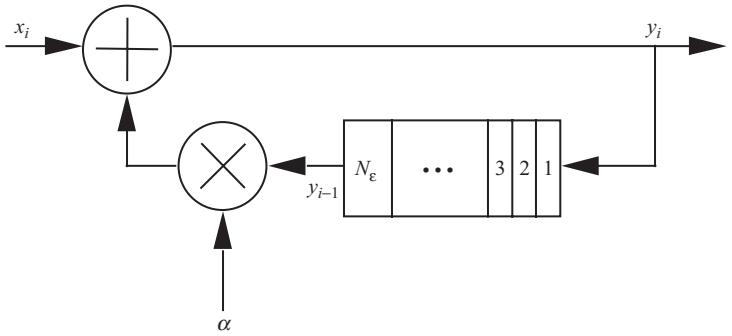


Fig. 10-8. A weighted running accumulator.

10.5.1 A Brief Discussion of Performance and Its Bounds for Open- and Closed-Loop Symbol-Timing Techniques

As we have noted in other chapters, an appropriate measure of the performance of an unbiased estimator of a parameter is its variance, which is equal to the mean-squared value of the error between the estimator and the parameter. Although it is usually difficult to arrive at an analytical expression for the variance of the MAP or MMS estimator, there exist many lower bounds on this quantity that can be evaluated analytically. The most popular of these bounds is the Cramer–Rao (C-R) bound [3–5] since it can be obtained directly from the conditional probability density function (pdf) of the received signal given the unknown epoch, which as shown earlier has the analytically desirable Gaussian form. In particular, the C-R bound on the variance of any unbiased estimator $\hat{\varepsilon}$ of ε is given by

$$\sigma_\varepsilon^2 \geq \left[E\left\{ \left(\frac{\partial \ln p\big(r(t)\,|\varepsilon\big)}{\partial \varepsilon} \right)^2 \right\} \right]^{-1} = -\left[E\left\{ \frac{\partial^2 \ln p\big(r(t)\,|\varepsilon\big)}{\partial \varepsilon^2} \right\} \right]^{-1} \triangleq \sigma_{C-R}^2 \tag{10-50}$$

In order to evaluate the derivative required in Eq. (10-50), it is necessary that the pulse shape be differentiable—the same condition as needed to form the MAP estimation loop. If the pulse shape is in fact differentiable and the other conditions for the C-R bound to exist [3–5] are satisfied, then for large SNR γ, the C-R bound varies inversely as the SNR, i.e., $\sigma_{C-R}^2 = C\gamma^{-1}$, where the constant of proportionality, C, depends on the particular pulse shape and its second derivative [5]. It is also true that, if the C-R bound is achievable, then clearly the MMS

estimator will be the one that achieves it, but so does the MAP estimator. That is, in this situation the MAP estimator is asymptotically (large SNR) efficient. On the other hand, if the C-R bound is not achievable, then by definition the more complex MMS estimator still will achieve the smallest estimator variance, and the MAP estimator may in fact result in a larger variance. That is, in this situation one cannot guarantee how close the MAP estimator comes to the C-R bound.

It can similarly be shown that closed loops motivated by the MAP approach, i.e., those that employ an error signal derived from the derivative of the LF, have a mean-squared timing error that varies inversely linearly with SNR. In the closed-loop case, one must make an appropriate adjustment to the term "SNR" to reflect the relation between the reciprocal of the loop bandwidth and the observation time of the open-loop estimate, analogously to what was done for the carrier synchronization case in Chapter 8. In particular, if the two-sided loop bandwidth is defined as equal to the reciprocal of the observation time (which is appropriate for a noise bandwidth definition), then the mean-squared timing error of the closed loop satisfies the C-R bound, i.e., $\sigma_\varepsilon^2 = \gamma^{-1}$.

When the pulse is not differentiable, such as the rectangular pulse that is characteristic of NRZ modulation, then as previously mentioned the C-R bound does not exist. One might consider trying to use the C-R bound in such situations by approximating the square pulse with a trapezoidal shape (which leads to derivatives at the edges that are rectangular pulses, as discussed in Section 10.2 for the DTTL), and then taking the limit as the slope of the edges approaches infinity. Unfortunately, when this is done the C-R bound becomes directly proportional to the inverse of the slope, and thus in the limit as the slope approaches infinity for any finite SNR, the bound degenerates to being useless, i.e., $\sigma_{C-R}^2 \to 0$.

To get around this enigma, researchers have investigated other bounds on the estimator variance that exist even when the pulse is non-differentiable. A variety of these bounds [6–9] are reviewed and compared in [10]. All of the results obtained in these references are for the case of a transmitted signal corresponding to either a single pulse, a periodic repetition of a single pulse, or a known sequence of pulses, such as a pseudo-noise (PN) code, and as such correspond to navigation, radar, and direct sequence spread spectrum system applications. For the case of data communication, where the transmitted waveform is a sequence of pulses with random (unknown) polarity, in order to make the results given in the above references applicable in this situation, one must draw an appropriate equivalence between the two scenarios. It is relatively straightforward to show that the C-R bound (which again does not apply in the square-pulse case) on the variance of the delay estimator for a random pulse stream of N symbols is equivalent to the C-R bound on the variance of a single pulse of N times the energy. Although establishing this equivalence is more formidable for the other bounds that do apply to the square-pulse case, we anticipate that a similar be-

havior will occur. Proceeding with this intuitive assumption, we now discuss the behavior of the various bounds considered in [10] as they would apply to the NRZ communication problem.

What is interesting about all the bounds in [6–9] is that, for rectangular pulses, they all predict (for large SNR) an inverse square-law behavior with SNR, i.e., $\sigma^2 \geq C\gamma^{-2}$, as opposed to the inverse linear behavior one might expect (at least from C-R bound considerations which granted do not apply here). The difference between the various bounds is the constant of proportionality, C. Monte Carlo simulations performed in [10] show that, in this situation, the MAP and MMS estimators have a similar inverse square-law behavior with SNR and come quite close to the tightest of the lower bounds. Thus, since the closed-loop schemes derived from the above-mentioned approximations to the MAP approach can achieve only inverse linear behavior with SNR, it behooves one to reexamine the possibility of using open-loop epoch estimation with the hope of obtaining a simple sequential structure that will allow for an improvement in performance as well as the ability to track variations in the parameter.

10.5.2 Formulation of the Sliding-Window Estimator

To arrive at the sliding-window version of the MAP estimator, we slightly modify the approach taken in Section 10.2 by assuming now that the observation of the received signal extends over the entire past up to the present time $t = NT$, and furthermore that the unknown parameter, ε, is constant over this observation, i.e., in the interval $-\infty \leq t \leq NT$.[4] In this case, the estimate at time NT becomes

$$\hat{\varepsilon}_N = \underset{\hat{\varepsilon}}{\operatorname{argmax}} \, \Lambda\left(\hat{\varepsilon}; NT\right) \tag{10-51}$$

where

$$\Lambda\left(\varepsilon; NT\right) \triangleq \sum_{l=-\infty}^{N-1} \ln \cosh \left(\frac{2\sqrt{P}}{N_0} \int_{-\infty}^{NT} r(t) p\left(t - lT - \varepsilon T\right) dt \right) \tag{10-52}$$

Ignoring the partial (less than a full symbol interval) contribution of the $N-1$st pulse $p\left(t - (N-1)T - \varepsilon T\right)$ [since the full contribution will be picked up in the LF for the next interval, namely, $\Lambda\left(\varepsilon; (N+1)T\right)$], we can rewrite Eq. (10-52) as

[4] Shortly we shall say more about how to tailor the results to the more practical case where the parameter is dynamic but slowly (with respect to the symbol duration) varying.

$$\Lambda\left(\varepsilon; NT\right) \triangleq \sum_{l=-\infty}^{N-2} \ln\cosh\left(\frac{2\sqrt{P}}{N_0}\int_{lT+\varepsilon}^{(l+1)T+\varepsilon} r(t)dt\right) \tag{10-53}$$

The LF as defined in Eq. (10-53) is a function of a parameter ε that takes on a continuum of values in the interval $0 \le t \le T$. In order to construct a practical implementation of Eq. (10-53), it is customary to quantize the uncertainty interval, i.e., approximate the continuous variable εT by a discrete variable that takes on values $\varepsilon_i T = iT/N_\varepsilon \triangleq i\Delta,\ i = 0, 1, 2, \cdots, N_\varepsilon - 1$. Since the quantization increment Δ determines the resolution to which the parameter can be estimated, the value of N_ε is chosen to satisfy this requirement. The time-quantized LF can now be written as

$$\Lambda\left(\varepsilon_i; NT\right) \triangleq \sum_{l=-\infty}^{N-2} \ln\cosh\left(\frac{2\sqrt{P}}{N_0}\int_{(lN_\varepsilon+i)\Delta}^{((l+1)N_\varepsilon+i)\Delta} r(t)dt\right),$$

$$i = 0, 1, 2, \cdots, N_\varepsilon - 1 \tag{10-54}$$

The integral in Eq. (10-54), which represents the integration of the received signal over the lth symbol interval (shifted by the epoch $\varepsilon_i T = i\Delta$), can be expressed as a sum of N_ε integrals over each quantization interval. In particular,

$$\Lambda\left(\varepsilon_i; NT\right) \triangleq \sum_{l=-\infty}^{N-2} \ln\cosh\left(\frac{2\sqrt{P}}{N_0}\sum_{k=0}^{N_\varepsilon-1}\int_{(lN_\varepsilon+i+k)\Delta}^{(lN_\varepsilon+i+k+1)\Delta} r(t)dt\right),$$

$$i = 0, 1, 2, \cdots, N_\varepsilon - 1 \tag{10-55}$$

Thus, the quantized MAP estimator of εT at time $t = NT$ is given by

$$\hat{\varepsilon}_N T = \hat{i}_N \Delta = \left[\underset{i}{\operatorname{argmax}}\, \Lambda\left(\hat{\varepsilon}_i; NT\right)\right]\Delta \tag{10-56}$$

The MAP estimate of symbol epoch at time $t = (N+1)\,T$ (i.e., one symbol time later) is given by

$$\hat{\varepsilon}_{N+1} T = \left[\underset{i}{\operatorname{argmax}}\, \Lambda\big(\hat{\varepsilon}_i; (N+1)\,T\big)\right]\Delta \tag{10-57}$$

where

$$\Lambda\left(\varepsilon_i;(N+1)\,T\right)=\sum_{l=-\infty}^{N-1}\ln\cosh\left(\frac{2\sqrt{P}}{N_0}\sum_{k=0}^{N_\varepsilon-1}\int_{(lN_\varepsilon+i+k)\Delta}^{(lN_\varepsilon+i+k+1)\Delta}r(t)dt\right)$$

$$=\Lambda\left(\varepsilon_i;NT\right)+\ln\cosh\left(\frac{2\sqrt{P}}{N_0}\sum_{k=0}^{N_\varepsilon-1}\int_{[(N-1)N_\varepsilon+i+k]\Delta}^{[(N-1)N_\varepsilon+i+k+1]\Delta}r(t)dt\right)\tag{10-58}$$

Thus, every T seconds (as in a closed-loop symbol synchronizer that updates its error signal every symbol time, e.g., the DTTL), we obtain an epoch estimate. An implementation of Eq. (10-56) that does not require a parallel bank of N_ε correlators as is traditional for an N_ε-quantized MAP parameter estimator is illustrated in Fig. 10-7. First, $\int_{(lN_\varepsilon+i+k)\Delta}^{(lN_\varepsilon+i+k+1)\Delta}r(t)dt$ is computed, which represents the integral of the received signal in the kth quantization (sample) interval of the lth symbol corresponding to the ith epoch position. Next, N_ε successive integrals, each scaled by $2\sqrt{P}/N_0$, are summed for each epoch position that, because of the recursive nature of the index i in Eq. (10-55), can be implemented by a sliding accumulator. That is, N_ε successive outputs of the sliding accumulator represent the argument of the "ln cosh" function in Eq. (10-55) for the N_ε epoch positions corresponding to the lth symbol. Next, we take the hyperbolic cosine of these outputs and pass them to a running accumulator (with delay equal to a symbol time or equivalently N_ε sample times.) Thus, in accordance with Eq. (10-58) each output of the running accumulator (which occurs every Δ seconds) is an accumulation of inputs spaced N_ε samples ($N\Delta=T$ seconds) apart. The output of this running accumulator in N_ε successive sampling intervals then is the quantized LF of Eq. (10-58) for the current symbol interval, namely, the Nth. The "comparator" and "store maximum" blocks then proceed to find the maximum of these N_ε likelihood values for the Nth symbol interval, after which the estimate is output. The "store maximum" block then is reset, and the procedure is repeated for the next (i.e., the N+1st) symbol interval. It is important that the "store maximum" block be reset in each symbol interval so that an erroneous symbol epoch in one symbol interval does not propagate to succeeding intervals, that is, the symbol epoch estimate for each symbol interval should be made from the maximum of the set of N_ε LF samples for that interval and not by comparison with the maximum of samples from any previous interval.

Because of the assumption that the unknown parameter being estimated, i.e., symbol epoch, is constant over the observation, the implementation in Fig. 10-7 includes a running accumulator with uniform weighting. In the more practical

case, where the parameter is dynamic but slowly varying, one can only assume that the unknown parameter is constant over a finite number of symbol intervals. As such, the uniform running accumulator should be replaced with a weighted running accumulator that reflects a fading memory and is analogous to what is done in a closed-loop architecture by using a digital filter following the error signal. The simplest method for accomplishing this is illustrated in Fig. 10-8, where the feedback term is multiplied by a constant $\alpha < 1$. This achieves a running accumulator with a geometric weighting that has the input–output characteristic

$$y_i = \sum_{m=0}^{\infty} \alpha^m x_{i-m} \tag{10-59}$$

Finally, since the running accumulator also accomplishes the data detection (matched-filter) function, then the epoch estimate index, $\hat{i}_N$, of Eq. (10-56) can be used to determine in each symbol interval which running accumulator output to use for making a hard decision on that symbol.

10.5.3 Extension to Other Pulse Shapes

When the pulse shape is other than rectangular, then, strictly speaking, the simplification that allows the bank of N_ε correlators to be replaced by a sliding accumulator as in Fig. 10-7 is not possible. However, if N_ε is large and the pulse shape is approximated by a piecewise constant staircase function with N_ε steps, then the correlation of the received signal and the pulse shape in a quantization interval can be written as

$$\int_{i\Delta}^{(i+1)\Delta} r(t)p(t)dt = p_i \int_{i\Delta}^{(i+1)\Delta} r(t)dt \tag{10-60}$$

where p_i is the assumed constant value of $p\,(t)$ in the interval $i\Delta \le t \le (i+1)\,\Delta$. In view of Eq. (10-60), the only modification of Fig. 10-7 that is necessary to allow for the inclusion of an arbitrary pulse shape is to replace the uniform sliding accumulator with a *weighted* sliding accumulator (see Fig. 10-9), where the weights are equal to the piecewise constant values of $p(t)$. Furthermore, for sufficiently large N_ε, one can approximately replace the integral of $r(t)$ over the quantization interval by the value of $r(t)$ at the midpoint of this interval times the duration of this interval, Δ. As such, the integrator at the input of Fig. 10-7 can be replaced simply by a uniform sampler at rate $1/\Delta$.

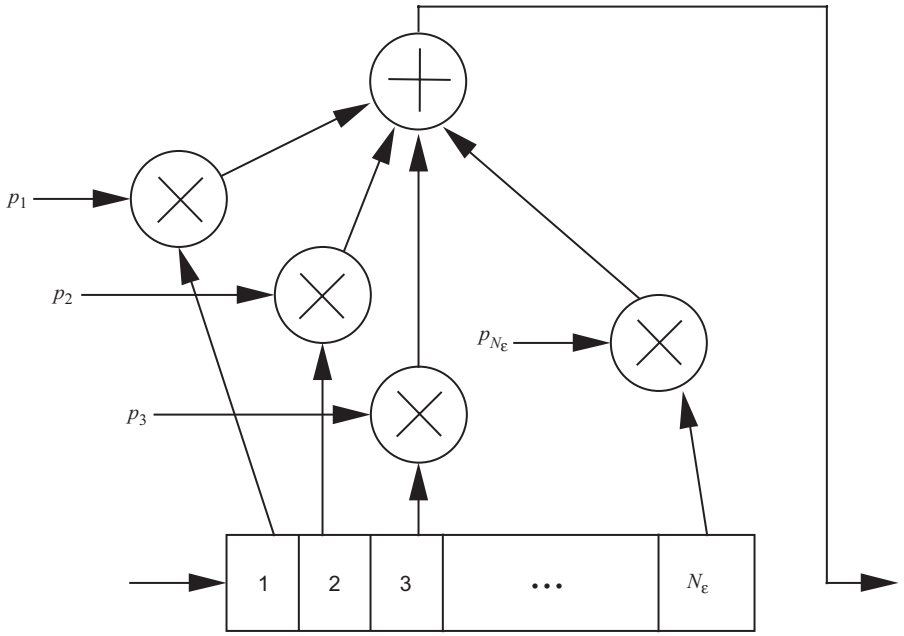


Fig. 10-9. A weighted sliding-window accumulator.

10.6 Symbol Synchronization in the Absence of Carrier Phase Information

10.6.1 Suboptimum Schemes

In addition to "optimum" symbol synchronizers, such as the ones discussed thus far that are motivated by the MAP estimation approach, several other suboptimum schemes have been proposed in the literature that offer the advantage of a simpler implementation and at the same time perform nearly as well as the more complex optimum ones. One of the more popular of these ad hoc schemes that draws its roots from the squaring loop used for carrier synchronization is called the "filter and square symbol synchronizer," whose tracking performance was analyzed in [11] for the case of an NRZ input and a single-pole Butterworth low-pass filter for $H(s)$. A block diagram of this synchronizer is provided in Fig. 10-10. The operation of this scheme is briefly summarized as follows.

For a binary NRZ input described by $s(t,\varepsilon)=\sqrt{P}\sum_{n=-\infty}^{\infty} d_n p(t-nT-\varepsilon T)$, the output of the filter is given as $\hat{s}(t,\varepsilon) = \sqrt{P}\sum_{n=-\infty}^{\infty} d_n \hat{p}(t-nT-\varepsilon T)$, where[5]

[5] Here, the hat on $s(t,\varepsilon)$ and $p(t)$ is simply used to denote the result of low-pass filtering by $H(s)$.

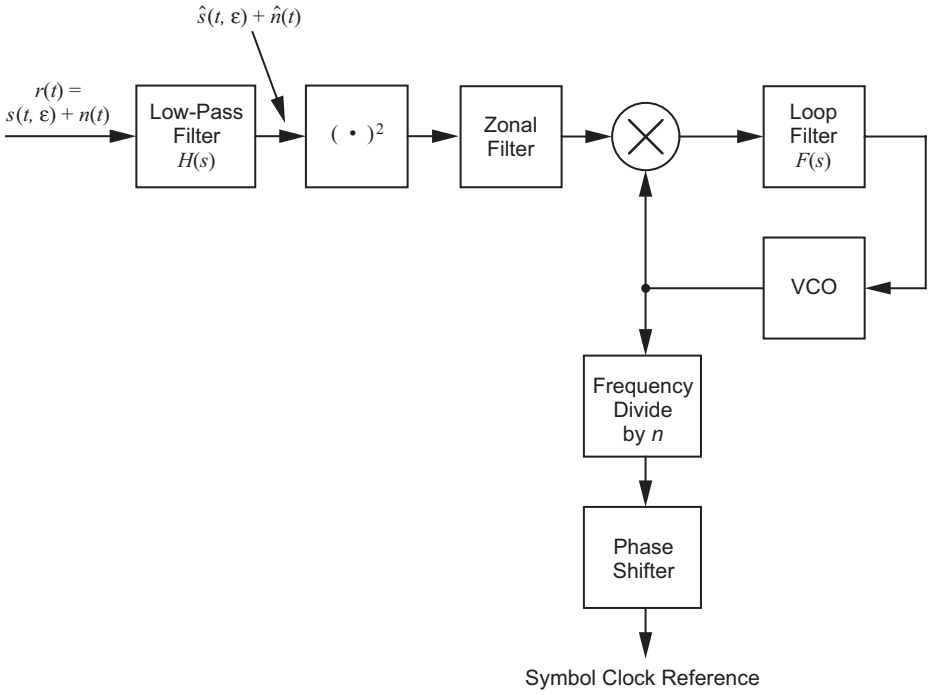


Fig. 10-10. Filter and square symbol synchronizer.

$$\hat{p}(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} P(\omega) H(\omega) d\omega \tag{10-61}$$

with $P(\omega)$ the Fourier transform of the NRZ rectangular pulse $p(t)$. Squaring $\bar{s}(t, \varepsilon)$ results in

$$\hat{s}^2(t, \varepsilon) = P \sum_{n=-\infty}^{\infty} \hat{p}^2(t - nT - \varepsilon T) + P \sum_{m=-\infty}^{\infty} \sum_{n=-\infty}^{\infty} d_m d_n \hat{p}(t - mT - \varepsilon T) \hat{p}(t - nT - \varepsilon T) \tag{10-62}$$

which after ensemble averaging over the random data becomes

$$\overline{\hat{s}^2(t, \varepsilon)} = P \sum_{n=-\infty}^{\infty} \hat{p}^2(t - nT - \varepsilon T) \tag{10-63}$$

The term $\sum_{n=-\infty}^{\infty} \hat{p}^2 (t - nT - \varepsilon T)$ is periodic with fundamental period equal to T and thus possesses a line spectrum with harmonics that are multiples of the data rate, each of which carries along the symbol-timing information. Thus, following this signal with a zonal filter (to extract, say, the nth harmonic), a sinusoidal tone is generated at $f = n/T$ that can be tracked by a phase-locked loop (PLL) whose voltage-controlled oscillator (VCO) output after frequency division by n and an appropriate phase shift[6] represents a symbol-timing clock that is synchronous with the input data stream.

Shortly thereafter [12], a generalization of the filter and square symbol synchronizer was proposed in which the square-law device was replaced by a delay-and-multiply operation (see Fig. 10-11). The resulting configuration, referred to as a "cross-spectrum symbol synchronizer (CSSS)," allowed in general for a delay element equal to a fraction α of the symbol time, where the value of α would be chosen to optimize the tracking performance in the sense of minimizing the mean-squared timing error. It is clear from a comparison of Figs. 10-10 and 10-11 that the filter and square-law symbol synchronizer is a special case of the cross-spectrum symbol synchronizer corresponding to $\alpha = 0$. Once again assuming a

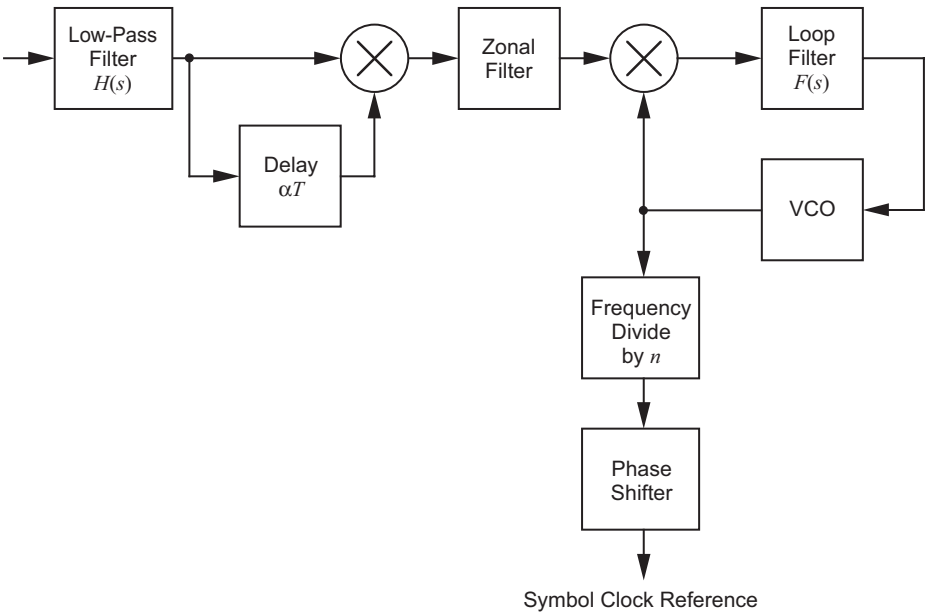


Fig. 10-11. Cross-spectrum symbol synchronizer.

[6] The phase shifter is required to cancel the known phase shift inherent in the nth harmonic of the Fourier series representation of the signal component in the output of the squaring device.

single-pole Butterworth low-pass filter for $H(s)$, the line spectrum at the output of the delay-and-multiply operation was analyzed in [12] as a function of the fractional delay α for both low and high SNRs. In particular, for a given value of SNR and α, it was shown that there exists an optimum filter bandwidth-to-data rate ratio[7] in the sense of minimizing the mean-squared timing error and that the optimum value of α in each case was equal to $1/2$. Furthermore, in addition to $\alpha = 1/2$ optimizing the performance for the best choice of filter bandwidth-to-data rate ratio, it also resulted in a significant improvement in robustness with regard to variations in this ratio.

Although the filter and square symbol synchronizer and its generalization, the cross-spectrum symbol synchronizer, were initially proposed as real baseband schemes that implicitly assumed perfect carrier synchronization, it is straightforward to modify them so as to be useful in a noncoherent carrier phase environment. Specifically, if we now model the signal component of the input in complex form as

$$\tilde{s}(t,\varepsilon) = \sqrt{2P}e^{j\theta_c}\sum_{n=-\infty}^{\infty} d_n p(t - nT - \varepsilon T) \tag{10-64}$$

where θ_c denotes the unknown carrier phase, then performing the delay-and-multiply function in complex conjugate form again will result in a zonal filter output that is a tone at the nth harmonic of the data rate that can be tracked by a PLL. Furthermore, the performance of this scheme will be independent of the value of θ_c. A block diagram of the real noncoherent version of the cross-spectrum synchronizer is illustrated in Fig. 10-12, where the input is now the *bandpass* received signal whose signal component is given by $s(t,\varepsilon) = \mathrm{Re}\left\{\tilde{s}(t,\varepsilon)e^{j\omega_c t}\right\}$ with ω_c denoting the carrier frequency. In what follows, we present the tracking performance of the symbol synchronizer in Fig. 10-12, drawing heavily on the detailed results already contained in [11] and [12].

In accordance with the above, the received bandpass signal is given by

$$\begin{aligned} r(t) &= s(t,\varepsilon) + n(t) \\ &= \sqrt{2P}m(t)\cos(\omega_c t + \theta_c) + \sqrt{2}\left[n_c(t)\cos\omega_c t - n_c(t)\sin\omega_c t\right] \end{aligned} \tag{10-65}$$

where $n_c(t), n_s(t)$ are independent low-pass Gaussian noise processes with single-sided power spectral density N_0 W/Hz. After demodulation with quadrature reference signals

[7] This phenomenon is entirely synergistic with the tracking performance of the Costas or squaring loop as exemplified by its squaring-loss behavior as a function of the ratio of arm filter bandwidth to data rate (see Chapter 8).

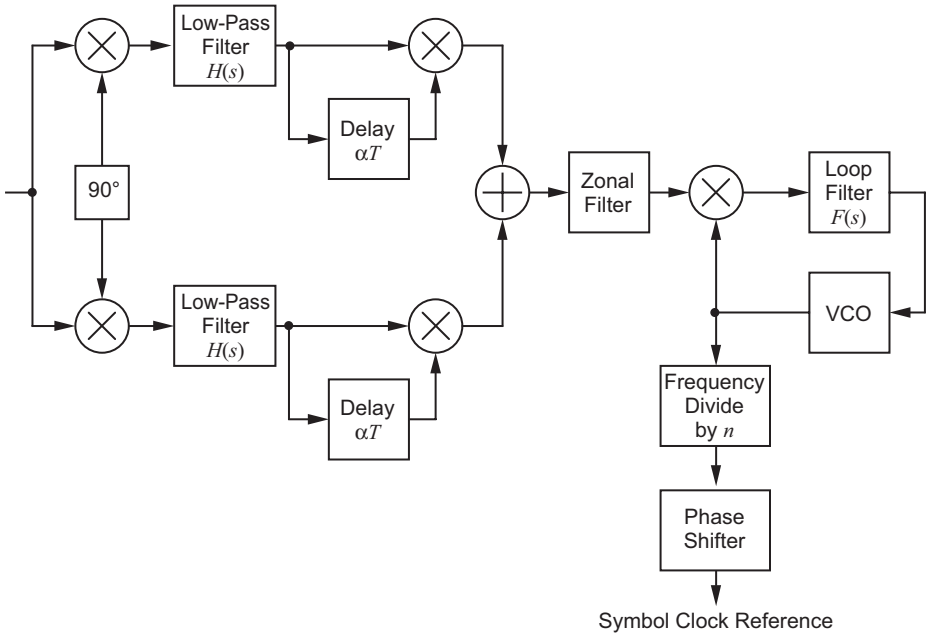


Fig. 10-12. Noncoherent cross-spectrum symbol synchronizer.

$$r_c(t) = \sqrt{2}\cos\omega_c t$$
$$r_s(t) = -\sqrt{2}\sin\omega_c t \tag{10-66}$$

and then filtering and delay-and-multiplying, we obtain the I and Q low-pass signals

$$\begin{aligned}\hat{x}_c(t) &= P\hat{m}(t)\,\hat{m}(t-\alpha T)\cos^2\theta_c + \hat{n}_c(t)\,\hat{n}_c(t-\alpha T)\\ &\quad + \sqrt{P}\cos\theta_c\left[\hat{m}(t)\,\hat{n}_c(t-\alpha T) + \hat{m}(t-\alpha T)\,\hat{n}_c(t)\right]\\ \hat{x}_s(t) &= P\hat{m}(t)\,\hat{m}(t-\alpha T)\sin^2\theta_c + \hat{n}_s(t)\,\hat{n}_s(t-\alpha T)\\ &\quad + \sqrt{P}\sin\theta_c\left[\hat{m}(t)\,\hat{n}_s(t-\alpha T) + \hat{m}(t-\alpha T)\,\hat{n}_s(t)\right]\end{aligned} \tag{10-67}$$

Summing these I and Q signals produces

$$x(t) = P\hat{m}(t)\hat{m}(t-\alpha T) + \hat{n}_c(t)\hat{n}_c(t-\alpha T) + \hat{n}_s(t)\hat{n}_s(t-\alpha T)$$

$$+ \sqrt{P}\hat{m}(t)\left[\hat{n}_c(t-\alpha T)\cos\theta_c + \hat{n}_s(t-\alpha T)\sin\theta_c\right]$$

$$+ \sqrt{P}\hat{m}\,(t-\alpha T)\left[\hat{n}_c(t)\cos\theta_c + \hat{n}_s(t)\sin\theta_c\right] \tag{10-68}$$

whose signal × signal ($S \times S$) component [the first term on the right-hand side of Eq. (10-68)] is identical to that of the phase coherent cross-spectral symbol synchronizer and as such is independent of the carrier phase. It now remains to investigate to what extent the noise × noise ($N \times N$) component [the second and third terms on the right-hand side of Eq. (10-68)] and the signal × noise ($S \times N$) component [the fourth and fifth terms on the right-hand side of Eq. (10-68)] have changed and what impact these changes have on the tracking performance of the loop.

As is typical of all synchronization loops of this type, the tracking performance as measured by the mean-squared timing error can be characterized by the "squaring loss," which represents the degradation[8] in this measure due to the nonlinear nature ($S \times S$, $S \times N$, and $N \times N$ operations) of the loop. Specifically, the squaring loss is formed from a scaled version of the ratio of the power in the $S \times S$ component to the equivalent noise power spectral density of the sum of the $S \times N$ and $N \times N$ components, all evaluated at the nth harmonic of the data rate. As we shall see shortly, it will not be necessary to redo the evaluations of these component contributions to the squaring loss from what was done in [11] and [12] for the phase-coherent symbol synchronizer. Rather, we shall simply be able to make direct use of the evaluations found there with simple or no modification at all. As such the evaluation of the squaring loss itself will follow immediately almost by inspection.

To evaluate the equivalent noise power spectral densities of the $S \times N$ and $N \times N$ components, namely, $N'_{0_{S\times N}}$ and $N'_{0_{N\times N}}$, respectively, we must first com-

[8] As we shall see shortly, the squaring loss can at times exceed 0 dB and thus, in reality, can represent a gain rather than a loss. The reason for using such a nomenclature here nonetheless is by analogy with its usage in the carrier sync application, where it represents the additional degradation of the mean-squared *phase* error relative to that of a linear carrier tracking loop such as a PLL, and hence its value there can never exceed 0 dB. The difference between the two usages is centered around the fact that in the carrier sync application the phase error can vary over a range of 2π rad, whereas in the symbol sync application the normalized (to the T-second symbol duration) timing error can vary over a range of unity. Thus, there is a scale factor of $(2\pi)^2$ that comes into play when relating the mean-squared phase error of the sinusoidal clock supplied by the PLL portion of the cross-spectrum symbol synchronizer to the mean-squared normalized timing error of this same reference when used as a symbol sync clock. The important point to keep in mind is that the squaring loss is just a relative measure of performance and thus is useful in comparing different sync schemes.

pute their autocorrelation function. The autocorrelation of the $S \times N$ component is by definition

$$\begin{aligned}
R_{sn}(\tau) =& P\,E\left\{\left[\hat{m}(t)\left[\hat{n}_c(t-\alpha T)\cos\theta_c + \hat{n}_s(t-\alpha T)\sin\theta_c\right]\right.\right. \\
&+\hat{m}(t-\alpha T)\left[\hat{n}_c(t)\cos\theta_c + \hat{n}_s(t)\sin\theta_c\right]\Big] \\
&\times\left[\hat{m}(t+\tau)\left[\hat{n}_c(t-\alpha T+\tau)\cos\theta_c + \hat{n}_s(t-\alpha T+\tau)\sin\theta_c\right]\right. \\
&\left.\left.+\hat{m}(t-\alpha T+\tau)\left[\hat{n}_c(t+\tau)\cos\theta_c + \hat{n}_s(t+\tau)\sin\theta_c\right]\right]\right\} \\
=& 2PR_{\hat{m}}(\tau)\left[R_{\hat{n}_c}(\tau)\cos^2\theta_c + R_{\hat{n}_s}(\tau)\sin^2\theta_c\right] \\
&+ PR_{\hat{m}}(\tau+\alpha T)\left[R_{\hat{n}_c}(\tau-\alpha T)\cos^2\theta_c + R_{\hat{n}_s}(\tau-\alpha T)\sin^2\theta_c\right] \\
&+ PR_{\hat{m}}(\tau-\alpha T)\left[R_{\hat{n}_c}(\tau+\alpha T)\cos^2\theta_c + R_{\hat{n}_s}(\tau+\alpha T)\sin^2\theta_c\right] \qquad (10\text{-}69)
\end{aligned}$$

which after recognizing that $R_{\hat{n}_c}(\tau) = R_{\hat{n}_s}(\tau) = R_{\hat{n}}(\tau)$ simplifies to

$$\begin{aligned}
R_{sn}(\tau) =& P\left[2R_{\hat{m}}(\tau)R_{\hat{n}}(\tau) + R_{\hat{m}}(\tau+\alpha T)R_{\hat{n}}(\tau-\alpha T)\right. \\
&\left.+R_{\hat{m}}(\tau-\alpha T)R_{\hat{n}}(\tau+\alpha T)\right] \qquad (10\text{-}70)
\end{aligned}$$

Again it can be observed that the autocorrelation in Eq. (10-8) is independent of the carrier phase θ_c and furthermore is identical to the analogous result for the phase-coherent cross-spectrum symbol synchronizer as given in Eq. (10) of [12].[9]

Next, the autocorrelation of the $N \times N$ component is obtained as

$$\begin{aligned}
R_{nn}(\tau) =& E\left\{\left[\hat{n}_c(t)\hat{n}_c(t-\alpha T) + \hat{n}_s(t)\hat{n}_s(t-\alpha T)\right]\right. \\
&\left.\times\left[\hat{n}_c(t+\tau)\hat{n}_c(t-\alpha T+\tau) + \hat{n}_s(t+\tau)\hat{n}_s(t-\alpha T+\tau)\right]\right\} \\
=& 2\left[R_{\hat{n}}^2(\alpha T) + R_{\hat{n}}^2(\tau) + R_{\hat{n}}(\tau-\alpha T)R_{\hat{n}}(\tau+\alpha T)\right] \qquad (10\text{-}71)
\end{aligned}$$

[9] Note that the multiplicative factor P has been included here in the definition of $R_{sn}(\tau)$ whereas in [12], where P is denoted by S, it has been erroneously omitted in defining the total noise power spectral density.

which is exactly twice the analogous result for the phase-coherent cross-spectrum symbol synchronizer as given in [12]. Thus, since the equivalent noise power spectral densities are computed from the Fourier transforms of the autocorrelations evaluated at the nth harmonic of the data rate, i.e., $N'_{0_{S\times N}} = 2\int_{-\infty}^{\infty} R_{sn}(\tau)\, e^{j2\pi n\tau/T} d\tau$ and $N'_{0_{N\times N}} = 2\int_{-\infty}^{\infty} R_{nn}(\tau)\, e^{j2\pi n\tau/T} d\tau$, then ignoring the zero frequency term $R_n^2(\alpha T)$ as was done in [12] (since it leads to a power spectral line component at the zeroth harmonic of the data rate which is eliminated by the zonal filter), we conclude that

$$\begin{aligned} N'_{0_{S\times N}}|_{noncoh.} &= N'_{0_{S\times N}}|_{coh.} \\ N'_{0_{N\times N}}|_{noncoh.} &= 2N'_{0_{N\times N}}|_{coh.} \end{aligned} \tag{10-72}$$

Finally, since, as previously stated, the $S \times S$ component of the noncoherent cross-spectral symbol synchronizer is identical to that of the phase coherent one, then letting $|C_n|^2$ denote the normalized power in this component at the nth harmonic of the data rate, the squaring loss of the former is obtained as (see Eq. (48) of [12] with minor corrections applied)

$$\begin{aligned} S_L|_{\text{noncoh.}} &= (2\pi n)^2 PN_0 \left[\frac{2|C_n|^2\Big|_{\text{noncoh.}}}{N'_{0_{S\times N}}\Big|_{\text{noncoh.}} + N'_{0_{N\times N}}\Big|_{\text{noncoh.}}} \right] \\ &= (2\pi n)^2 PN_0 \left[\frac{2|C_n|^2\Big|_{\text{coh.}}}{N'_{0_{S\times N}}\Big|_{\text{coh.}} + 2N'_{0_{N\times N}}\Big|_{\text{coh.}}} \right] \end{aligned} \tag{10-73}$$

At this point, it is straightforward to evaluate Eq. (10-73) by making use of the expressions in [12] for $|C_n|^2\big|_{\text{coh.}}$, $N'_{0_{S\times N}}\big|_{\text{coh.}}$, and $N'_{0_{N\times N}}\big|_{coh.}$. A summary of these results for the special case of a single-pole Butterworth low-pass filter for $H(s)$ (with 3-dB cutoff frequency f_c), random (transition density equal to 0.5) NRZ data, $n = 1$ (tracking of the first harmonic), and either $\alpha = 0$ (the filter and square-law implementation) or $\alpha = 0.5$ (a half-symbol delay that was shown in [12] to be optimum in the sense of minimizing the squaring loss at the best ratio of low-pass filter bandwidth to symbol time) is given in the following.[10]

[10] These results were not explicitly given in [12] but have been independently derived here after considerable manipulation and integral evaluation.

For $\alpha = 0$:

$$|C_1|^2 = \frac{\left[1 - \exp(-2\pi R)\right]^2}{(2\pi R)^2} \frac{1}{[1 + 1/R^2][1 + 1/4R^2]}$$

$$N'_{0_{S\times N}}\Big|_{\text{coh.}} = \frac{4PN_0}{1 + 1/R^2}\left\{1 - \frac{1 - \exp(-2\pi R)}{8\pi R}\left[\frac{6 + 1/R^2 + 1/R^4}{[1 + 1/R^2][1 + 1/4R^2]}\right]\right\} \quad (10\text{-}74)$$

$$N'_{0_{N\times N}}\Big|_{\text{coh.}} = \frac{PN_0}{1 + 1/4R^2}\left(\frac{\pi R}{2E_s/N_0}\right)$$

For $\alpha = 0.5$:

$$|C_1|^2 = \frac{1}{(2\pi)^2[1 + 1/R^2][1 + 1/4R^2]} \times \left\{\frac{\left[\exp(-\pi R)\left[3 - \exp(-2\pi R)\right] - 2\right]^2}{4R^2} + 4\right\}$$

$$N'_{0_{S\times N}}\Big|_{\text{coh.}} = \frac{2PN_0}{1 + 1/R^2}\left\{1 - \frac{1}{4\pi R}\left[\frac{1 - 1/R^2}{1 + 1/R^2} + \frac{1}{2}\left(\frac{1 + 1/R^2}{1 + 1/4R^2}\right)\right]\left[3 - 4\exp(-2\pi R) + \exp(-4\pi R)\right]\right\} \quad (10\text{-}75)$$

$$N'_{0_{N\times N}}\Big|_{\text{coh.}} = \frac{PN_0}{1 + 1/4R^2}\left(\frac{\pi R}{4E_s/N_0}\right)\left[1 - \exp(-2\pi R)\right]$$

where $R \triangleq f_c T$ and $E_s = PT$ is the symbol energy.

Figure 10-13 is an illustration of $S_L\big|_{\text{noncoh.}}$ as computed from Eq. (10-73) together with Eq. (10-74) or Eq. (10-75) versus R with E_s/N_0 as a parameter. Also shown in dashed lines are the corresponding plots of the squaring-loss performance for the coherent cross-spectrum symbol synchronizer, namely $S_L\big|_{\text{coh.}}$, as previously obtained in [12] or equivalently from Eq. (10-73) by ignoring the factor of two in front of $N'_{0_{N\times N}}\big|_{\text{coh.}}$. We observe that the noncoherent symbol synchronizer performs almost as well as the coherent one at high SNR (where the $S \times N$ noise dominates over the $N \times N$ noise), whereas at low SNR (where the $N \times N$ noise dominates over the $S \times N$ noise) there is a more significant degradation of the former relative to the latter. Next, as was the case for the coherent symbol synchronizers, the noncoherent cross-spectrum scheme with

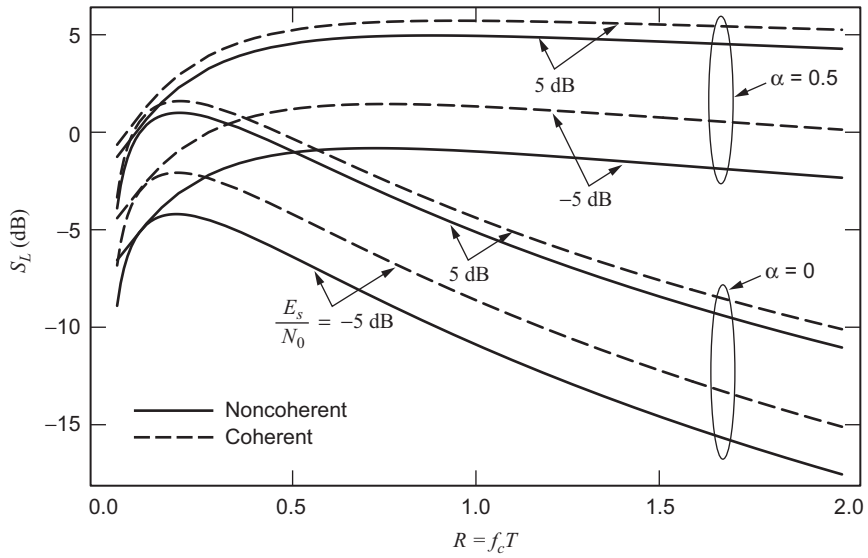


Fig. 10-13. Squaring-loss performance of noncoherent and coherent cross-spectrum symbol synchronizers.

half-symbol delay provides an improvement in performance over the filter and square-law scheme when implemented with the optimum value of bandwidth–time product R. Furthermore, although the cross-spectrum schemes exhibit a dependence on the bandwidth–time product for all values of α, this dependence is considerably reduced by the use of a half-symbol delay, particularly when compared with that for $\alpha = 0$.

To explain the much slower roll-off of the squaring loss performance with R for the half-symbol delay case, we reason as follows. In the limit of large low-pass filter bandwidth (theoretically no filtering at all), when $\alpha = 0$ the signal component of the output of the delay-and-multiply circuit (equivalent to a squaring operation in this case) is a squared NRZ waveform which simply is a constant equal to unity and as such does not contain a harmonic at $1/T$. This is born out by the fact that the normalized signal power of the harmonic at $1/T$ as given by $|C_1|^2$ in Eq. (10-74) is equal to zero in the limit of $R \to \infty$. On the other hand, in the same limit with $\alpha = 0.5$, the output of the delay-and-multiply circuit randomly alternates between a ± 1 square wave at the data rate and a $+1$ constant. The average of these two waveforms is a unipolar $(0, 1)$ square wave at the data rate whose Fourier series expansion clearly contains a nonzero harmonic at $1/T$. Once again this is born out by the fact that, using Eq. (10-75), in the limit of $R \to \infty$ and $E_s/N_0 \to \infty$ we have $|C_1|^2 = 1/\pi^2$. Since, for large R, the $N'_{0_{N\times N}}\big|_{\text{coh.}}$ term dominates over the $N'_{0_{S\times N}}\big|_{\text{coh.}}$ and since for $\alpha = 0$ and $\alpha = 0.5$ they both have the same behavior (except for a factor of two smaller

for the latter), then when taking the ratio of $|C_1|^2$ to the sum of $N'_{0_{S\times N}}\big|_{\text{coh.}}$ and $N'_{0_{N\times N}}\big|_{\text{coh.}}$, the squaring loss for the half-symbol delay case will decay with R much less rapidly than for the zero-delay (squaring) case.

It is now of interest to compare the performance of the noncoherent cross-spectrum symbol synchronizers to that of the coherent DTTL whose squaring loss is obtained from Eq. (10-40) using the relation $\sigma_\lambda^2 = 1/\rho S_L$, i.e.,

$$S_L = \frac{2\left[\text{erf}\left(\sqrt{\frac{E_s}{N_0}}\right) - \frac{\xi}{2}\sqrt{\frac{E_s/N_0}{\pi}}\exp\left(-\frac{E_s}{N_0}\right)\right]^2}{\xi\left\{1 + \frac{\xi}{2}\left(\frac{E_s}{N_0}\right) - \frac{\xi}{2}\left[\frac{1}{\sqrt{\pi}}\exp\left(-\frac{E_s}{N_0}\right) + \sqrt{\frac{E_s}{N_0}}\,\text{erf}\left(\sqrt{\frac{E_s}{N_0}}\right)\right]^2\right\}} \tag{10-76}$$

Figures 10-14(a) and 10-14(b) are plots of the squaring loss given by Eq. (10-76) versus E_s/N_0 in dB and, for comparison, the optimum (with respect to choice of R) squaring loss for the coherent and noncoherent cross-spectrum schemes corresponding to $\alpha = 0$ and $\alpha = 0.5$, respectively. In the case of Fig. 10-14(a), we observe that, regardless of its window width, the DTTL outperforms the noncoherent cross-spectrum (filter and square) scheme over the entire range of SNR illustrated. On the other hand, when compared to the coherent cross-spectrum scheme, for sufficiently large window width, the DTTL performance will suffer a degradation at low values of SNR. This should not be surprising since, as mentioned earlier in the chapter, the DTTL is derived from a high SNR approximation to the MAP symbol synchronizer which itself is motivated by the MAP estimation approach only in the limit of infinitesimally small window width.[11] With reference to Fig. 10-14(b), we observe that the performance of the coherent cross-spectrum scheme is quite competitive with that of the DTTL having a window width $\xi = 0.5$, and even the noncoherent cross-spectrum scheme can slightly outperform this DTTL at high SNR. As the window width is increased beyond a value of one-half, the cross-spectrum symbol sync schemes will clearly outperform the DTTL over the entire range of SNRs.

[11] The window width, ξ, of the DTTL corresponds to the approximation of the derivative of an NRZ pulse at a transition point in the data stream, namely, a delta function, with a finite-width rectangular pulse. Thus, the validity of the approximation, as well as the tracking performance of the closed-loop DTTL, monotonically improves as the window width becomes smaller and smaller. However, while in principle the MAP approach suggests an infinitesimally small window width, in practice there is a lower limit on its value since the width of the tracking region is directly proportional to ξ. Thus, if the window width is made too small, the ability of the loop to remain in lock will severely diminish. The choice of window width is determined by the condition $\sigma_\lambda \ll \xi$.

While it is difficult analytically to obtain the limiting behavior of the cross-spectrum schemes when E_s/N_0 approaches infinity, it can be shown numerically that, for both the noncoherent and coherent versions, the optimum value of R is approximately equal to 1.1, and the accompanying value of squaring loss is 6.84 dB.

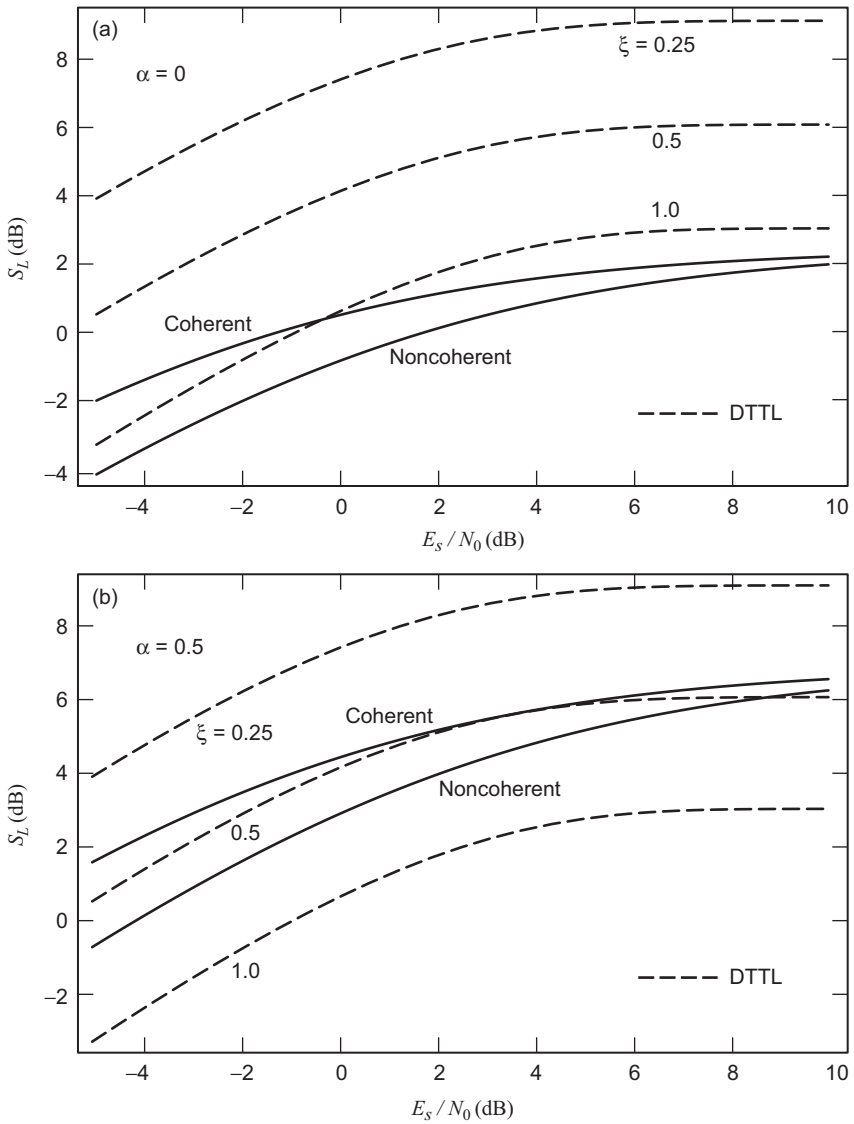


Fig. 10-14. A comparison of the squaring-loss performance of noncoherent and coherent cross-spectrum symbol synchronizers with that of the DTTL: (a) $\alpha = 0$ and (b) $\alpha = 0.5$.

10.6.2 The Noncoherent DTTL

In this section, we return to the ML approach for obtaining a symbol synchronizer in the absence of carrier phase information with particular emphasis on the necessary modifications of the conventional (coherent) DTTL structures as treated earlier in this chapter. We shall see that, in the low SNR region, the modification resembles that found for the suboptimum schemes discussed in the previous subsection, i.e., the independent addition of the symbol sync component derived from the quadrature carrier arm, whereas for the high SNR region, the structure involves a nonlinear cross-coupling of symbol sync components from both the in-phase and quadrature carrier arms. Wherever possible, results will be obtained from a combination of theory and simulation. Before proceeding, it should be mentioned that the MAP approach to symbol sync was considered in [13] in the context of arriving at a non-data-aided recursive algorithm for symbol timing. Although at first glance it might appear that the approach taken there corresponds to noncoherent symbol sync since the carrier phase was assumed to be unknown but independent from symbol to symbol,[12] in reality the derivation of the MAP estimate of symbol sync was preceded by a recursive estimate of the carrier phase which justifies such an assumption. Our emphasis here, as mentioned above, is on interpreting the likelihood function derived from such an approach in such a way as to arrive at noncoherent versions of the DTTL. For the sake of brevity and consistent with the original derivation of the coherent DTTL, we shall focus only on the BPSK ($M = 2$), NRZ case.

10.6.2.1 MAP Symbol Sync Estimation in the Absence of Carrier Phase Information. The input to the receiver is a bandpass signal modeled by the combination of Eq. (10-65) together with Eq. (10-64). The first step is to demodulate the received signal with the quadrature carrier reference signals

$$\begin{aligned} r_c(t) &= \sqrt{2}\cos\omega_c t \\ r_s(t) &= -\sqrt{2}\sin\omega_c t \end{aligned} \tag{10-77}$$

resulting in the pair of baseband observables in the nth symbol interval $(n+\varepsilon)T \leq t \leq (n+1+\varepsilon)T$

[12] As we shall see shortly, the appropriate assumption for truly noncoherent symbol sync is an unknown carrier phase that is constant over the duration of the observation, i.e., a sequence of symbols.

$$\begin{aligned} x_{cn}(t) &= \sqrt{P} d_n p\big(t-(n+\varepsilon)T\big)\cos\theta_c + n_c(t)\cos\theta_c - n_s(t)\sin\theta_c \\ &= s(t,\varepsilon,d_n)\cos\theta_c + n_{cn}(t,\theta_c) \\ x_{sn}(t) &= \sqrt{P} d_n p\big(t-(n+\varepsilon)T\big)\sin\theta_c + n_c(t)\sin\theta_c + n_s(t)\cos\theta_c \\ &= s(t,\varepsilon,d_n)\sin\theta_c + n_{sn}(t,\theta_c) \end{aligned} \tag{10-78}$$

or, equivalently, in complex form,

$$\tilde{x}_n(t) = x_{cn}(t) + j x_{sn}(t) = s(t,\varepsilon,d_n)\, e^{j\theta_c} + \tilde{n}_n(t,\theta_c) \tag{10-79}$$

where

$$\begin{aligned} \tilde{n}_n(t,\theta) &= n_{cn}(t) + j n_{sn}(t) = \tilde{n}_n(t) e^{j\theta_c} \\ \tilde{n}_n(t) &= n_c(t) + j n_s(t) \end{aligned} \tag{10-80}$$

Then, for an observation of duration $T_0 = NT$ seconds, i.e., N iid symbols, the CLF (conditioning is now on both the unknown carrier phase θ_c and fractional symbol timing offset ε) is given by

$$\begin{aligned} L(\mathbf{d},\varepsilon,\theta_c) &= \frac{1}{\pi N_0}\exp\left(-\frac{1}{N_0}\int_{T_0}\left|\tilde{x}(t) - s(t,\varepsilon,\mathbf{d})\, e^{j\theta_c}\right|^2 dt\right) \\ &= C\exp\left(\frac{2}{N_0}\,\mathrm{Re}\left\{\int_{T_0}\tilde{x}(t)\, s(t,\varepsilon,\mathbf{d})\, e^{-j\theta_c} dt\right\}\right) \end{aligned} \tag{10-81}$$

where $\tilde{x}(t) = \big(\tilde{x}_1(t), \tilde{x}_2(t), \cdots, \tilde{x}_N(t)\big)$ is the collection of complex observables and C is a constant that is independent of the unknown parameters and also reflects the constant energy nature of the BPSK modulation. As before, because of the iid property of the data symbols, the CLF can be expressed as the product of per-symbol CLFs, namely,

$$L(\mathbf{d},\varepsilon,\theta_c) = \prod_{n=0}^{N-1}\exp\left(\frac{2}{N_0}\,\mathrm{Re}\left\{\int_{T_n(\varepsilon)}\tilde{x}_n(t)\, s(t,\varepsilon,d_n)\, e^{-j\theta_c} dt\right\}\right) \tag{10-82}$$

where $T_n(\varepsilon)$ denotes the time interval $(n+\varepsilon)T \le t \le (n+1+\varepsilon)T$ and for simplicity we have ignored all multiplicative constants since they do not affect the parameter estimation.

The issue that arises now is the order in which to perform the averaging over the unknown data sequence and the unknown carrier phase. Suppose that one attempts to first average over the carrier phase. In order to do this, we rewrite Eq. (10-82) in the form

$$L(\mathbf{d},\varepsilon,\theta_c) = \exp\left(\frac{2}{N_0}\,\mathrm{Re}\left\{\sum_{n=1}^{N}\int_{T_n(\varepsilon)} \tilde{x}_n(t)\, s(t,\varepsilon,d_n)\, e^{-j\theta_c} dt\right\}\right)$$

$$= \exp\left\{\frac{2}{N_0} R(\mathbf{d},\varepsilon)\cos\left[\theta_c - \alpha(\mathbf{d},\varepsilon)\right]\right\} \tag{10-83}$$

where

$$R(\mathbf{d},\varepsilon) = \left|\sum_{n=0}^{N-1}\int_{T_n(\varepsilon)} \tilde{x}_n(t) s(t,\varepsilon,d_n) dt\right|$$

$$= \left|\sum_{n=0}^{N-1} d_n\sqrt{P}\int_{T_n(\varepsilon)} \tilde{x}_n(t) p\big(t-(n+\varepsilon)T\big) dt\right| \tag{10-84}$$

$$\alpha(\mathbf{d},\varepsilon) = \arg\left\{\sum_{n=0}^{N-1}\int_{T_n(\varepsilon)} \tilde{x}_n(t) s(t,\varepsilon,d_n) dt\right\}$$

Averaging over the uniformly distributed carrier phase, we get[13]

$$L(\mathbf{d},\varepsilon) = I_0\left(\frac{2}{N_0} R(\mathbf{d},\varepsilon)\right)$$

$$= I_0\left(\frac{2\sqrt{P}}{N_0}\left|\sum_{n=0}^{N-1} d_n \int_{T_n(\varepsilon)} \tilde{x}_n(t)\, p\big(t-(n+\varepsilon)T\big) dt\right|\right) \tag{10-85}$$

[13] At this point, it should be re-emphasized that our approach differs from that in [13] in that in the latter the *per symbol* likelihood function is averaged over the carrier phase and then, because of the iid nature of the data, an LF is formed from the product of these phase-averaged LFs. Forming the LF in such a way implicitly assumes that the carrier phase varies independently from symbol to symbol, which is in opposition to our assumption that the carrier phase is constant over the observation.

The difficulty now lies in analytically averaging over the data sequence in Eq. (10-85) when N is large. Thus, in order to obtain simple metrics, before averaging over the data, we must first simplify matters by approximating the nonlinear (Bessel) function in Eq. (10-85). For small arguments (e.g., low SNR), the following approximation is appropriate:

$$I_0(x) \cong 1 + \frac{x^2}{4} \tag{10-86}$$

Applying Eq. (10-86) to Eq. (10-85) and defining the real observables

$$\begin{aligned} X_{cn}(\varepsilon) &\triangleq \int_{T_n(\varepsilon)} \tilde{x}_{cn}(t)\, p\big(t-(n+\varepsilon)T\big)dt \\ X_{sn}(\varepsilon) &\triangleq \int_{T_n(\varepsilon)} \tilde{x}_{sn}(t)\, p\big(t-(n+\varepsilon)T\big)dt \end{aligned} \tag{10-87}$$

we obtain

$$\begin{aligned} L(\mathbf{d},\varepsilon) &= I_0\left(\frac{2\sqrt{P}}{N_0}\left|\sum_{n=0}^{N-1} d_n\big(X_{cn}(\varepsilon)+jX_{sn}(\varepsilon)\big)\right|\right) \\ &\cong 1+\frac{P}{N_0^2}\left(\sum_{n=0}^{N-1} d_n X_{cn}(\varepsilon)\right)^2+\frac{P}{N_0^2}\left(\sum_{n=0}^{N-1} d_n X_{sn}(\varepsilon)\right)^2 \end{aligned} \tag{10-88}$$

Finally, averaging over the iid data sequences gives the simplified LF

$$L(\varepsilon) = 1+\frac{P}{N_0^2}\sum_{n=0}^{N-1} X_{cn}^2(\varepsilon)+\frac{P}{N_0^2}\sum_{n=0}^{N-1} X_{sn}^2(\varepsilon) \tag{10-89}$$

To arrive at a closed-loop symbol sync structure motivated by this LF, we proceed in the usual way by differentiating the LF with respect to ε and using the result to form the error signal in the loop. Taking the partial derivative of Eq. (10-89) with respect to ε and again ignoring multiplicative constants gives

$$\frac{\partial L(\varepsilon)}{\partial \varepsilon} = \sum_{n=0}^{N-1} X_{cn}(\varepsilon)\frac{dX_{cn}(\varepsilon)}{d\varepsilon}+X_{sn}(\varepsilon)\frac{dX_{sn}(\varepsilon)}{d\varepsilon} \tag{10-90}$$

each of whose terms is analogous to that which forms the error signal in the low SNR version of the coherent DTTL, i.e., the LDTTL. Thus, the low SNR version of the noncoherent DTTL, herein given the acronym NC-LDTTL, is nothing more than the parallel combination of two independent coherent LDTTLs acting on the I and Q baseband signals. A block diagram of this structure is given in Fig. 10-15, and the analysis of its performance will follow in the next subsection.

For large SNR, we need to approximate $I_0(x)$ in Eq. (10-85) by its large argument form, which behaves as $\exp(|x|)$. Thus, in this case the CLF would be approximated as

$$L(\mathbf{d},\varepsilon) \cong \exp\left(\frac{2\sqrt{P}}{N_0}\left|\sum_{n=0}^{N-1} d_n \int_{T_n(\varepsilon)} \tilde{x}_n(t)\, p\big(t-(n+\varepsilon)T\big)dt\right|\right) \tag{10-91}$$

which unfortunately does not ease the burden of averaging over the data sequence.

Suppose now instead we first average the CLF over the data. Then from Eq. (10-82) we have

$$\begin{aligned} L(\varepsilon,\theta_c) &= \underset{\mathbf{d}}{E}\left\{\prod_{n=0}^{N-1}\exp\left(\frac{2}{N_0}\operatorname{Re}\left\{\int_{T_n(\varepsilon)} \tilde{x}_n(t)\, s(t,\varepsilon,d_n)\, e^{-j\theta_c}dt\right\}\right)\right\} \\ &= \prod_{n=0}^{N-1}\underset{d_n}{E}\left\{\exp\left(d_n\frac{2\sqrt{P}}{N_0}\operatorname{Re}\left\{\int_{T_n(\varepsilon)} \tilde{x}_n(t)\, p\big(t-(n+\varepsilon)T\big)e^{-j\theta_c}dt\right\}\right)\right\} \\ &= \prod_{n=0}^{N-1}\cosh\left(\frac{2\sqrt{P}}{N_0}\left[X_{cn}(\varepsilon)\cos\theta_c + X_{sn}(\varepsilon)\sin\theta_c\right]\right) \\ &= \prod_{n=0}^{N-1}\cosh\left(\frac{2\sqrt{P}}{N_0}\sqrt{\big(X_{cn}(\varepsilon)\big)^2+\big(X_{sn}(\varepsilon)\big)^2}\cos(\theta_c-\beta_n)\right) \end{aligned} \tag{10-92}$$

where

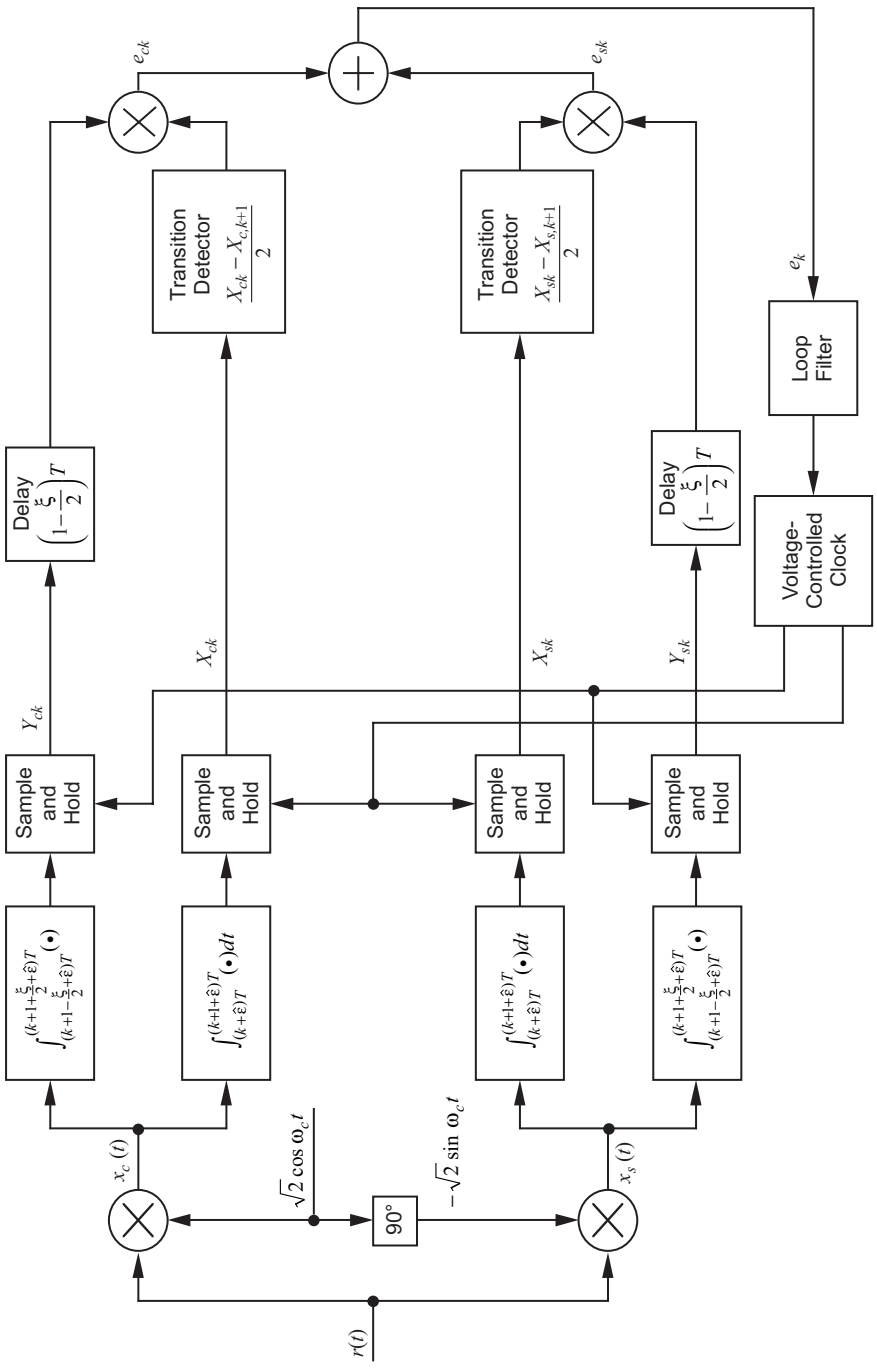


Fig. 10-15. Noncoherent linear data transition tracking loop (NC-LDTTL).

$$\beta_n = \tan^{-1} \frac{X_{sn}(\varepsilon)}{X_{cn}(\varepsilon)} \tag{10-93}$$

If we now apply the large argument (e.g., high SNR) approximation to the hyperbolic cosine function, namely,

$$\cosh x \cong \frac{\exp(|x|)}{2} \tag{10-94}$$

then Eq. (10-92) becomes (ignoring multiplicative constants)

$$L(\varepsilon, \theta_c) = \exp\left\{ \sum_{n=0}^{N-1} \left| \frac{2\sqrt{P}}{N_0} \sqrt{\left(X_{cn}(\varepsilon)\right)^2 + \left(X_{sn}(\varepsilon)\right)^2} \cos(\theta_c - \beta_n) \right| \right\} \tag{10-95}$$

which still presents difficulty in analytically averaging over the unknown parameter, in this case θ_c. Thus, having failed on both attempts at averaging the CLF over *both* the carrier phase and the data sequence at high SNR, we are forced to deviate from the true ML approach in favor of one that will provide a simple metric.

Another approach, albeit suboptimum, that can achieve near-ML performance is to choose (rather than average over) the value of the unknown parameter that maximizes the CLF. With reference to Eq. (10-85), in the limit of no noise (infinite SNR), the CLF $L(\mathbf{d}, \varepsilon)$, or equivalently the argument of the Bessel function, would be maximized when

$$d_n = \operatorname{sgn}\left[\int_{T_n(\varepsilon)} s(t, \varepsilon, d_n) p\big(t - (n+\varepsilon)T\big) dt \right] \tag{10-96}$$

i.e., all the signal vectors are aligned in the same direction, in which case the argument of the Bessel function (ignoring the constant multiplicative factor) would become

$$\left|\sum_{n=1}^{N} d_n \int_{T_n(\varepsilon)} s(t,\varepsilon,d_n) e^{j\theta_c} p\big(t-(n+\varepsilon)T\big)dt\right| =$$

$$\sum_{n=1}^{N}\left|\int_{T_n(\varepsilon)} s(t,\varepsilon,d_n) e^{j\theta_c} p\big(t-(n+\varepsilon)T\big)dt\right|$$

Thus, as a high SNR approximation of this limiting case, we propose the ad hoc unconditional LF

$$\mathrm{LF}(\varepsilon) = I_0\left(\frac{2\sqrt{P}}{N_0}\sum_{n=0}^{N-1}\left|\int_{T_n(\varepsilon)} \tilde{x}_n(t)\, p\big(t-(n+\varepsilon)\,T\big)dt\right|\right) \tag{10-97}$$

or, equivalently, taking the natural logarithm of Eq. (10-97), the log-likelihood function (LLF)

$$\begin{aligned}\mathrm{LLF}(\varepsilon) = \ln \mathrm{LF}(\varepsilon) &= \ln I_0\left(\frac{2\sqrt{P}}{N_0}\sum_{n=0}^{N-1}\left|\int_{T_n(\varepsilon)} \tilde{x}_n(t) p\big(t-(n+\varepsilon)T\big)dt\right|\right)\\ &= \ln I_0\left(\frac{2\sqrt{P}}{N_0}\sum_{n=0}^{N-1}\left|X_{cn}(\varepsilon)+jX_{sn}(\varepsilon)\right|\right)\\ &= \ln I_0\left(\frac{2\sqrt{P}}{N_0}\sum_{n=0}^{N-1}\sqrt{X_{cn}^2(\varepsilon)+X_{sn}^2(\varepsilon)}\right)\end{aligned} \tag{10-98}$$

For large arguments, the nonlinearity in Eq. (10-98) can be approximated (to within a scaling constant) as

$$\ln I_0(x) \cong |x| \tag{10-99}$$

which after substitution in Eq. (10-98) yields

$$\mathrm{LLF}(\varepsilon) = \frac{2\sqrt{P}}{N_0}\sum_{n=0}^{N-1}\sqrt{X_{cn}^2(\varepsilon)+X_{sn}^2(\varepsilon)} \tag{10-100}$$

Thus, analogous to Eq. (10-90), differentiating LLF(ε) with respect to ε, the error signal in a closed-loop configuration should be formed from

$$e = \sum_{n=0}^{N-1} \left[\frac{X_{cn}(\varepsilon)}{\sqrt{X_{cn}^2(\varepsilon) + X_{sn}^2(\varepsilon)}} \frac{dX_{cn}(\varepsilon)}{d\varepsilon} + \frac{X_{sn}(\varepsilon)}{\sqrt{X_{cn}^2(\varepsilon) + X_{sn}^2(\varepsilon)}} \frac{dX_{sn}(\varepsilon)}{d\varepsilon} \right] \tag{10-101}$$

A noncoherent DTTL-type symbol synchronizer that is motivated by using Eq. (10-101) as its error signal is illustrated in Fig. 10-16. The analysis of its performance will be discussed later on. In the meantime it is interesting to note that if one were to consider the coherent case wherein $X_{sn}(\varepsilon)$ would be absent, then setting $X_{sn}(\varepsilon)$ equal to zero in Eq. (10-101) gives

$$e = \sum_{n=0}^{N-1} \frac{X_{cn}(\varepsilon)}{\sqrt{X_{cn}^2(\varepsilon)}} \frac{dX_{cn}(\varepsilon)}{d\varepsilon} = \sum_{n=0}^{N-1} \frac{dX_{cn}(\varepsilon)}{d\varepsilon} \operatorname{sgn} X_{cn}(\varepsilon) \tag{10-102}$$

which is exactly the error signal that would be derived from the MAP approach under the assumption of perfectly known carrier phase and thus motivates the construction of the conventional nonlinear DTTL.

Before proceeding, it is interesting at this point to note that the LLF in Eq. (10-98) is the same[14] as that given in [13, Eq. 6.279], which is obtained by averaging over the carrier phase under the assumption that it varies independently from symbol to symbol as discussed in Footnote 13. The LLF in Eq. (10-98) can also be obtained from Eq. (10-95) by again assuming that the carrier phase varies independently from symbol to symbol, which is tantamount to replacing θ_c with θ_{cn} in this equation, and then *maximizing* over the sequence of carrier phases. It is clear from Eq. (10-95) that this maximization would occur for $\theta_{cn} = \beta_n$, in which case we obtain the LF

$$L(\varepsilon) = \exp\left\{ \sum_{n=0}^{N-1} \left| \frac{2\sqrt{P}}{N_0} \sqrt{\left(X_{cn}(\varepsilon)\right)^2 + \left(X_{sn}(\varepsilon)\right)^2} \right| \right\} \tag{10-103}$$

[14] The only difference is a factor of $\sqrt{T}$ in the argument of the Bessel function, i.e., $2\sqrt{P}/N_0$ in our result versus $2\sqrt{E}/N_0 = 2\sqrt{PT}/N_0$ in that of [13], which comes about because of the difference in the normalization of the carrier reference signals between the two approaches.

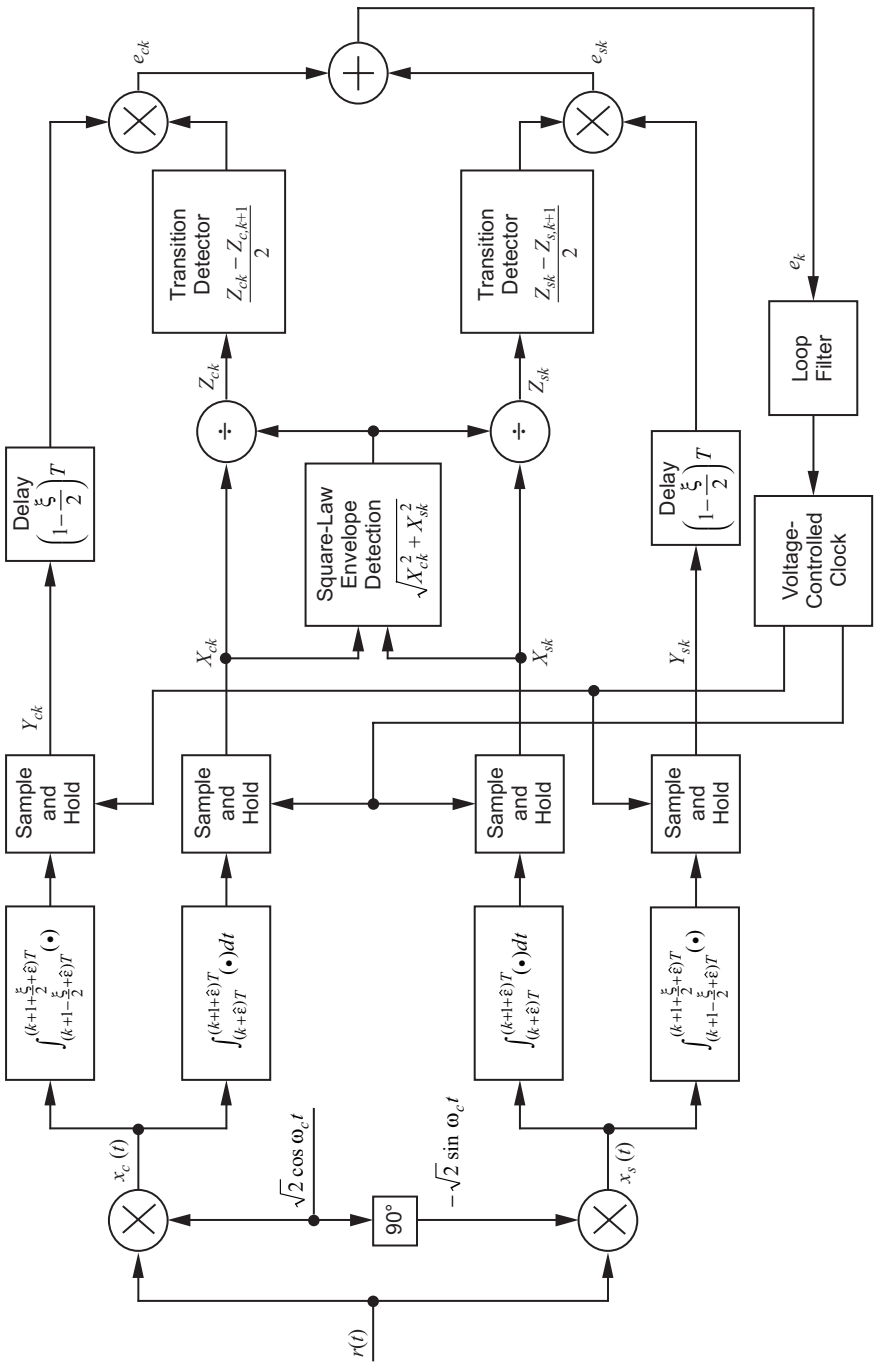


Fig. 10-16. Noncoherent nonlinear data transition tracking loop (NC-NLDTTL).

Finally, taking the natural logarithm of Eq. (10-103) produces a result identical to Eq. (10-100) for the LLF.

10.6.2.2 Tracking Performance of the Noncoherent Linear DTTL. With reference to Fig. 10-15, the upper and lower channel (herein referred to as "cosine" and "sine" channels) inputs to the I&D filters in the nth symbol interval are described by Eq. (10-78). The local clock produces a timing reference for the I and Q I&D filters of each of these channels that depends on the estimate $\hat{\varepsilon}$ of ε. As such, the outputs of these same filters are respectively given by (assuming for simplicity that all gains are set equal to unity)

$$\begin{aligned} X_{ck} &= \int_{kT+\hat{\varepsilon}}^{(k+1)T+\hat{\varepsilon}} x_{ck}(t)dt \\ &= \cos\theta_c \overbrace{\int_{kT+\hat{\varepsilon}}^{(k+1)T+\hat{\varepsilon}} s(t,\varepsilon,d_k)dt}^{c_k} + \cos\theta_c \overbrace{\int_{kT+\hat{\varepsilon}}^{(k+1)T+\hat{\varepsilon}} n_c(t)\,dt}^{\nu_{ck}} \\ &\quad - \sin\theta_c \overbrace{\int_{kT+\hat{\varepsilon}}^{(k+1)T+\hat{\varepsilon}} n_s(t)\,dt}^{\nu_{sk}} \end{aligned} \tag{10-104}$$

$$\begin{aligned} Y_{ck} &= \int_{(k+1-\frac{\xi}{2})T+\hat{\varepsilon}}^{(k+1+\frac{\xi}{2})T+\hat{\varepsilon}} x_{ck}(t)dt \\ &= \cos\theta_c \overbrace{\int_{(k+1-\frac{\xi}{2})T+\hat{\varepsilon}}^{(k+1+\frac{\xi}{2})T+\hat{\varepsilon}} s(t,\varepsilon,d_k)dt}^{b_k} + \cos\theta_c \overbrace{\int_{(k+1-\frac{\xi}{2})T+\hat{\varepsilon}}^{(k+1+\frac{\xi}{2})T+\hat{\varepsilon}} n_c(t)\,dt}^{\mu_{ck}} \\ &\quad - \sin\theta_c \overbrace{\int_{(k+1-\frac{\xi}{2})T+\hat{\varepsilon}}^{(k+1+\frac{\xi}{2})T+\hat{\varepsilon}} n_s(t)\,dt}^{\mu_{sk}} \end{aligned}$$

and

$$
\begin{aligned}
X_{sk} &= \int_{kT+\hat{\varepsilon}}^{(k+1)T+\hat{\varepsilon}} x_{sk}(t)dt \\
&= \sin\theta_c \overbrace{\int_{kT+\hat{\varepsilon}}^{(k+1)T+\hat{\varepsilon}} s(t,\varepsilon,d_k)dt}^{c_k} + \sin\theta_c \overbrace{\int_{kT+\hat{\varepsilon}}^{(k+1)T+\hat{\varepsilon}} n_c\left(t\right)dt}^{\nu_{ck}} \\
&\quad + \cos\theta_c \overbrace{\int_{kT+\hat{\varepsilon}}^{(k+1)T+\hat{\varepsilon}} n_s\left(t\right)dt}^{\nu_{sk}}
\end{aligned}
\tag{10-105}
$$

$$
\begin{aligned}
Y_{sk} &= \int_{(k+1-\frac{\xi}{2})T+\hat{\varepsilon}}^{(k+1+\frac{\xi}{2})T+\hat{\varepsilon}} x_{sk}(t)dt \\
&= \sin\theta_c \overbrace{\int_{(k+1-\frac{\xi}{2})T+\hat{\varepsilon}}^{(k+1+\frac{\xi}{2})T+\hat{\varepsilon}} s(t,\varepsilon,d_k)dt}^{b_k} + \sin\theta_c \overbrace{\int_{(k+1-\frac{\xi}{2})T+\hat{\varepsilon}}^{(k+1+\frac{\xi}{2})T+\hat{\varepsilon}} n_c\left(t\right)dt}^{\mu_{ck}} \\
&\quad + \cos\theta_c \overbrace{\int_{(k+1-\frac{\xi}{2})T+\hat{\varepsilon}}^{(k+1+\frac{\xi}{2})T+\hat{\varepsilon}} n_s\left(t\right)dt}^{\mu_{sk}}
\end{aligned}
$$

Since μ_{ck} and ν_{ck} are not independent, and likewise for μ_{sk} and ν_{sk}, it is convenient as before to express them in terms of a new set of variables:

$$
\begin{aligned}
\nu_{ck} &= N_{ck} + M_{ck},\ \mu_{ck} = M'_{ck} + N'_{c,k+1} \\
\nu_{sk} &= N_{sk} + M_{sk},\ \mu_{sk} = M'_{sk} + N'_{s,k+1}
\end{aligned}
\tag{10-106}
$$

where

$$N_{ck} = \int_{kT+\hat{\varepsilon}}^{(k+\frac{1}{2})T+\hat{\varepsilon}} n_c(t)dt,\ M_{ck} = \int_{(k+\frac{1}{2})T+\hat{\varepsilon}}^{(k+1)T+\hat{\varepsilon}} n_c(t)dt$$
$$N'_{ck} = \int_{kT+\hat{\varepsilon}}^{(k+\frac{\xi}{2})T+\hat{\varepsilon}} n_c(t)dt,\ M'_{ck} = \int_{(k+1-\frac{\xi}{2})T+\hat{\varepsilon}}^{(k+1)T+\hat{\varepsilon}} n_c\,(t)\,dt \tag{10-107}$$

with the properties

N_{ck}, M_{cn} are mutually independent for all k, n

N'_{ck}, M'_{cn} are mutually independent for all k, n

N'_{ck}, M_{cn} and M'_{ck}, N_{cn} are mutually independent for all k, n

N'_{ck}, N'_{cn} and M_{ck}, M_{cn} are mutually independent for all $k \neq n$

Furthermore, all $M_{ck}, M'_{ck}, N_{ck}, N'_{ck}$, and their sums are Gaussian random variables with zero mean and variances

$$\sigma^2_{M_{ck}} = \sigma^2_{N_{ck}} = N_0T/4$$
$$\sigma^2_{M'_{ck}} = \sigma^2_{N'_{ck}} = \xi N_0T/4 \tag{10-108}$$

Analogous definitions and properties apply to the sine channel noise variables.

Taking the difference of two successive soft decisions, X_{ck} and $X_{c,k+1}$ (or X_{sk} and $X_{s,k+1}$), and multiplying the average of the result by the quadrature I&D output, Y_{ck} (or Y_{sk}), delayed by $(1-\xi/2)\,T$ gives the sine and cosine channel error signal components in the kth symbol interval:

$$\begin{aligned} e_{ck} = {} & \left(b_k\cos\theta_c + \left(M'_{ck}+N'_{c,k+1}\right)\cos\theta_c - \left(M'_{sk}+N'_{s,k+1}\right)\sin\theta_c\right) \\ & \times\frac{1}{2}\left\{\left[c_k\cos\theta_c + (N_{ck}+M_{ck})\cos\theta_c - (N_{sk}+M_{sk})\sin\theta_c\right]\right. \\ & \left. -\ \left[c_{k+1}\cos\theta_c + (N_{c,k+1}+M_{c,k+1})\cos\theta_c - (N_{s,k+1}+M_{s,k+1})\sin\theta_c\right]\right\} \end{aligned} \tag{10-109}$$

and

$$
\begin{aligned}
e_{sk} = & \left(b_k \sin\theta_c + \left(M'_{ck} + N'_{c,k+1}\right)\sin\theta_c + \left(M'_{sk} + N'_{s,k+1}\right)\cos\theta_c\right) \\
& \times \frac{1}{2}\left\{\left[c_k \sin\theta_c + (N_{ck} + M_{ck})\sin\theta_c + (N_{sk} + M_{sk})\cos\theta_c\right]\right. \\
& \left. - \left[c_{k+1}\sin\theta_c + (N_{c,k+1} + M_{c,k+1})\sin\theta_c + (N_{s,k+1} + M_{s,k+1})\cos\theta_c\right]\right\}
\end{aligned} \tag{10-110}
$$

The total error signal, e_k, is the sum of the two components in Eqs. (10-110) and (10-109).

10.6.2.3 S-Curve Performance. The S-curve is by definition the statistical average of the error signal over the signal and noise probability distributions. Letting $\lambda \triangleq \varepsilon - \hat{\varepsilon}$ denote the normalized timing error ($-1/2 \le \lambda \le 1/2$), the S-curve $g(\lambda)$ becomes

$$
\begin{aligned}
g(\lambda) &= E_{n,s}\{e_{ck} + e_{sk}\} \\
&= E_s\left\{b_k\left(\frac{c_k - c_{k+1}}{2}\right)\cos^2\theta_c + b_k\left(\frac{c_k - c_{k+1}}{2}\right)\sin^2\theta_c\right\} \\
&= E_s\left\{b_k\left(\frac{c_k - c_{k+1}}{2}\right)\right\}
\end{aligned} \tag{10-111}
$$

which is independent of the carrier phase error as expected and also identical to the result for the coherent LDTTL. Thus, using the results from Section 4.1 with a slight simplification in the notation, i.e., ignoring the gain constants K_1 and K_2, we have

$$
g_n(\lambda) \triangleq \frac{g(\lambda)}{PT^2} = \begin{cases} \lambda\left(1 - \dfrac{\xi}{4}\right) - \dfrac{3}{2}\lambda^2, & 0 \le \lambda \le \dfrac{\xi}{2} \\ \dfrac{\xi}{2}(1 - 2\lambda), & \dfrac{\xi}{2} \le \lambda \le \dfrac{1}{2} \end{cases} \tag{10-112}
$$

As noted there, the normalized S-curve for the LDTTL is independent of SNR, whereas that for the conventional (nonlinear) DTTL is highly dependent on SNR. Taking the derivative of Eq. (10-112) with respect to λ and evaluating the result at $\lambda = 0$ gives the slope of the normalized S-curve at the origin, namely,

$$K_g \triangleq \frac{dg(\lambda)}{d\lambda}\,|_{\lambda=0} = PT^2\left(1-\frac{\xi}{4}\right) \tag{10-113}$$

10.6.2.4 Noise Performance. The equivalent noise $n_\lambda(t)$ perturbing the loop is characterized by the variation of the loop error signal around its mean (the S-curve) as in Eq. (10-27). Following the same approach as that taken in Section 10.3.2, then after some laborious analysis, we arrive at the equivalent noise power spectral density given by Eq. (10-30), where now

$$\begin{aligned} E_{n,s}\left\{e_n^2\,|_{\lambda=0}\right\} &= P^2T^4\left[\frac{\xi}{4R_s}\left(1+\frac{\xi}{2}+\frac{1}{R_s}\right)\right] \\ E_{n,s}\left\{e_n e_{n+m}\,|_{\lambda=0}\right\} &= \begin{cases} -P^2T^4\dfrac{\xi^2}{32R_s}, & m=1 \\ 0, & m>1 \end{cases} \end{aligned} \tag{10-114}$$

Substituting Eq. (10-114) into Eq. (10-30) gives the desired equivalent power spectral density as

$$N_0' = P^2T^5\frac{\xi}{2R_s}\left(1+\frac{\xi}{4}+\frac{1}{R_s}\right) \tag{10-115}$$

Interestingly enough, the result for the coherent LDTTL is given by [see Eq. (10-37)]

$$N_0' = P^2T^5\frac{\xi}{2R_s}\left(1+\frac{\xi}{4}+\frac{1}{2R_s}\right) \tag{10-116}$$

although by comparison the mathematics employed to arrive at Eq. (10-116) is considerably simpler than that needed to arrive at Eq. (10-115).

10.6.2.5 Mean-Squared Timing-Error Performance. The mean-squared timing error σ_λ^2 of the noncoherent LDTTL is readily computed for a first-order loop filter and large loop SNR conditions using the relation in Eq. (10-39), where now K_g is obtained from Eq. (10-113) and N_0' from Eq. (10-115). Making the appropriate substitutions in Eq. (10-39) gives the result

$$\sigma_\lambda^2\,|_{\text{NC-LDTTL}} = \frac{\xi\left(1+\dfrac{\xi}{4}+\dfrac{1}{R_s}\right)}{2\rho\left(1-\dfrac{\xi}{4}\right)^2} \tag{10-117}$$

which is to be compared with a similar result in Eq. (10-40) for the coherent LDTTL.

10.6.2.6 Tracking Performance of the Noncoherent Nonlinear DTTL. For the nonlinear noncoherent DTTL illustrated in Fig. 10-16, analogous to Eqs. (10-109) and (10-110), the error signal components are now given by

$$\begin{aligned} e_{ck} &= Y_{ck} \times \frac{1}{2}\left[\frac{X_{ck}}{\sqrt{X_{ck}^2+X_{sk}^2}} - \frac{X_{c,k+1}}{\sqrt{X_{c,k+1}^2+X_{s,k+1}^2}}\right] \\ &= \left(b_k\cos\theta_c + \left(M'_{ck}+N'_{c,k+1}\right)\cos\theta_c - \left(M'_{sk}+N'_{s,k+1}\right)\sin\theta_c\right) \\ &\quad\times\frac{1}{2}\left[\frac{(c_k+N_{ck}+M_{ck})\cos\theta_c-(N_{sk}+M_{sk})\sin\theta_c}{\sqrt{(c_k+N_{ck}+M_{ck})^2+(N_{sk}+M_{sk})^2}}\right. \\ &\quad\left.-\frac{(c_{k+1}+N_{c,k+1}+M_{c,k+1})\cos\theta_c-(N_{s,k+1}+M_{s,k+1})\sin\theta_c}{\sqrt{(c_{k+1}+N_{c,k+1}+M_{c,k+1})^2+(N_{s,k+1}+M_{s,k+1})^2}}\right] \end{aligned} \tag{10-118}$$

and

$$\begin{aligned} e_{sk} &= Y_{sk} \times \frac{1}{2}\left[\frac{X_{sk}}{\sqrt{X_{ck}^2+X_{sk}^2}} - \frac{X_{s,k+1}}{\sqrt{X_{c,k+1}^2+X_{s,k+1}^2}}\right] \\ &= \left(b_k\sin\theta_c + \left(M'_{ck}+N'_{c,k+1}\right)\sin\theta_c + \left(M'_{sk}+N'_{s,k+1}\right)\cos\theta_c\right) \\ &\quad\times\frac{1}{2}\left[\frac{(c_k+N_{ck}+M_{ck})\sin\theta_c+(N_{sk}+M_{sk})\cos\theta_c}{\sqrt{(c_k+N_{ck}+M_{ck})^2+(N_{sk}+M_{sk})^2}}\right. \\ &\quad\left.-\frac{(c_{k+1}+N_{c,k+1}+M_{c,k+1})\sin\theta_c+(N_{s,k+1}+M_{s,k+1})\cos\theta_c}{\sqrt{(c_{k+1}+N_{c,k+1}+M_{c,k+1})^2+(N_{s,k+1}+M_{s,k+1})^2}}\right] \end{aligned} \tag{10-119}$$

The total error signal, e_k, is now the sum of Eqs. (10-118) and (10-119), which after some trigonometric simplification becomes

$$e_k = \frac{1}{2}\left(b_k + M'_{ck} + N'_{c,k+1}\right)\left[\frac{c_k + N_{ck} + M_{ck}}{\sqrt{\left(c_k + N_{ck} + M_{ck}\right)^2 + \left(N_{sk} + M_{sk}\right)^2}}\right.$$

$$\left. - \frac{c_{k+1} + N_{c,k+1} + M_{c,k+1}}{\sqrt{\left(c_{k+1} + N_{c,k+1} + M_{c,k+1}\right)^2 + \left(N_{s,k+1} + M_{s,k+1}\right)^2}}\right]$$

$$+ \frac{1}{2}\left(M'_{sk} + N'_{s,k+1}\right)\left[\frac{N_{sk} + M_{sk}}{\sqrt{\left(c_k + N_{ck} + M_{ck}\right)^2 + \left(N_{sk} + M_{sk}\right)^2}}\right.$$

$$\left. - \frac{N_{s,k+1} + M_{s,k+1}}{\sqrt{\left(c_{k+1} + N_{c,k+1} + M_{c,k+1}\right)^2 + \left(N_{s,k+1} + M_{s,k+1}\right)^2}}\right] \qquad (10\text{-}120)$$

which is clearly independent of the unknown carrier phase, θ_c, as desired. To analytically compute even just the S-curve, much less the equivalent power spectral density, is now a daunting if not impossible task. Thus, in order to determine the tracking performance of this scheme, we shall have to turn to results obtained from computer simulations. Before doing this, however, we do note that in the absence of noise (i.e., in the limit of infinite SNR), the error signal of Eq. (10-120) becomes

$$e_k = b_k \frac{\operatorname{sgn} c_k - \operatorname{sgn} c_{k+1}}{2} \qquad (10\text{-}121)$$

which is the exact same result as for the coherent conventional (nonlinear) DTTL, and thus one can anticipate that, in the limit of large SNR, the noncoherent scheme should suffer little or no performance penalty relative to the coherent one. To demonstrate this as well as the behavior of the symbol synchronizer in other SNR regions, we turn to results obtained from computer simulation.

Figures 10-17 and 10-18 are illustrations of the normalized S-curves for the NC-NLDTTL for two different values of normalized window width and a variety of SNR values. The curves were obtained by numerically averaging the error signal of Eq. (10-120) over the data and noise statistics Also superimposed on

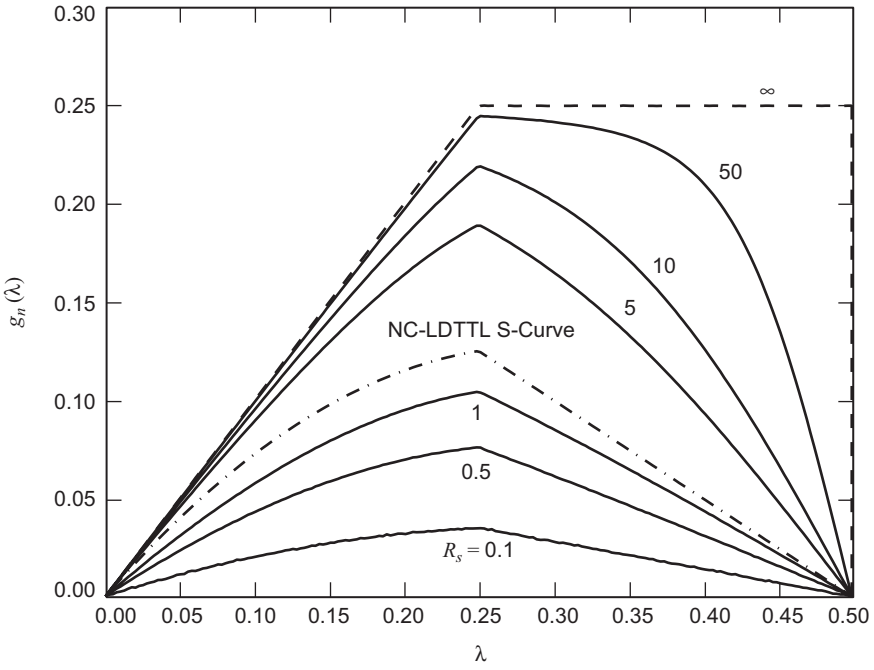


Fig. 10-17. NC-NLDTTL S-curve; ξ = 0.5**.**

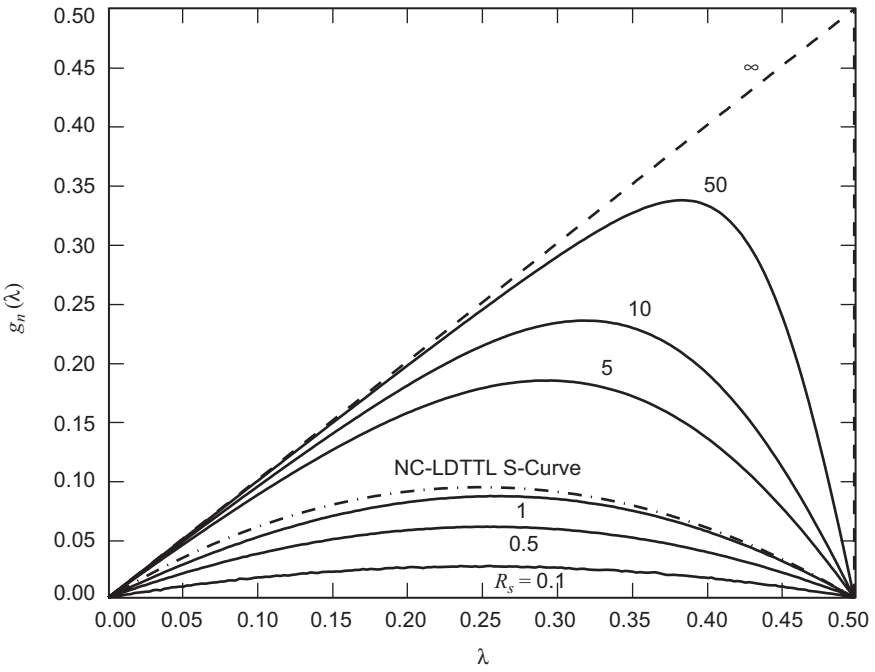


Fig. 10-18. NC-NLDTTL S-curve; ξ = 1.0**.**

these results are the corresponding S-curves for the NC-LDTTL as obtained from Eq. (10-112), which as previously mentioned are independent of SNR.

To demonstrate the performance trade-off of the noncoherent versus the coherent DTTL schemes as a function of SNR, Figs. 10-19 and 10-20 plot the mean-squared timing-error ratios $\sigma_\lambda^2\,|_{\text{NC-NLDTTL}}\,/\sigma_\lambda^2\,|_{\text{DTTL}}$ and $\sigma_\lambda^2\,|_{\text{NC-LDTTL}}\,/\sigma_\lambda^2\,|_{\text{LDTTL}}$, respectively, in dB versus SNR in dB for three different values of normalized window width ξ. For the first of the two ratios, the variance $\sigma_\lambda^2\,|_{\text{NC-NLDTTL}}$ is obtained by computer simulation and then divided by the variance of the conventional (nonlinear) DTTL obtained from the results in [1] as

$$\sigma_\lambda^2\,|_{\text{DTTL}} = \frac{\xi\left(1+\dfrac{\xi R_s}{2}-\dfrac{\xi}{2}\left[\dfrac{1}{\sqrt{\pi}}\exp\left(-R_s\right)+\sqrt{R_s}\,\text{erf}\sqrt{R_s}\right]^2\right)}{2\rho\left(\text{erf}\sqrt{R_s}-\dfrac{\xi}{2}\sqrt{\dfrac{R_s}{\pi}}\exp\left(-R_s\right)\right)^2} \tag{10-122}$$

The second of the two ratios is simply obtained from the division of Eq. (10-117) by Eq. (10-40). In both cases we observe that, as expected, the noncoherent and coherent performances approach each other as the SNR gets large (i.e., the above variance ratios approach unity or 0 dB). In the limit of infinitesimally small SNR, the noncoherent schemes pay a performance penalty with respect to the coherent schemes, which in the linear case is easily computed from Eqs. (10-117) and (10-40) to be 3 dB, while in the nonlinear case it appears to be somewhat less and mildly dependent on the window width. Finally, a comparison between the noncoherent linear and nonlinear DTTL performances is illustrated in Fig. 10-21, where the ratio of $\sigma_\lambda^2\,|_{\text{NC-LDTTL}}$ to $\sigma_\lambda^2\,|_{\text{NC-NLDTTL}}$ in dB is plotted versus SNR, R_s, in dB for a variety of different window widths. Analogous to a similar plot for the coherent DTTL schemes in Fig. 10-6, for each window width there exists a crossover point at which the variance ratio equals unity (or, equivalently, 0 dB), indicating the value of SNR that separates the SNR regions where one scheme is preferable over the other.

10.7 The Impact of Carrier Frequency Offset on Performance

Thus far in our discussions of noncoherent symbol synchronization, the word "noncoherent" was used to mean that the carrier phase was completely unknown [i.e., uniformly distributed in the interval $(-\pi, \pi)$] but at the same time the carrier frequency was assumed to be known exactly. Here, as an example,

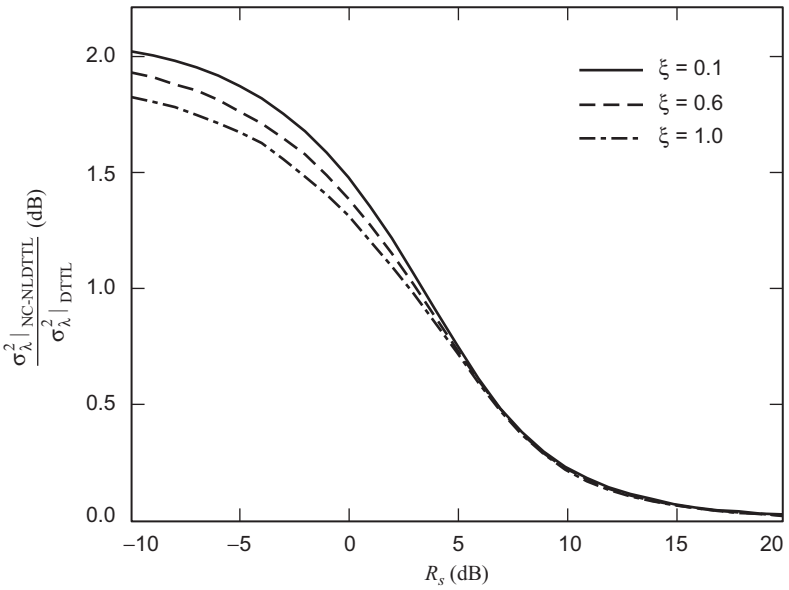


Fig. 10-19. The ratio of the mean-squared timing error for the noncoherent nonlinear DTTL to that of the coherent nonlinear DTTL with window width (ξ) as a parameter.

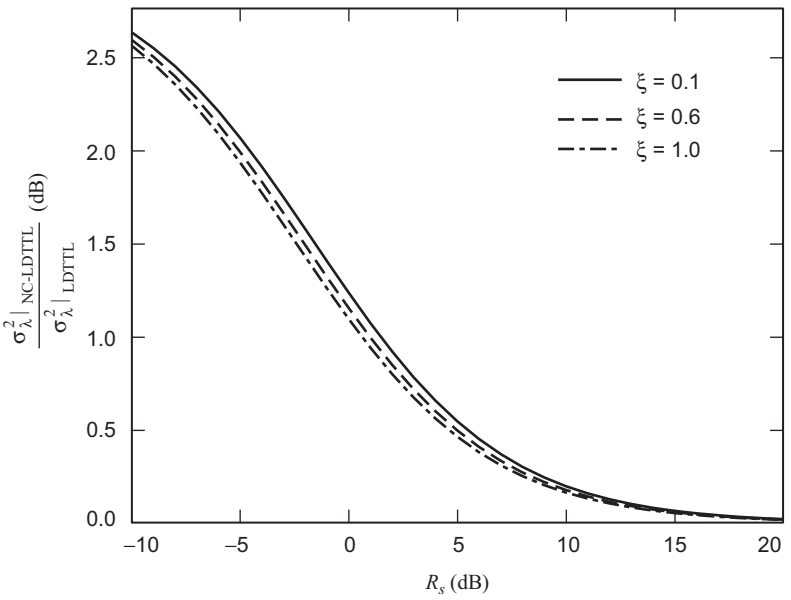


Fig. 10-20. The ratio of the mean-squared timing error for the noncoherent linear DTTL to that of the coherent linear DTTL with window width (ξ) as a parameter.

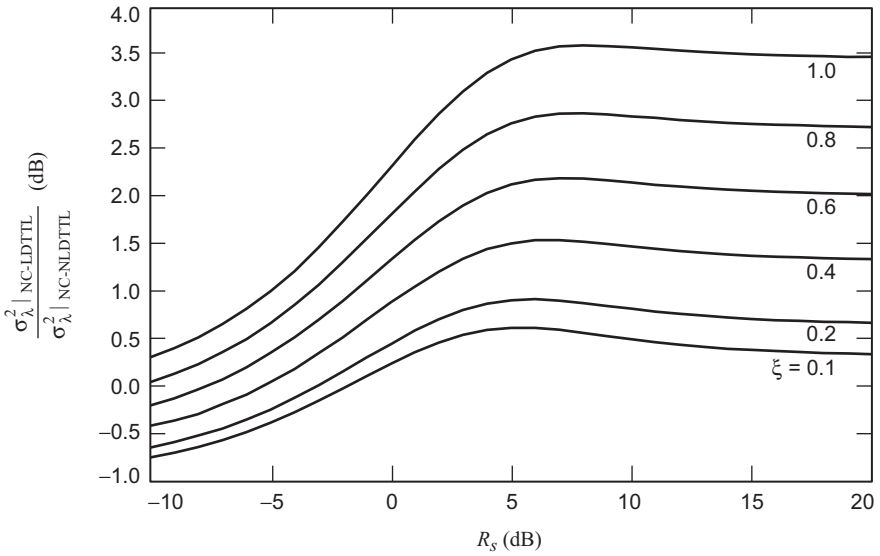


Fig. 10-21. The ratio of the mean-squared timing error for the noncoherent linear DTTL to that of the noncoherent nonlinear DTTL with window width (ξ) as a parameter.

we investigate the impact on the performance of the NC-LDTTL of imperfect knowledge of the carrier frequency, i.e., the presence of a fixed carrier frequency offset. In particular, we shall briefly rederive the expression for the variance of the normalized timing error in the loop and demonstrate its exact dependence on the ratio of symbol rate to frequency offset. Comparing this expression with that corresponding to the zero frequency offset case allows us to assess the additional degradation in performance that arises from the presence of the offset.

Consider the NC-LDTTL illustrated in Fig. 10-15, whose bandpass input $r(t) = s(t,\varepsilon) + n(t)$ is as before a BPSK modulation consisting of a binary NRZ data stream direct-modulated onto a carrier plus noise. While the signal component $s(t,\varepsilon)$ of this input will still be modeled as in the combination of Eq. (10-65) with Eq. (10-64), as we shall discuss momentarily it will be mathematically convenient to choose a slightly different representation of the bandpass noise $n(t)$ than the one given in Eq. (10-65).

The first step is to demodulate the received signal with the quadrature carrier reference signals

$$\begin{aligned} r_c(t) &= \sqrt{2}\cos\hat{\omega}_c t \\ r_s(t) &= -\sqrt{2}\sin\hat{\omega}_c t \end{aligned} \tag{10-123}$$

whose frequency is now assumed to be in error by an amount $\Delta\omega = \omega_c - \hat{\omega}_c$. Since the choice of frequency and phase used for representation of a bandpass noise process is arbitrary, for the purpose of analysis we choose these parameters corresponding to those of the carrier demodulation reference signals, in which case we have

$$n(t) = \sqrt{2}\left[n_c(t)\cos\hat{\omega}_c t - n_s(t)\sin\hat{\omega}_c t\right] \tag{10-124}$$

where $n_c(t), n_s(t)$ are as before low-pass AWGN processes with two-sided power spectral density (PSD) $N_0/2$ W/Hz. Thus, demodulating $r(t)$ with the quadrature reference signals of Eq. (10-123) results in the pair of baseband observables in the kth symbol interval $(k+\varepsilon)T \le t \le (k+1+\varepsilon)T$,

$$\begin{aligned} x_{ck}(t) &= \sqrt{P}d_k p\big(t-(k+\varepsilon)T\big)\cos\left(\Delta\omega t + \theta_c\right) + n_c(t) \\ x_{sk}(t) &= \sqrt{P}d_k p\big(t-(k+\varepsilon)T\big)\sin\left(\Delta\omega t + \theta_c\right) + n_s(t) \end{aligned} \tag{10-125}$$

Analogous to Eq. (10-104), the outputs of the I and Q I&D filters are now

$$\begin{aligned} X_{ck} &= \int_{kT+\hat{\varepsilon}}^{(k+1)T+\hat{\varepsilon}} x_{ck}(t)dt \\ &= \overbrace{\int_{kT+\hat{\varepsilon}}^{(k+1)T+\hat{\varepsilon}} \sqrt{P}d_k\cos\left(\Delta\omega t+\theta_c\right)dt}^{c_{ck}} + \overbrace{\int_{kT+\hat{\varepsilon}}^{(k+1)T+\hat{\varepsilon}} n_c(t)\,dt}^{\nu_{ck}} \\ Y_{ck} &= \int_{\left(k+1-\frac{\xi}{2}\right)T+\hat{\varepsilon}}^{\left(k+1+\frac{\xi}{2}\right)T+\hat{\varepsilon}} x_{ck}(t)dt \\ &= \overbrace{\int_{\left(k+1-\frac{\xi}{2}\right)T+\hat{\varepsilon}}^{\left(k+1+\frac{\xi}{2}\right)T+\hat{\varepsilon}} \sqrt{P}d_k\cos\left(\Delta\omega t+\theta_c\right)dt}^{b_{ck}} + \overbrace{\int_{\left(k+1-\frac{\xi}{2}\right)T+\hat{\varepsilon}}^{\left(k+1+\frac{\xi}{2}\right)T+\hat{\varepsilon}} n_c(t)\,dt}^{\mu_{ck}} \end{aligned} \tag{10-126}$$

and

$$
\begin{aligned}
X_{sk} &= \int_{kT+\hat{\varepsilon}}^{(k+1)T+\hat{\varepsilon}} x_{sk}(t)dt \\
&= \overbrace{\int_{kT+\hat{\varepsilon}}^{(k+1)T+\hat{\varepsilon}} \sqrt{P} d_k \sin\left(\Delta\omega t + \theta_c\right) dt}^{c_{sk}} + \overbrace{\int_{kT+\hat{\varepsilon}}^{(k+1)T+\hat{\varepsilon}} n_s\left(t\right) dt}^{\nu_{sk}} \\
Y_{sk} &= \int_{\left(k+1-\frac{\xi}{2}\right)T+\hat{\varepsilon}}^{\left(k+1+\frac{\xi}{2}\right)T+\hat{\varepsilon}} x_{sk}(t)dt \\
&= \overbrace{\int_{\left(k+1-\frac{\xi}{2}\right)T+\hat{\varepsilon}}^{\left(k+1+\frac{\xi}{2}\right)T+\hat{\varepsilon}} \sqrt{P} d_k \sin(\Delta\omega t + \theta_c) dt}^{b_{sk}} + \overbrace{\int_{\left(k+1-\frac{\xi}{2}\right)T+\hat{\varepsilon}}^{\left(k+1+\frac{\xi}{2}\right)T+\hat{\varepsilon}} n_s\left(t\right) dt}^{\mu_{sk}}
\end{aligned}
\tag{10-127}
$$

Taking the difference of two successive soft decisions, X_{ck} and $X_{c,k+1}$ (or X_{sk} and $X_{s,k+1}$), and multiplying the average of the result by the quadrature I&D output Y_{ck} (or Y_{sk}), delayed by $(1-\xi/2)\,T$, gives the sine and cosine channel error signal components in the kth symbol interval:

$$
e_{ck} = \left(b_{ck} + M'_{ck} + N'_{c,k+1}\right) \frac{\left(c_{ck} + N_{ck} + M_{ck}\right) - \left(c_{c,k+1} + N_{c,k+1} + M_{c,k+1}\right)}{2}
\tag{10-128}
$$

and

$$
e_{sk} = \left(b_{sk} + M'_{sk} + N'_{s,k+1}\right) \frac{\left(c_{ck} + N_{sk} + M_{sk}\right) - \left(c_{s,k+1} + N_{s,k+1} + M_{s,k+1}\right)}{2}
\tag{10-129}
$$

The total error signal, e_k, is again the sum of the two components in Eqs. (10-128) and (10-129).

10.7.1 S-Curve Performance

Using Eqs. (10-128) and (10-129), the S-curve $g\left(\lambda\right)$ is evaluated as

$$
\begin{aligned}
g(\lambda) &= E_{n,s}\left\{e_{ck} + e_{sk}\right\} \\
&= E_s\left\{b_{ck}\left(\frac{c_{ck} - c_{c,k+1}}{2}\right) + b_{sk}\left(\frac{c_{sk} - c_{s,k+1}}{2}\right)\right\}
\end{aligned}
\tag{10-130}
$$

which as we shall see shortly is still independent of the carrier phase but is now dependent on the frequency offset. Evaluating the signal terms in Eq. (10-130), whose definition appears in Eqs. (10-126) and (10-127), and carrying out the necessary integrations, we arrive at the following results:

$$\begin{aligned} c_{ck} =& \sqrt{P} d_{k-1} \lambda T \left(\frac{\sin \pi \eta \lambda}{\pi \eta \lambda} \right) \cos \left(2\pi \eta \left(k + \frac{\lambda}{2} \right) + \theta \right) \\ &+ \sqrt{P} d_k \lambda T \left(\frac{\sin \pi \eta (1-\lambda)}{\pi \eta (1-\lambda)} \right) \cos \left(2\pi \eta \left(k + \frac{1}{2} + \frac{\lambda}{2} \right) + \theta \right) \end{aligned} \tag{10-131}$$

$$\begin{aligned} b_{ck} =& \sqrt{P} d_k \left(\frac{\xi}{2} + \lambda \right) T \left(\frac{\sin \pi \eta \left(\frac{\xi}{2} + \lambda \right)}{\pi \eta \left(\frac{\xi}{2} + \lambda \right)} \right) \cos \left(2\pi \eta \left(k + 1 + \frac{\lambda - \frac{\xi}{2}}{2} \right) + \theta \right) \\ &+ \sqrt{P} d_{k+1} \left(\frac{\xi}{2} - \lambda \right) T \left(\frac{\sin \eta \left(\frac{\xi}{2} - \lambda \right)}{\eta \left(\frac{\xi}{2} - \lambda \right)} \right) \cos \left(2\pi \eta \left(k + 1 + \frac{\lambda + \frac{\xi}{2}}{2} \right) + \theta \right) \end{aligned}$$

and

$$\begin{aligned} c_{sk} =& -\sqrt{P} d_{k-1} \lambda T \left(\frac{\sin \pi \eta \lambda}{\pi \eta \lambda} \right) \sin \left(2\pi \eta \left(k + \frac{\lambda}{2} \right) + \theta \right) \\ &- \sqrt{P} d_k \lambda T \left(\frac{\sin \pi \eta (1-\lambda)}{\pi \eta (1-\lambda)} \right) \sin \left(2\eta \left(k + \frac{1}{2} + \frac{\lambda}{2} \right) + \theta \right) \end{aligned} \tag{10-132}$$

$$\begin{aligned} b_{sk} =& -\sqrt{P} d_k \left(\frac{\xi}{2} + \lambda \right) T \left(\frac{\sin \pi \eta \left(\frac{\xi}{2} + \lambda \right)}{\pi \eta \left(\frac{\xi}{2} + \lambda \right)} \right) \sin \left(2\pi \eta \left(k + 1 + \frac{\lambda - \frac{\xi}{2}}{2} \right) + \theta \right) \\ &- \sqrt{P} d_{k+1} \left(\frac{\xi}{2} - \lambda \right) T \left(\frac{\sin \pi \eta \left(\frac{\xi}{2} - \lambda \right)}{\pi \eta \left(\frac{\xi}{2} - \lambda \right)} \right) \sin \left(2\pi \eta \left(k + 1 + \frac{\lambda + \frac{\xi}{2}}{2} \right) + \theta \right) \end{aligned}$$

where $\eta \triangleq \Delta\omega T/2\pi = \Delta f T$. Finally, substituting Eqs. (10-131) and (10-132) into Eq. (10-130) and performing the average over the data symbols gives, after some trigonometric simplification,[15]

$$
\begin{aligned}
g_n(\lambda) \triangleq & \frac{g(\lambda)}{PT^2} \\
= & \frac{1}{2}(1-\lambda)\left(\frac{\xi}{2}+\lambda\right)\left(\frac{\sin\pi\eta(1-\lambda)}{\pi\eta(1-\lambda)}\right)\left(\frac{\sin\pi\eta\left(\frac{\xi}{2}+\lambda\right)}{\pi\eta\left(\frac{\xi}{2}+\lambda\right)}\right)\cos\pi\eta\left(1-\frac{\xi}{2}\right) \\
& -\frac{1}{2}\lambda\left(\frac{\xi}{2}+\lambda\right)\left(\frac{\sin\pi\eta\lambda}{\pi\eta\lambda}\right)\left(\frac{\sin\pi\eta\left(\frac{\xi}{2}+\lambda\right)}{\pi\eta\left(\frac{\xi}{2}+\lambda\right)}\right)\cos\pi\eta\frac{\xi}{2} \\
& -\frac{1}{2}(1-\lambda)\left(\frac{\xi}{2}-\lambda\right)\left(\frac{\sin\pi\eta(1-\lambda)}{\pi\eta(1-\lambda)}\right)\left(\frac{\sin\pi\eta\left(\frac{\xi}{2}-\lambda\right)}{\pi\eta\left(\frac{\xi}{2}-\lambda\right)}\right)\cos\pi\eta\left(1-\frac{\xi}{2}\right); \\
& \qquad 0 \le \lambda \le \frac{\xi}{2}
\end{aligned}
\tag{10-133}
$$

In the case of no frequency offset ($\eta = 0$), the S-curve of Eq. (10-133) simplifies to Eq. (10-112) as it should.

Taking the derivative of Eq. (10-133) with respect to λ and evaluating the result at $\lambda = 0$ gives the slope of the S-curve at the origin as

$$
\begin{aligned}
K_g \triangleq \frac{dg(\lambda)}{d\lambda}\Big|_{\lambda=0} &= PT^2\left\{\left(\frac{\sin\pi\eta\xi/2}{\pi\eta\xi/2}\right)\right. \\
&\left.\times\left[1+\left(\frac{\sin\pi\eta}{\pi\eta}\right)\cos\pi\eta(1-\xi/2)-\frac{\xi}{4}\cos\pi\eta\xi/2\right]-\cos\pi\eta\xi/2\right\}
\end{aligned}
\tag{10-134}
$$

which in the case of no frequency offset ($\eta = 0$) simplifies to Eq. (10-113).

---

[15] For the sake of brevity, we do not present the result for the S-curve in the region $\xi/2 \le \lambda \le 1/2$ since for the purpose of mean-squared timing-error evaluation we have already seen that only the slope of the S-curve at the origin ($\lambda = 0$) is needed.

10.7.2 Noise Performance

As before, we must determine the PSD at $\lambda = 0$ of the equivalent noise, $n_\lambda(t)$, perturbing the loop as given by Eq. (10-30). Following an analogous approach to that taken for the zero frequency offset case, the following results are obtained:

$$E_{n,s}\left\{e_k^2\,|_{\lambda=0}\right\} = P^2T^4\left(\frac{\xi}{4R_s}\right)\left[\left(\frac{\sin\pi\eta}{\pi\eta}\right)^2 + \frac{\xi}{2}\left(\frac{\sin\pi\eta\xi/2}{\pi\eta\xi/2}\right)^2 + \frac{1}{R_s}\right] \tag{10-135}$$

$$E_{n,s}\left\{e_k e_{k+1}\,|_{\lambda=0}\right\} = \begin{cases} -P^2T^4\left(\dfrac{\xi^2}{32R_s}\right)\cos 2\pi\eta\,(1-\xi/2)\,, & m=1 \\ 0, & m>1 \end{cases}$$

where as before $R_s \triangleq PT/N_0$ denotes the detection symbol SNR. Substituting Eq. (10-135) into Eq. (10-30) gives the desired equivalent PSD as

$$N_0' = P^2T^5\left(\frac{\xi}{2R_s}\right)\left\{\left(\frac{\sin\pi\eta}{\pi\eta}\right)^2 + \frac{\xi}{2}\left[\left(\frac{\sin\pi\eta\xi/2}{\pi\eta\xi/2}\right)^2 - \frac{1}{2}\cos 2\pi\eta\,(1-\xi/2)\right] + \frac{1}{R_s}\right\} \tag{10-136}$$

10.7.3 Mean-Squared Timing-Error Performance

The mean-squared timing error σ_λ^2 of the LDTTL in the presence of frequency offset is now readily computed using the results in Eqs. (10-134) and (10-136) in Eq. (10-39), resulting in

$$\sigma_\lambda^2 = \frac{\xi\left\{\left(\dfrac{\sin\pi\eta}{\pi\eta}\right)^2 + \dfrac{\xi}{2}\left[\left(\dfrac{\sin\pi\eta\frac{\xi}{2}}{\pi\eta\frac{\xi}{2}}\right)^2 - \dfrac{1}{2}\cos 2\pi\eta\left(1-\dfrac{\xi}{2}\right)\right] + \dfrac{1}{R_s}\right\}}{2\rho\left\{\left(\dfrac{\sin\pi\eta\frac{\xi}{2}}{\pi\eta\frac{\xi}{2}}\right)\left[1 + \left(\dfrac{\sin\pi\eta}{\pi\eta}\right)\cos\pi\eta\left(1-\dfrac{\xi}{2}\right) - \dfrac{\xi}{4}\cos\pi\eta\dfrac{\xi}{2}\right] - \cos\pi\eta\dfrac{\xi}{2}\right\}^2} \tag{10-137}$$

For zero frequency offset ($\eta = 0$), Eq. (10-137) simplifies to Eq. (10-117).

To demonstrate the additional degradation in performance of the NC-LDTTL due to frequency offset, Fig. 10-22 plots the mean-squared timing-error ratio $\sigma_\lambda^2/\sigma_\lambda^2\,|_{\Delta f=0}$ in dB versus SNR in dB for several different values of normalized frequency offset, $\eta = \Delta fT$, and two different values of normalized window width, ξ. We observe from these results that over a wide range of SNRs the degradation is virtually insensitive to this parameter. Furthermore, the degradation also appears to be relatively insensitive to window width. Finally, for values of normalized frequency offset less than 0.1, it can be observed that the performance degradation is quite small (i.e., less than 0.5 dB).

10.7.4 A Final Note

Thus far in our discussion of the noncoherent DTTL in this section we have assumed an input signal in the form of BPSK modulation. Before concluding this section, we wish to point out that it is possible to apply the same MAP estimation approach to QPSK to arrive at LFs that can be used to motivate closed-loop symbol synchronizers for this modulation. Without going into detail, it can be shown that, under the same approximations used to derive the low and high SNR versions of the noncoherent DTTLs for BPSK, the LFs that result from this approach when applied to QPSK are identical to those obtained for BPSK. Equivalently stated, noncoherent symbol synchronization of QPSK with a DTTL type of structure takes on the exact same form as that for BPSK.

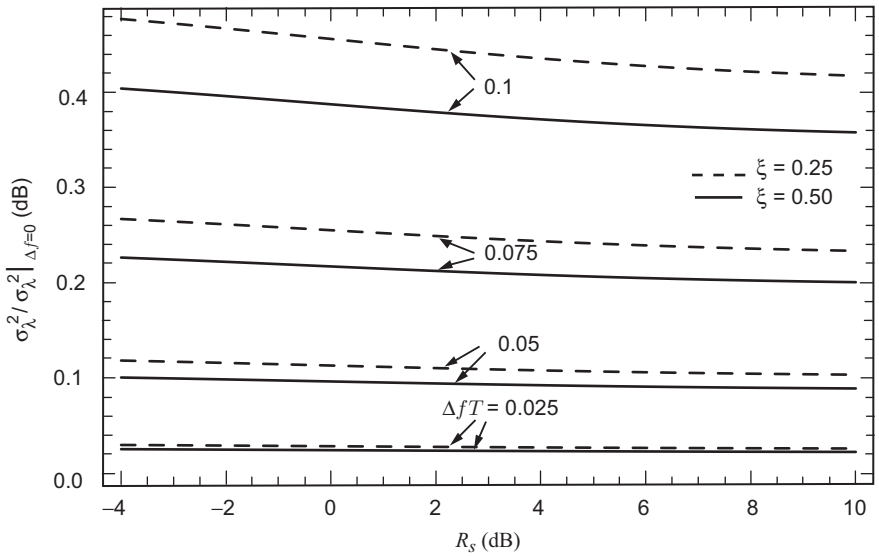


Fig. 10-22. Mean-squared timing-error performance degradation versus detection SNR with normalized frequency error as a parameter.

10.8 Coarse Estimation of Symbol Timing for Use in SNR Estimation

In our discussions of SNR estimation for autonomous receiver operation in Chapter 6, we considered the behavior of the split-symbol moments estimator (SSME) in the presence of unknown symbol timing. Although originally derived on the basis of perfect symbol sync information, it was shown there that, in the presence of unknown symbol timing, the performance of a conventional SSME becomes quite sensitive to the amount of symbol-timing error. In fact, in order to properly operate at all, the form of the estimator now requires knowledge (an estimate) of the symbol timing [via the parameters $\hat{h}^+$ and $\hat{h}^-$ in Eq. (6-54)]. Thus, it is essential that one provide a coarse estimate of symbol timing to the SNR estimator,[16] preferably derived from the same statistics as those used to form this estimator itself.

Using the relations for the mean of the sums U^+ and U^- of the squared magnitudes of the sum and difference half-symbol I&D outputs [see Eqs. (6-49) and (6-51)], we have

$$E\left\{U^+\right\} - E\left\{U^-\right\} = 2\sigma^2\left(\overline{h}^+ - \overline{h}^-\right)R = 2\sigma^2\left(1 - 2\left|\varepsilon\right|\right)R \tag{10-138}$$

While Eq. (10-138) provides the needed measure of the symbol timing, it also depends on σ^2 and R, which are parameters that themselves need to be estimated. Thus, it behooves us to find another measure of the product of σ^2 and R that is independent of the symbol sync timing, which could then be used together with Eq. (10-138) to isolate the $1 - 2\left|\varepsilon\right|$ factor.

Also considered in Chapter 6 was a modification of the conventional SSME that produced a set of observables obtained from oversampling the symbol interval by a factor L and as such was capable of providing improved performance at SNRs above 3 dB. It was also shown there that this same modification had the added advantage of reducing the sensitivity of the estimator performance to incorrect symbol-timing information and, in the limit of sufficiently large L, became completely insensitive to knowledge of ε. In fact, for L approaching infinity, the difference of the means considered in Eq. (10-138) now became [see Eq. (6-110)]

$$E\left\{U^+\right\} - E\left\{U^-\right\} = 2\sigma^2 R \tag{10-139}$$

[16] Actually, because of the noncoherent nature of the SNR estimator, it requires only an estimate of the *magnitude* of the symbol sync timing.

Hence, taking the ratio of Eq. (10-138) to Eq. (10-139) gives

$$\frac{\left(E\left\{U^{+}\right\}-E\left\{U^{-}\right\}\right)|_{L=1}}{\left(E\left\{U^{+}\right\}-E\left\{U^{-}\right\}\right)|_{L=\infty}}=1-2\left|\varepsilon\right| \tag{10-140}$$

which is the desired result. Thus, it follows from Eq. (10-140) that a coarse estimator of symbol timing based on the same statistics used to produce the SNR estimator is given by

$$\left|\hat{\varepsilon}\right|=\frac{1}{2}\left[1-\frac{\left(U^{+}-U^{-}\right)|_{L=1}}{\left(U^{+}-U^{-}\right)|_{L=\infty}}\right] \tag{10-141}$$

Since in practice one deals with a finite oversampling factor, what follows next is a quantification of the difference of means $E\left\{U^{+}\right\}-E\left\{U^{-}\right\}$ as a function of L to enable one to determine how large one must make L in order to reach the limiting value as in Eq. (10-139).

For a given oversampling factor L, we can characterize the fractional (with respect to T) symbol timing by $\varepsilon=\left(l_{\varepsilon}+1/2+\delta\right)/L,\ l_{\varepsilon}=0,1,2,\cdots,L-1$, where l_{ε} represents the integer number of symbol interval subdivisions and $|\delta|\leq 1/2$ represents the remaining fraction of a subdivision. Then, it is straightforward to see that for $L-1$ subdivisions both the entire first and second half-symbol I&Ds correspond to the same symbol, and thus the contribution of each of these subdivisions to the mean-squared accumulations U^{+} and U^{-} *is independent of* ε! More explicitly, for $l=1,2,\cdots,L-l_{\varepsilon}-1$, assuming for simplicity no frequency uncertainty, the first and second half-symbol I&D outputs are given by (see Section 6.11 for a description of the notation)

$$\begin{aligned}Y_{\alpha kl}&=\frac{md_{k}}{2}e^{j\phi}+n_{\alpha kl}\\ Y_{\beta kl}&=\frac{md_{k}}{2}e^{j\phi}+n_{\beta kl}\end{aligned} \tag{10-142}$$

and similarly for $l=L-l_{\varepsilon}+1,L-l_{\varepsilon}+2,\cdots,L$,

$$\begin{aligned}Y_{\alpha kl}&=\frac{md_{k+1}}{2}e^{j\phi}+n_{\alpha kl}\\ Y_{\beta kl}&=\frac{md_{k+1}}{2}e^{j\phi}+n_{\beta kl}\end{aligned} \tag{10-143}$$

where $n_{\alpha kl}$ and $n_{\beta kl}$ are complex Gaussian zero-mean noise variables with variance $\sigma^2 L$. This leaves one subdivision, i.e., the one corresponding to $l = L - l_\varepsilon$, to be affected by the lack of symbol sync knowledge. For this one subdivision, the first and second half-symbol I&D outputs are given by

$$\begin{aligned} Y_{\alpha kl}\,|_{l=L-l_\varepsilon} &= \frac{m}{2}\left[d_k\left(\frac{1}{2}-\delta\right)+d_{k+1}\delta\right]e^{j\phi}+n_{\alpha kl}\,|_{l=L-l_\varepsilon}, \\ Y_{\beta kl} &= \frac{md_{k+1}}{2}e^{j\phi}+n_{\beta kl}; \quad 0\le\delta\le\frac{1}{2} \\ Y_{\alpha kl}\,|_{l=L-l_\varepsilon} &= \frac{md_k}{2}e^{j\phi}+n_{\alpha kl}\,|_{l=L-l_\varepsilon}, \\ Y_{\beta kl} &= \frac{m}{2}\left[d_k\,|\delta|+d_{k+1}\left(\frac{1}{2}-|\delta|\right)\right]e^{j\phi}+n_{\beta kl}; \quad -\frac{1}{2}\le\delta\le 0 \end{aligned} \tag{10-144}$$

Forming the sum and difference signals from Eqs. (10-142) through (10-144), we get

$$\begin{aligned} u_{kl}^{+} &= md_k e^{j\phi}+n_{\alpha kl}+n_{\beta kl} \\ u_{kl}^{-} &= n_{\alpha kl}+n_{\beta kl}, \quad l=1,2,\cdots,L-l_\varepsilon-1 \\ u_{kl}^{+} &= md_{k+1}e^{j\phi}+n_{\alpha kl}+n_{\beta kl} \\ u_{kl}^{-} &= n_{\alpha kl}+n_{\beta kl}, \quad l=L-l_\varepsilon+1,L-l_\varepsilon+2,\cdots,L \\ u_{kl}^{\pm} &= m\left[d_k\left(\frac{1}{2}-\delta\right)+d_{k+1}\left(\delta\pm\frac{1}{2}\right)\right]e^{j\phi}+n_{\alpha kl}\pm n_{\beta kl}, \\ &\qquad 0\le\delta\le\frac{1}{2}, \quad l=L-l_\varepsilon \\ u_{kl}^{\pm} &= m\left[d_k\left(\frac{1}{2}\pm|\delta|\right)\pm d_{k+1}\left(|\delta|-\frac{1}{2}\right)\right]e^{j\phi}+n_{\alpha kl}\pm n_{\beta kl}, \\ &\qquad -\frac{1}{2}\le\delta\le 0, \quad l=L-l_\varepsilon \end{aligned} \tag{10-145}$$

Finally, the means of $U^{\pm}=(1/NL)\sum_{k=1}^{N}\sum_{l=1}^{L}\left|u_{kl}^{\pm}\right|^2$ are easily shown to be

$$E\left\{U^{+}\right\} = 2\sigma^2\left[L + R\left(1 - \frac{\frac{1}{2} - 2|\delta|^2}{L}\right)\right]$$

$$E\left\{U^{-}\right\} = 2\sigma^2\left[L + R\left(\frac{\frac{1}{2} - 2|\delta| + 2|\delta|^2}{L}\right)\right], \quad 0 \le |\delta| \le \frac{1}{2} \tag{10-146}$$

which agrees with the results in Eq. (6-110) when $|\delta| = 1/2$, i.e., a fractional (with respect to a subdivision of T) timing error equal to $1/2$.

References

[1] M. K. Simon, "An Analysis of the Steady-State Phase Noise Performance of a Digital Data-Transition Tracking Loop," *ICC '69 Conference Record*, Boulder, Colorado, pp. 20-9–20-15, June 1969. Also see M. K. Simon, *An Analysis of the Steady-State Phase Noise Performance of a Digital Data-Transition Tracking Loop*, JPL Technical Report 900-222, Jet Propulsion Laboratory, Pasadena, California, November 21, 1968.

[2] W. C. Lindsey and M. K. Simon, *Telecommunication Systems Engineering*, Englewood Cliffs, New Jersey: Prentice-Hall, 1973. Reprinted by Dover Press, New York, 1991.

[3] H. Cramer, *Mathematical Methods of Statistics*, Princeton, New Jersey: Princeton University, 1951.

[4] C. Rao, *Linear Statistical Inference and Its Application*, New York: John Wiley & Sons, 1965.

[5] H. V. Trees, *Detection, Estimation, and Modulation Theory*, New York: John Wiley & Sons, 1968.

[6] J. Ziv and M. Zakai, "Some Lower Bounds on Signal Parameter Estimation," *IEEE Transactions on Information Theory*, vol. IT-15, pp. 386–391, May 1969.

[7] D. Chasan, M. Zakai, and J. Ziv, "Improved Lower Bounds on Signal Parameter Estimation," *IEEE Transactions on Information Theory*, vol. IT-21, pp. 90–93, January 1975.

[8] I. Ibragimov and R. Khas′minskii, "Parameter Estimation for a Discontinuous Signal in White Gaussian Noise," *Problemy Peredachi Informatsii*, vol. 11, pp. 31–43, July–September 1975, as translated in *Problems of Information Transmission*, pp. 203–212, September 1976.

[9] A. Terent′yev, "Distribution of the Time of Absolute Maximum at the Output of a Matched Filter," *Radio Engineering and Electronic Physics*, vol. 13, no. 4, pp. 569–573, 1968.

[10] K. Kosbar, *Open and Closed Loop Delay Estimation with Applications to Pseudonoise Code Tracking*, Ph.D. thesis, University of Southern California, Los Angeles, California, July 1988.

[11] J. Holmes, "Tracking Performance of the Filter and Square Bit Synchronizer," *IEEE Transactions on Communications*, vol. COM-28, pp. 1154–1158, August 1980.

[12] R. D. McCallister and M. K. Simon, "Cross-Spectrum Symbol Synchronizer," *ICC'81 Conference Record*, pp. 34.3.1–34.3.6, 1981.

[13] S. Haykin, *Communication Systems*, New York: John Wiley & Sons, 4th ed., 2001.

Chapter 11
Implementation and Interaction of Estimators and Classifiers

Jon Hamkins and Hooman Shirani-Mehr

In each of the previous chapters, a method was proposed to estimate or classify a given signal parameter based on observations of the received signal. When tractable, the estimation or classification was derived from the maximum-likelihood (ML) principle, i.e., the parameter was estimated in the way that would best explain the observations. When ML solutions were impractical, reduced-complexity approximations to the ML solutions, or ad hoc estimators/classifiers were proposed.

In this final chapter, we explain how the algorithms of the previous chapters may be incorporated into a single, practical, and operational autonomous radio. This chapter is the bookend to Chapter 1, once again addressing the overall architecture of the autonomous radio and the interactions of its components. In particular, we summarize the algorithms that result from the analysis of the earlier chapters, show their interdependence, and construct an explicit sequence of coarse and fine estimation/classification that accomplishes all the functions of an autonomous radio. We have used the technique outlined in this chapter to write a software implementation of an autonomous radio that successfully identifies and processes an Electra-like signal with unknown attributes.

The chapter is organized as follows. In Section 11.1, we review the conventional approach to converting continuous-time signal processing to the discrete-time processing appropriate for a software implementation. In Section 11.2, we review a sequence of estimator/classifier actions that provides the first, coarse estimates of signal parameters, and a technique to refine the estimates by feeding back the coarse estimates to all the estimator modules. The individual estima-

tion/classification modules are discussed in the order in which they process the signal, including modulation index estimation in Section 11.3.1.

11.1 Signal Model

The complex baseband representation of the received signal is given by Eq. (1-8), which we restate here:

$$\tilde{r}(t) = \sqrt{2P_d} \sum_{l=-\infty}^{\infty} d_l(t) p[t - lT - \varepsilon T] e^{j[\omega_r t + \theta_c]} + \sqrt{2P_c} e^{j[\omega_r t + \theta_c(t)]} + \tilde{n}(t) \quad (11\text{-}1)$$

In order to process this continuous-time signal digitally, we sample the signal at regular time intervals, separated by T_s seconds.

Although there is nothing new about sampling a continuous-time signal, it is helpful to be explicit about the variance of the noise samples, which is related to the bandwidth of Eq. (1-1). If we follow the convention we have used throughout the monograph that $\tilde{n}(t) = \sqrt{2}\big(n_c(t) + jn_s(t)\big)$ and that each of $n_c(t)$ and $n_s(t)$ has two-sided power spectral density (PSD) $N_0/2$, we quickly see that a *sample* $\tilde{n}(kT_s)$ will have variance $R_{\tilde{n}}(0) = 2N_0\delta(0)$ per dimension, which is not bounded.

Instead, following standard practice [1], we implicitly assume that the passband signal $r(t)$ in Eq. (1-1) can be sent through an ideal passband filter that introduces negligible distortion to the signal but which eliminates the noise frequency components outside the passband. If the filter has bandwidth ω_s and is centered at the carrier ω_c, then the passband noise $n(t)$ at the filter output has PSD

$$S_n(\omega) = \begin{cases} N_0/2, & |\omega - \omega_c| \le \omega_s/2 \text{ or } |\omega + \omega_c| \le \omega_s/2 \\ 0, & \text{otherwise} \end{cases} \quad (11\text{-}2)$$

In complex baseband, the PSD of each of $n_c(t)$ and $n_s(t)$ is given by

$$S_{n_c}(\omega) = S_{n_s}(\omega) = \begin{cases} N_0/2, & |\omega| \le \omega_s/2 \\ 0, & |\omega| > \omega_s/2 \end{cases} \quad (11\text{-}3)$$

(Note that the PSD of each of $\sqrt{2}n_c(t)$ and $\sqrt{2}n_s(t)$ is twice this amount.) If $\omega_s = 2\pi/T_s$, then based on the above, each passband noise sample $n(kT_s)$ has variance N_0/T_s, and each complex baseband noise sample $\tilde{n}(kT_s)$ also has variance N_0/T_s in each dimension.

Thus, we model the discrete-time complex baseband signal as

$$\tilde{r}[k] \triangleq \sqrt{2P_d} \sum_{l=-\infty}^{\infty} d_k(kT_s)p[kT_s - lT - \varepsilon T]e^{j[\omega_r kT_s + \theta_c]}$$

$$+ \sqrt{2P_c}e^{j[\omega_r kT_s + j\theta_c(kT_s)]} + \tilde{n}(kT_s) \tag{11-4}$$

where each noise complex sample $\tilde{n}(kT_s)$ has variance N_0/T_s per dimension.

After reception of the baseband signal in Eq. (11-4), the signal is separated into its data-modulated and residual carrier components. We assume that, when a residual carrier signal is used, its spectrum is distinct enough from that of the data-modulated portion of the signal that these two components may be ideally separated. A high-pass filter extracts the data-modulation component, while a low-pass filter (LPF) extracts the residual carrier component:

$$\tilde{r}_d[k] \triangleq \sqrt{2P_d} \sum_{l=-\infty}^{\infty} d_k(kT_s)p[kT_s - lT - \varepsilon T]e^{j[\omega_r kT_s + \theta_c]} + \tilde{n}'(kT_s) \tag{11-5}$$

$$\tilde{r}_c[k] \triangleq \sqrt{2P_c}e^{j[\omega_r kT_s + j\theta_c(kT_s)]} + \tilde{n}''(kT_s) \tag{11-6}$$

The complex process $\tilde{n}'(kT_s)$ is the same as $\tilde{n}(kT_s)$ except that it has had a notch of spectrum removed. Generally the notch is small relative to the overall bandwidth, and so the complex sample $\tilde{n}'(kT_s)$ can be assumed to have variance N_0/T_s per dimension, as before.

The situation for $\tilde{n}''(kT_s)$ is a little more complicated. If the low-pass bandwidth ω_{lp} satisfies $\omega_{lp} < 2\pi/T_s$, then sampling at rate T_s results in a correlated $\tilde{n}''(kT_s)$ noise sequence. To avoid this situation, we may sample the residual carrier component at a different frequency, every $T_c = \omega_{lp}/(2\pi)$ seconds. Redefining $r_c[k]$ in this way results in

$$\tilde{r}_c[k] \triangleq \sqrt{2P_c}e^{j[\omega_r kT_c + j\theta_c(kT_c)]} + \tilde{n}''(kT_c) \tag{11-7}$$

where now $\{\tilde{n}''(kT_c)\}$ is an *uncorrelated* complex noise sequence with variance N_0/T_c per dimension.

11.2 Interaction of Estimator and Classifiers

Chapter 1 described a sequence of operations that accomplishes the task of estimating all the signal parameters shown in Table 11-1. This order is summarized in Fig. 1-3. We now become more explicit about the inputs and outputs from each of the modules and how they are connected.

Table 11-1. Signal parameters to be estimated and classified.

Notation	Description
β	Modulation index
ω_r	Residual frequency
T	Symbol duration
$p(t)$	Data format/pulse shape
R	Signal-to-noise ratio
ε	Fractional symbol timing
θ_c	Carrier phase
M	Modulation order (in M-PSK signals)

The autonomous radio begins in a *coarse estimation* phase in which the signal flow is unidirectional, without feedback. In coarse estimation, shown in Fig. 11-1, each estimation module has parameter estimates from the modules to its left, but no parameter estimates from the modules on its right. For example, the modulation index estimator operates without knowledge of the modulation type. Of course, every module has access to the $\tilde{r}[k]$ observables. Thus, in coarse estimation we employ the algorithms that require the least parameter information, which results in worse performance as compared to conventional receivers that rely on known attributes of the signal.

After the coarse estimation phase, estimates of all signal parameters are available to all blocks in the subsequent iterations. Therefore, the better-performing estimation techniques can be used in the *fine estimation* phase. For example, the modulation index estimator can make use of the modulation order $\hat{M}$, the data-transition tracking loop (DTTL) can make use of $\hat{\theta}_c$ to operate coherently, and so on. In the fine estimation phase, the modules can iteratively update their estimates until convergence takes place.

A complete functional diagram of both the coarse and fine estimation phases is shown in Fig. 11-2. In coarse estimation, the switches are in the "coarse" position. After all parameter estimates are available, the switches may be placed in the "fine" position. The received discrete-time signal is shown on the left. It enters the modulation index classifier and then, depending on the result, the appropriate structure is used to correct any residual frequency. Following this, the residual carrier (if any is detected) and data components are separated using low- and high-pass filters, and the joint estimator for data rate, data format, signal-to-noise ratio (SNR), and coarse symbol synchronization (sync) is run. These parameters, along with the carrier and data signal, are fed to the carrier

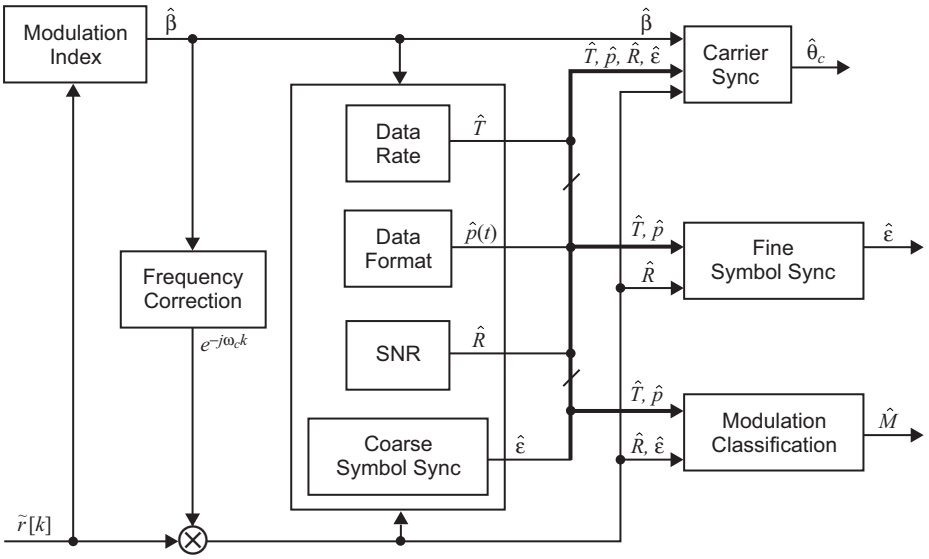


Fig. 11-1. Signal flow in coarse estimation.

sync, fine symbol sync, and modulation classifier structures. Depending on the modulation index, one of several carrier loops may be chosen. When carrier lock is detected, the coherent versions of any of the estimators may be used, as indicated in Fig. 11-2.

11.3 Coarse and Fine Estimators/Classifiers

In this section, we summarize the operation of the individual modules as they operate in coarse and fine modes.

11.3.1 Modulation Index Estimation

In the coarse estimation phase, the modulation index estimator requires no parameter estimates, other than the minimum symbol period T^*. It directly estimates the carrier and data powers by integrating over a sufficient epoch. In the fine estimation phase, it may operate coherently and use knowledge of the date rate, modulation type, symbol boundaries, and pulse shape to improve its performance.

The coarse estimator is given by Eq. (3-49), which becomes, after transforming to discrete-time,

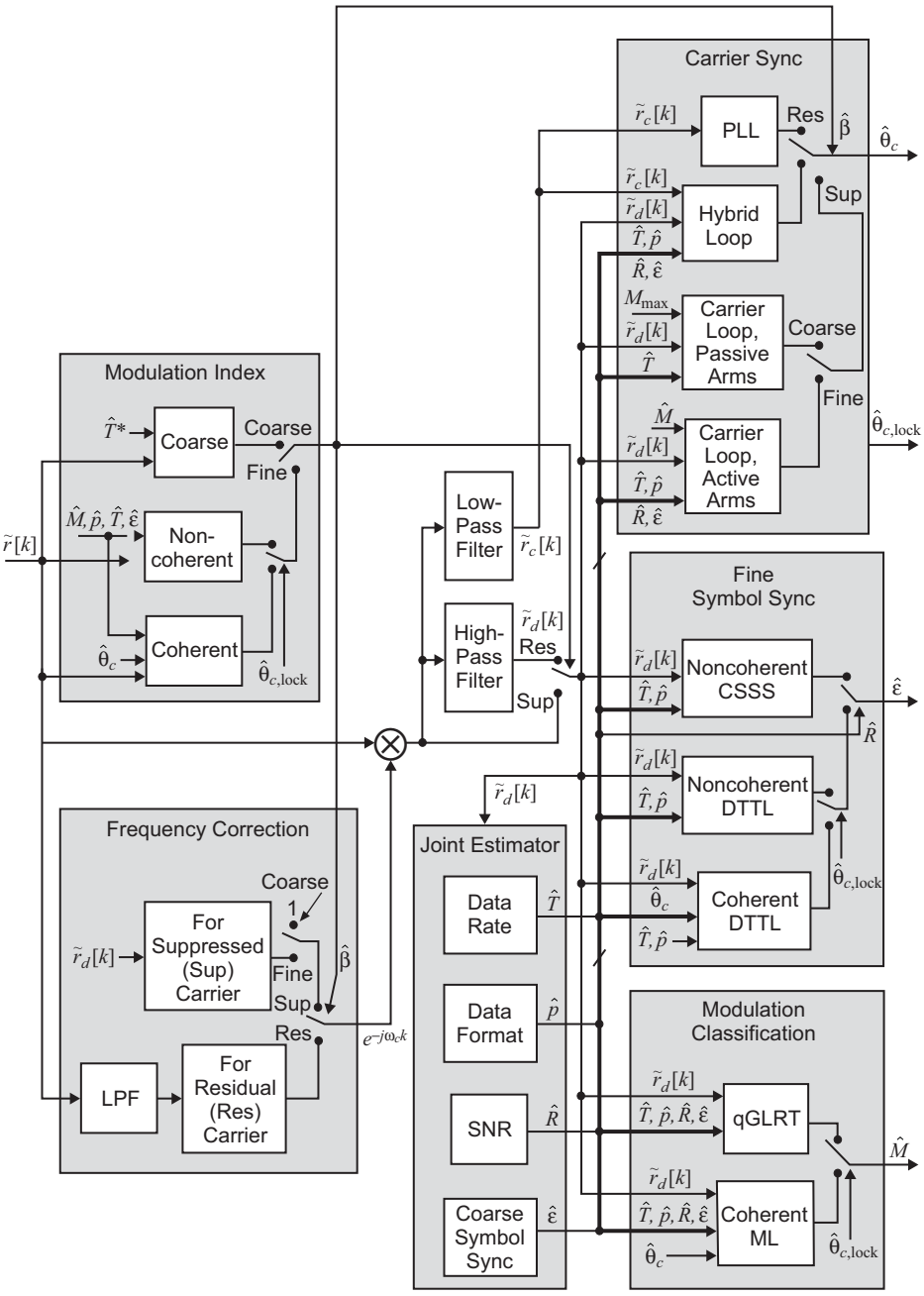


Fig. 11-2. A more detailed functional diagram of the autonomous radio architecture.

$$\hat{\beta} = \cot^{-1}\left[\sqrt{\frac{2\left[\left(\sum_{k=0}^{K'-1} y_c[k]\right)^2 + \left(\sum_{k=0}^{K'-1} y_s[k]\right)^2\right]}{K'\sum_{k=0}^{K'-1}\left[\left(y_c[k] - y_c[k - T^*/T_s]\right)^2 + \left(y_s[k] - y_s[k - T^*/T_s]\right)^2\right]}}\right] \tag{11-8}$$

where $K' = KT^*/T_s$, and $y_c[k] = y_c(kT_s)$ and $y_s[k] = y_s(kT_s)$, which are defined in Eq. (3-42).

In the fine estimation phase, we may replace T^* in Eq. (11-8) by the estimate $\hat{T}$ at the output of the data rate estimator. Also, timing estimates will allow us to define y_c and y_s as in Eq. (3-32) instead of as in Eq. (3-42), i.e., with the symbol-timing offset removed, which improves the fidelity of summations in Eq. (11-8).

If the modulation type is estimated to be binary phase-shift keying (BPSK) in the coarse phase, we may improve the modulation index estimate in the fine estimation phase by using the modulation index estimator for coherent BPSK in Eq. (3-9), which in discrete-time becomes

$$\cot\hat{\beta} = \frac{\sum_{k=0}^{K\hat{T}/T_s-1} y_s[k]}{\sum_{k=0}^{K-1}\left|\sum_{l=k\hat{T}/T_s}^{(k+1)\hat{T}/T_s} y_c[l]p\left[l - k\hat{T}/T_s\right]\right|} \tag{11-9}$$

If the carrier tracking loop is not yet in lock, then the noncoherent BPSK modulation index estimator from Eq. (3-39) may be used. If the modulation type is a higher-order M-ary phase-shift keying (M-PSK), we may use a discrete-time version of Eq. (3-29).

11.3.2 Frequency Correction

Frequency correction is performed in the coarse estimation phase only if the modulation index estimator determines that the signal contains a residual carrier ($\hat{\theta}_c < \pi/2$). In that case, the technique of Section 4.1 can be applied. The scheme for a residual carrier signal is illustrated in Fig. 11-3.

In the fine estimation phase, when the symbol rate $1/T$, fractional symbol timing ε, and pulse shape $p(t)$ are known, the techniques of Section 4.2 or Section 4.3 may be used. Figure 11-4 illustrates one of these schemes. The input to the block diagram is the output of a matched filter, which requires prior knowledge of T, M, ε, and $p(t)$.

Examples of the acquisition and tracking performance are illustrated in Fig. 11-5. In Fig. 11-5(a), there is no residual frequency, and this is tracked

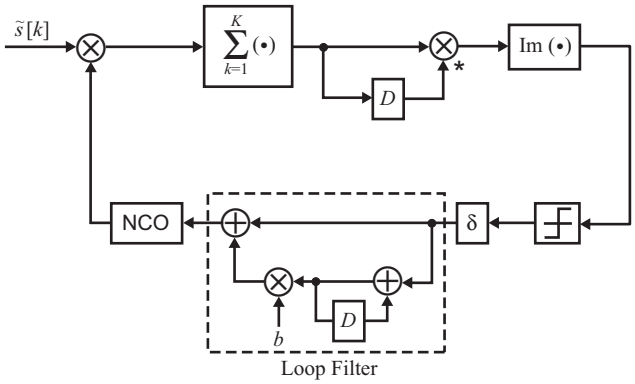


Fig. 11-3. Closed-loop frequency correction for a residual carrier signal (NCO = numerically controlled oscillator).

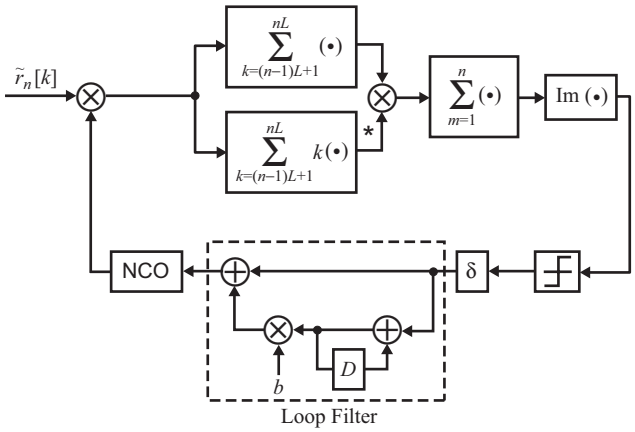


Fig. 11-4. Closed-loop frequency correction for a suppressed-carrier signal.

quite readily. In Fig. 11-5(b), the residual frequency is a half percent of the sample rate, and we see a short acquisition period followed by active tracking. In Fig. 11-5(c), the residual frequency is one percent of the sampling rate, and we see a longer acquisition period.

11.3.3 Joint Estimation of Data Rate, Data Format, SNR, and Coarse Symbol Timing

As shown in Fig. 11-2, the joint estimator for data rate, data format, SNR, and coarse symbol synchronization operates in the same way during both coarse and fine estimation phases. This is a consequence of the fact the the split-symbol moments estimator (SSME) for SNR estimation is independent of both

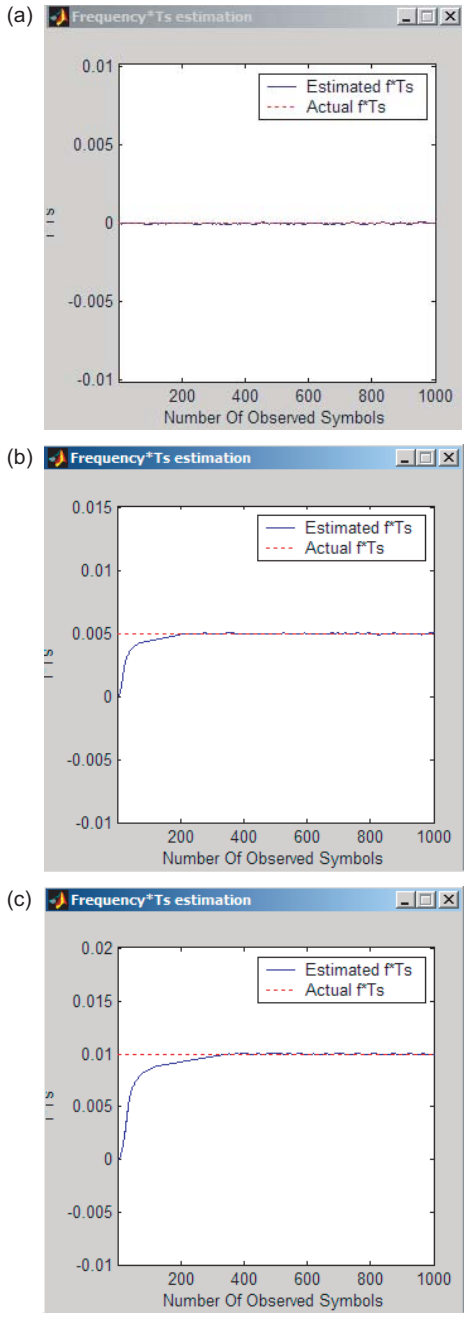


Fig. 11-5. Dynamic response of the frequency-tracking loop for a signal with SNR = 10 dB, $\theta_c = \pi/4$**: (a)** $\omega_r T_s/(2\pi) = 0$**, (b)** $\omega_r T_s/(2\pi) = 0.005$**, and (c)** $\omega_r T_s/(2\pi) = 0.01$**.**

the M-PSK modulation order and the carrier phase θ_c. This module takes the frequency-corrected version of $\tilde{r}[k]$ as input and produces $\hat{T}$, $\hat{p}(t)$, $\hat{R}$, and $\hat{\varepsilon}$ as output.

Figure 11-6 illustrates an example of the performance of the SNR estimator, assuming the residual frequency and symbol timing are known ($\omega_r = 0$ and $\varepsilon = 0$). The simulated signal used BPSK modulation, a symbol period of $T = 15$ μs, and a non-return to zero (NRZ) data format, also assumed known. The asterisk on Fig. 11-6 represents the estimator output, while the line (corresponding to $x = y$) is the desired output. With an observation of 100 samples, the SNR estimator is seen to perform quite well for this example.

11.3.4 Modulation Classification

The various modulation classifiers discussed in Chapter 9 each require knowledge of the data rate, pulse shape, and symbol timing in order to form a matched-filter output, i.e., the single-sample per symbol statistic $\tilde{r}_n$. That statistic is obtained by summing up L received samples, where L is the ratio of the sample rate to the symbol rate:

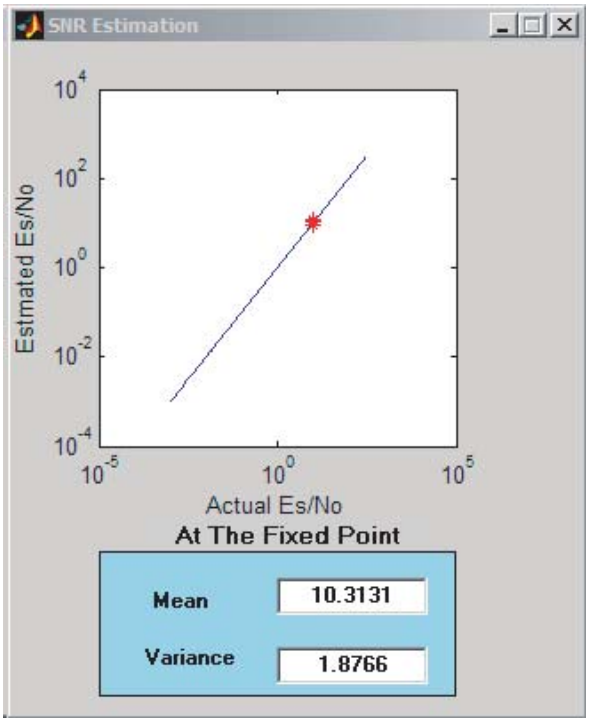


Fig. 11-6. Example output from an SNR estimator.

$$\tilde{r}_n = \sum_{k=(n-1)L+1}^{nL} \tilde{r}_d[k], \quad n = 1, 2, \cdots, N \tag{11-10}$$

with $\tilde{r}_d[k]$ given by Eq. (11-5).

The normalized quasi-log-likelihood ratio (nqLLR) modulation classifier does not need to know the SNR, which makes it a good candidate for modulation classification in the coarse estimation phase. However, the nqLLR classifier must still wait for the data rate and symbol-timing estimates to become available for Eq. (11-10) to be computed, at which point the SNR estimate is also available, since it comes from the same joint estimator. Therefore, there is no advantage in the coarse phase to using the nqLLR, even though in principle it requires less knowledge about the signal attributes.

Instead, the modulation classification in the coarse estimation phase is accomplished with the quasi-generalized-likelihood ratio test (qGLRT), discussed in Section 9.2.2. The qGLRT is based on the conditional-likelihood function $\mathrm{CLF}_M(\theta)$, given by Eq. (9-9) evaluated at a best-available estimate of θ_c. As shown in Chapter 9, this produces only a small loss compared to the ML classifier, which computes a full-blown average over θ_c:

$$\mathrm{LF}_M = \frac{M}{2\pi} \int_0^{2\pi/M} \mathrm{CLF}_M(\theta)\, d\theta \tag{11-11}$$

The estimate for θ_c is given by

$$\hat{\theta}_c^{(M)} = \frac{1}{M} \arg \sum_{n=0}^{N-1} \tilde{r}_n^M \tag{11-12}$$

For BPSK we have

$$\hat{\theta}_c^{(2)} = \frac{1}{2} \arg \sum_{n=0}^{N-1} \tilde{r}_n^2 \tag{11-13}$$

and for quadrature phase-shift keying (QPSK) we have

$$\hat{\theta}_c^{(4)} = \frac{1}{4} \arg \sum_{n=0}^{N-1} \tilde{r}_n^4 \tag{11-14}$$

and, thus, the qGLRT classifier metric is

$$\mathrm{LR} = \frac{\mathrm{CLF}_2\left(\hat{\theta}_c^{(2)}\right)}{\mathrm{CLF}_4\left(\hat{\theta}_c^{(4)}\right)} \tag{11-15}$$

If this is greater than unity, the modulation is declared to be BPSK; otherwise, the modulation is declared to be QPSK.

Figure 11-7 illustrates an example comparing the ML, generalized likelihood ratio test (GLRT), quasi-log-likelihood ratio (qLLR), and nqLLR modulation classifiers, when discriminating BPSK from QPSK. The performance is measured by probability of misclassification, which is the probability of deciding either BPSK or QPSK at the receiver when in fact the other modulation was transmitted. In the example, the symbol SNR is $E_s/N_0 = -4$ dB, and perfect residual frequency correction and symbol timing are assumed. As can be seen from the bar chart, even for such a low SNR, correct classification can still be accomplished about 90 percent of the time.

In the fine estimation phase, the phase tracking loop has locked onto the carrier phase θ_c. Thus, we switch to the coherent ML modulation classifier, which is the same as the qGLRT except that the phase estimate $\hat{\theta}_c$ coming from

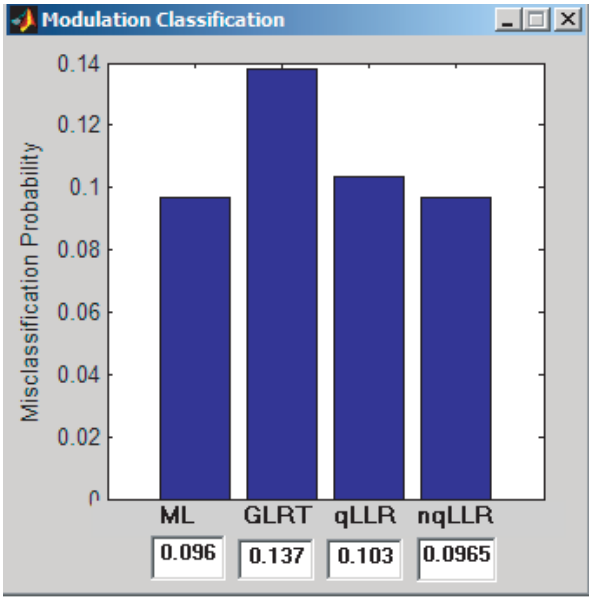


Fig. 11-7. Classification error probability of various classifiers for BPSK/QPSK, where $E_s/N_0 = -4$ dB, $\varepsilon = 0$, $\omega_r = 0$, using $N = 100$ observed symbols.

the carrier tracking loop is used in place of the GLRT estimate. In the case of BPSK/QPSK classification, Eq. (11-15) becomes

$$\mathrm{LR} = \frac{\mathrm{CLF}_2\left(\hat{\theta}_c\right)}{\mathrm{CLF}_4\left(\hat{\theta}_c\right)} \tag{11-16}$$

11.3.5 Carrier Synchronization

As shown in Fig. 11-2, carrier synchronization takes several forms, depending on the modulation index and coarse/fine operation.

If the modulation index estimator has determined that there is a residual carrier, then a phase-locked loop (PLL) may be used to lock onto the residual carrier signal. The residual carrier itself is the output of a low-pass filter of the received signal, which suppresses the data modulation (except for the portion of the spectrum at zero frequency). When a residual carrier is present, the PLL may be the best choice for carrier synchronization in the fine estimation mode as well, but this depends on the SNR and the value of the modulation index. In cases when the residual carrier is weak, a hybrid loop may outperform the PLL alone in the fine estimation phase.

If the modulation index estimator has determined that the carrier is suppressed, then the carrier synchronization in the coarse estimation phase relies on a carrier loop with passive arms. As discussed in Chapter 8, such a loop uses passive filters in each arm, so that the symbol timing and pulse shape need not be known. A universal loop can be constructed that will work for all M-PSK modulation orders up to some maximum $M_{\max}$. This is shown in Fig. 8-12 for $M \in \{2, 4, 8\}$, where the in-phase (I) and quadrature (Q) arm filters can be implemented with simple low-pass filters such as simple integrators. Since modulation classification is not yet available to the carrier loop during the coarse phase, the carrier loop begins in the coarse estimation phase configured for $M_{\max}$-PSK.

For suppressed carrier signals in the fine estimation phase, the switch for M in Fig. 8-12 can be set according to the modulation classifier output, and the passive arm filters can be replaced with matched filters that make use of the pulse shape and symbol-timing estimates, which results in improved performance as discussed in Chapter 8.

We now discuss the conversion of the passband continuous-time loops discussed in Chapter 8 to the discrete-time complex baseband loops suitable for a digital implementation. First we will discuss the continuous model for tracking a BPSK signal. For QPSK signals, the same approach applies.

Assume that the input to the loop is the BPSK suppressed carrier passband signal $r(t)$ in the form of

$$r(t) = s(t) = \sum_{n=1}^{N} a_n p\left[t - (n-1)\,T\right] \sin\left(\omega_c t + \theta_c\right) + n(t) \tag{11-17}$$

where a_n is random binary data corresponding to the nth transmitted BPSK symbol taking on values ± 1 with equal probability and for simplicity $p(t)$ is assumed to be a unit amplitude rectangular pulse. Let

$$\begin{aligned} w_s\,(t) &\triangleq \sin\left(\omega_c t + \hat{\theta}_c\right) \\ w_c\,(t) &\triangleq \cos\left(\omega_c t + \hat{\theta}_c\right) \end{aligned} \tag{11-18}$$

where $\hat{\theta}_c$ is the estimate of θ_c. Therefore, the nth transmitted BPSK symbol in the interval $(n-1)\,T \le t \le nT$ is given by

$$U = \int_{(n-1)T}^{nT} w_s\,(t)\,r\,(t)\,dt$$

Neglecting noise terms and double frequency terms results in

$$U \propto \cos\left(\theta_c - \hat{\theta}_c\right) \tag{11-19}$$

Similarly,

$$V = \int_{(n-1)T}^{nT} w_c\,(t)\,r\,(t)\,dt \propto \sin\left(\theta_c - \hat{\theta}_c\right)$$

Let $\phi \triangleq \theta_c - \hat{\theta}_c$. Then,

$$U \propto \cos\left(\phi\right) \tag{11-20}$$

$$V \propto \sin\left(\phi\right) \tag{11-21}$$

Now, consider $\tilde{r}_n\,[k]$, which is the input of the equivalent complex baseband, discrete-time loop. Assuming perfect carrier frequency estimation ($\omega_r = 0$),

$$\tilde{r}_n\,[k] = \frac{1}{L} e^{j[\theta_n + \theta_c]} + \tilde{n}_n\,[k] \tag{11-22}$$

The estimated signal $\hat{\tilde{r}}$ is the output of the discrete-time voltage-controlled oscillator (VCO) and has the form

$$\hat{\tilde{r}} = e^{-j\hat{\theta}_c} \tag{11-23}$$

where $\hat{\theta}_c$ is the estimate of θ_c. From Eqs. (11-22) and (11-23), the multiplication signal $\tilde{w}_n[k]$ has the form

$$\tilde{w}_n[k] = \tilde{r}_n[k] \times \hat{\tilde{r}} \tag{11-24}$$

$$= \left(\frac{1}{L}e^{j[\theta_n+\theta_c]} + \tilde{n}_n[k]\right) e^{-j\hat{\theta}_c} \tag{11-25}$$

$$= \frac{1}{L}e^{j\theta_n}e^{j\left[\theta_c-\hat{\theta}_c\right]} + \tilde{n}_n[k]\, e^{-j\hat{\theta}_c} \tag{11-26}$$

Since $\tilde{n}[k]$ and $\tilde{n}[k]\, e^{-j\hat{\theta}_c}$ have the same statistical distributions, we may write

$$\tilde{w}_n[k] = \frac{1}{L}e^{j(\theta_n+\phi)} + \tilde{n}_n[k] \tag{11-27}$$

The equivalent of integration in the continuous domain is summation in our domain. Therefore, the integrators are replaced with summations. Also note that, since we are dealing with complex signals, in order to extract the trigonometric functions (sine and cosine) of the signal angle, the loop requires extracting the imaginary and real parts, respectively, of the signal $\tilde{w}(k)$. Applying these modifications results in the loops that are illustrated in Figs. 11-8(a) and 11-8(b), where

$$S : \tilde{y}_n = \sum_{k=(n-1)L+1}^{nL} \tilde{x}_n[k]$$

Note that the loop filter is identical to the continuous case that is discussed in Chapter 8. Hence, the transfer function of the loop filter has the form

$$F(s) = \frac{1+\tau_2 s}{\tau_1 s}, \quad \tau_1 \gg \tau_2 \Rightarrow (\tau_1 = 1, \tau_2 = 0.01) \tag{11-28}$$

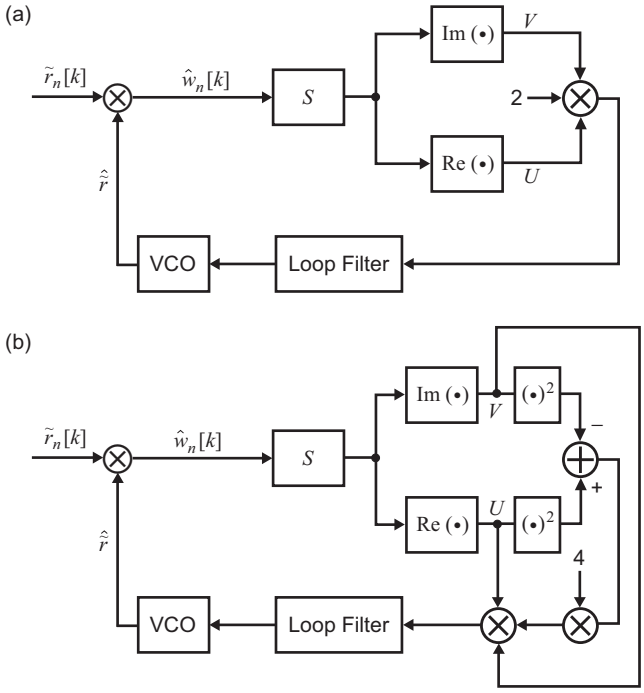


Fig. 11-8. Sampled implementation of a Costas-type loop capable of tracking: (a) BPSK and (b) QPSK.

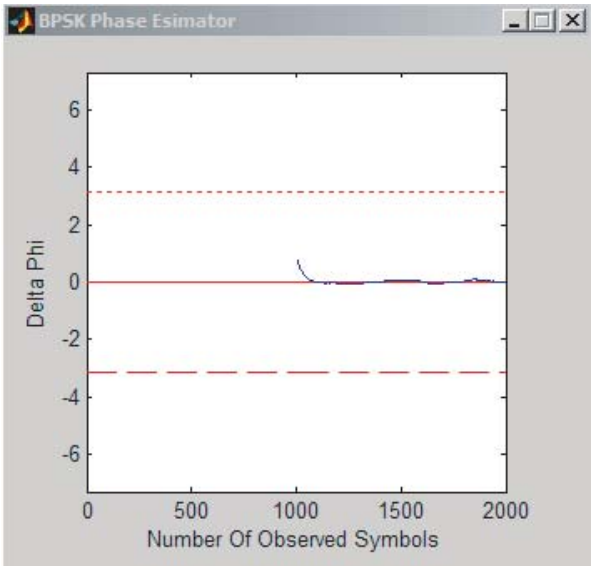
Figures 11-9(a) and 11-9(b) illustrate the sample acquisition behavior of each loop for a set of parameters. In each case, the phase error ϕ is shown as a function of time measured in terms of the number of observed samples at the receiver. As it is represented in the plots, the carrier phase tracking loop starts operating after the frequency correction is performed on 1000 symbols.

As can be observed, the loops do not always lock at zero phase error ($\phi = 0$). The reason is that, in BPSK and QPSK, the lock points of the loops are where $\sin(2\phi_{\text{BPSK}}) = 0$ and $\sin(4\phi_{\text{QPSK}}) = 0$, respectively, which result in an ambiguity of π for BPSK and $\pi/2$ for QPSK. Figures 11-9(a) and 11-9(b) show constant lines to indicate the other potential phases at which the carrier loop could lock.

11.3.6 Symbol Synchronization

In the coarse estimation phase, either the noncoherent cross-spectrum symbol synchronizer (CSSS) or the noncoherent DTTL may be used to acquire the symbol timing. The choice of which to use depends on the SNR, as discussed in Chapter 10, with the CSSS being preferred at low SNR. For either synchronizer, the symbol period and pulse shape must be known.

(a)



(b)

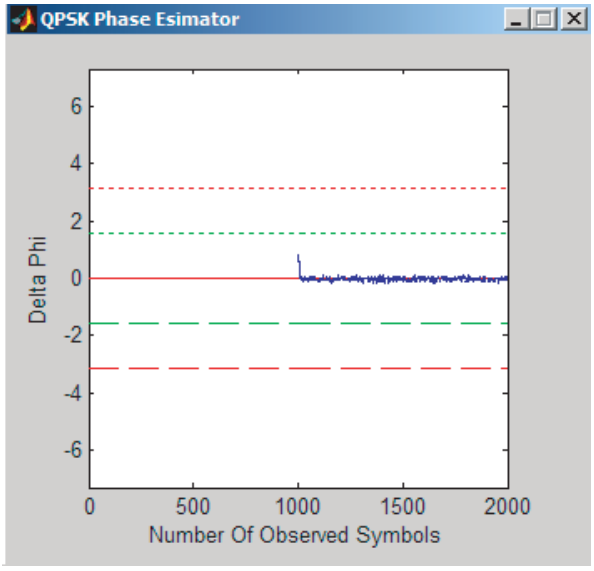


Fig. 11-9. Carrier synchronization loop performance: (a) BPSK and (b) QPSK.

In the fine estimation phase, a conventional coherent DTTL may be used. The remainder of this section discusses the method by which discrete-time matched-filter output samples are used by a DTTL. As discussed in the section on modulation classification, the complex observables corresponding to the matched-filter output at the time instants nL, where $n = 1, 2, \cdots, N$, are given by

$$\tilde{r}_n = \sum_{k=(n-1)L+1}^{nL} \tilde{r}_n\,[k], \quad n = 1, 2, \cdots, N \tag{11-29}$$

Thus,

$$\tilde{r}_n = \sum_{k=(n-1)L+1}^{nL} \left[\frac{1}{L} e^{j[\omega_r k T_s + \theta_n + \theta_c]} + \tilde{n}'_n\,[k]\right] \tag{11-30}$$

Therefore, the observation vector $\tilde{\mathbf{r}} = (\tilde{r}_1, \tilde{r}_2, \cdots, \tilde{r}_N)$ for a sequence of N symbols can be modeled as

$$\tilde{r}_n = e^{j[\omega_r k T_s + \theta_n + \theta_c]} + \tilde{n}'_n, \quad n = 1, 2, \cdots, N \tag{11-31}$$

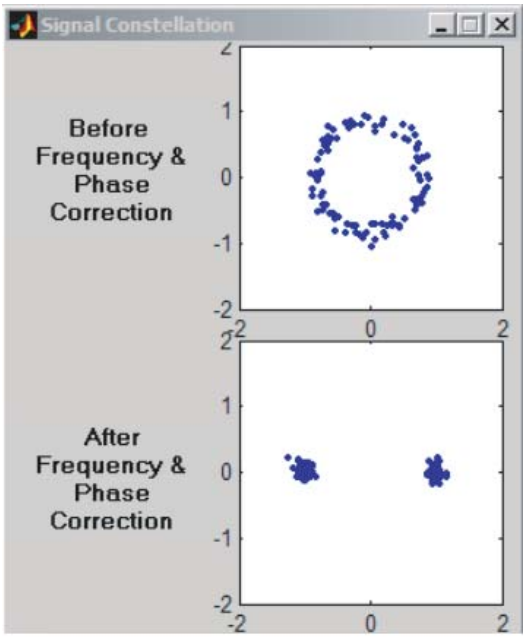
where $\tilde{n}'_n$ is a complex Gaussian random variable with mean zero and variance σ'^2 per dimension.

Figures 11-10(a) and 11-10(b) provide a visualization of the derotation of the signal constellation that takes place as a result of frequency and phase correction. The upper plot in each figure corresponds to the observation vector $\tilde{\mathbf{r}} = (\tilde{r}_1, \tilde{r}_2, \cdots, \tilde{r}_N)$, where $N = 100$ and a non-zero frequency offset (ω_r) was introduced to the system, and the lower plot represents the frequency- and phase-corrected version of $\tilde{\mathbf{r}}$.

Reference

[1] J. G. Proakis, *Digital Communications*, third ed., New York: McGraw Hill, Inc., 1995.

(a)



(b)

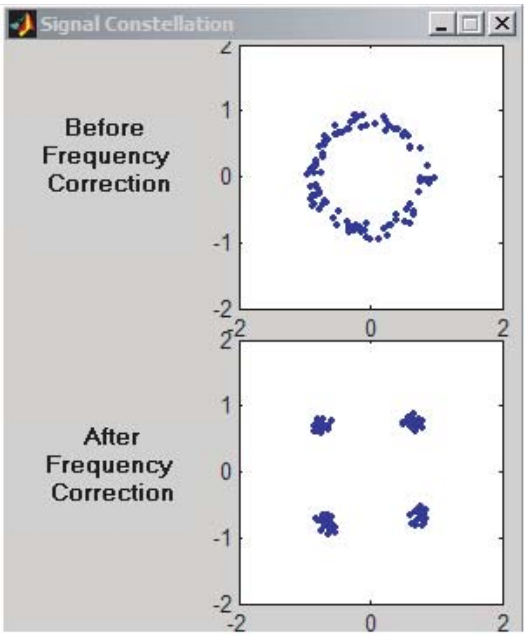


Fig. 11-10. Signal constellations for a signal with different modulations and the same parameters before and after frequency correction, with $E_s/N_0 =$ 20 dB**: (a) BPSK and (b) QPSK.**

Acronyms and Abbreviations

ADC	analog-to-digital converter
AGC	automatic gain control
AM/AM	amplitude-modulation-to-amplitude-modulation
AM/PM	amplitude-modulation-to-phase-modulation
ASIC	application-specific integrated circuit
AWGN	additive white Gaussian noise
BCH	Bose–Chaudhuri–Hocquenghem
BPSK	binary phase-shift keying
BT	bandwidth–time (product)
Caltech	California Institute of Technology
CCSDS	Consultative Committee for Space Data Systems
CIC	cascaded integrator comb
CLF	conditional-likelihood function
CMOS	complementary metal oxide semiconductor
CPFSK	continuous phase frequency-shift keyed
C-R	Cramer–Rao

CRB	C-R bounds
CRC	cyclic redundancy check
CSSS	cross-spectrum symbol synchronizer
CTL	carrier-tracking loop
CW	continuous wave
DAC	digital-to-analog converter
dB	decibel(s)
DESCANSO	Deep Space Communications and Navigation Systems Center of Excellence
DMF	derivative matched filter
DSN	Deep Space Network
DTTL	data-transition tracking loop
EDL	entry, descent, and landing
ESA	European Space Agency
FEC	forward error-correcting
FFT	fast Fourier transform
FMF	frequency-matched filter
FPGA	field programmable gate array
GLRT	generalized likelihood ratio test
GMSK	Gaussian-filtered minimum-shift keying
I	in-phase
I&D	integrate-and-dump
IEEE	Institute of Electrical and Electronics Engineers
IF	intermediate frequency

iid	independent, identically distributed
I-Q	in-phase–quadrature
JPL	Jet Propulsion Laboratory
kbps	kilobit(s) per second
ksps	kilosymbol(s) per second
LDTTL	linear data-transition tracking loop
LF	likelihood function
LLF	log-likelihood function
LPF	low-pass filter
LR	likelihood ratio
MAP	maximum a posteriori
Mbps	megabit(s) per second
MCAS	Micro Communications and Avionics Systems
MF	matched filter
MIT	Massachusetts Institute of Technology
ML	maximum likelihood
MMS	minimum mean-squared
***M*-PSK**	*M*-ary phase-shift keying
***M'*-PSK**	*M'*-ary phase-shift keying
MSK	minimum-shift keying
Msps	megasymbol(s) per second
NASA	National Aeronautics and Space Administration
NC-LDTTL	noncoherent linear data-transition tracking loop
NCO	numerically controlled oscillator

nqLLR	normalized quasi-log-likelihood ratio
NRZ	non-return to zero
OQPSK	offset quadrature phase-shift keying
pdf	probability density function
PLL	phase-locked loop
PN	pseudo-noise
PSD	power spectral density
PSK	phase-shift keying
Q	quadrature
QAM	quadrature amplitude modulation
qGLRT	quasi-generalized-likelihood ratio test
qLLR	quasi-log-likelihood ratio
QPSK	quadrature phase-shift keying
RC	resistor capacitor
Res	residual
RF	radio frequency
rms	root-mean-square
RS	Reed–Solomon
RV	random variable
S-band	around 2 GHz
SDR	software-defined radio
SNR	signal-to-noise ratio
SQPSK	staggered quadrature phase-shift keying
SSME	split-symbol moments estimator

ST-2	Space Technology-2
Sup	suppressed
SW	switch
sync	synchronization
T/R	transmitter/receiver
TT&C	telemetry, tracking, and control
UCLA	University of California, Los Angeles
UHF	ultra-high frequency
UQPSK	unbalanced quadrature phase-shift keying
U.S.	United States
USC	University of Southern California
USO	ultra-stable oscillator
VCO	voltage-controlled oscillator
X-band	around 8 GHz