Shaped Offset QPSK Capacity

By

Cenk Sahin

Submitted to the graduate degree program in Electrical Engineering and Computer Science and the Graduate Faculty of the University of Kansas in partial fulfillment of the requirements for the degree of Master of Science.

Erik Perrins, Chairperson

Committee members

Shannon Blunt

James Stiles

Date defended: _____

The Thesis Committee for Cenk Sahin certifies
that this is the approved version of the following thesis :

Shaped Offset QPSK Capacity

Erik Perrins, Chairperson

Date approved: _____

Abstract

In this work we compute the capacities and the pragmatic capacities of military-standard shaped-offset quadrature phase-shift keying (SOQPSK-MIL) and aeronautical telemetry SOQPSK (SOQPSK-TG). In the pragmatic approach, SOQPSK is treated as a modulation scheme as opposed to an encoder, and no iterations are performed between the SOQPSK demodulator and the outer binary decoder. We also evaluate the capacity of SOQPSK-TG constrained on a reduced complexity detection scheme based on the pulse amplitude modulation (PAM) representation of continuous phase modulation (CPM). The spectral efficiency of the PAM based SOQPSK-TG (SOQPSK-TG-PAM) is computed and shown to be superior to that of SOQPSK-MIL despite having the same detection complexity. We also show that the natural mapping of SOQPSK between input bits and SOQPSK waveforms maximizes the pragmatic capacity. Any other mapping such as differential encoding reduces the pragmatic capacity. We then focus on SOQPSK-TG due to its high spectral efficiency and present the performance results of a serially concatenated convolutional code (SCCC) SOQPSK-TG pragmatic scheme (SCCC-SOQPSK-TG), a low-density parity-check code (LDPC) SOQPSK-TG pragmatic scheme (LDPC-SOQPSK-TG), and a serially concatenated coded SOQPSK-TG scheme (SC-SOQPSK-TG). The LDPC scheme performs within 1.05 dB of the SOQPSK-TG capacity curve for various coding rates.

Contents

1 Background and Motivation **1**

2 SOQPSK Signal Model and Detection **4**

2.1 Introduction 4

2.2 Offset QPSK 5

2.3 CPM Signal Model 8

2.4 SOQPSK Trellis 13

2.5 SOQPSK Detection and BCJR Algorithm 15

3 Capacity and Spectral Efficiency of SOQPSK **21**

3.1 Introduction 21

3.2 Capacity of SOQPSK 22

3.3 Spectral Efficiency of SOQPSK 28

4 Pragmatic Capacity of SOQPSK **32**

4.1 Introduction 32

4.2 Pragmatic Coded Modulation 32

4.3 Pragmatic Coded SOQPSK 35

4.4 Pragmatic Capacity of SOQPSK 37

4.5 Optimal Mapping for Pragmatic Capacity of SOQPSK 41

5 Information Rates of SOQPSK Coding Schemes **43**

5.1 Introduction 43

5.2 SOQPSK-TG Coding Schemes 44

5.3 Simulation Results 46

6 Conclusions and Future Work **49**

A Proof of Equation (2.21) **51**

List of Figures

2.1 A QPSK signal constellation. 5

2.2 Rimoldi decomposition of the CPM modulator. 10

2.3 Transmitter model for SOQPSK. 11

2.4 Frequency pulse and phase response of SOQPSK-TG. 12

2.5 4-state time-varying trellis for the precoder. States are labeled with the state vector S_n and branches are labeled with the input-bit/output-symbol pair a_n/α_n. 13

2.6 (5,7) feedforward convolutional code (a) Encoder (b) One-step trellis. 5 and 7 are the integer values of the bit vectors of the digital filter tap coefficients [101] and [111]. 16

3.1 Channel capacities of SOQPSK-MIL, SOQPSK-TG, and SOQPSK-TG-PAM. 27

3.2 Power spectral densities of SOQPSK-MIL, SOQPSK-TG for a transmission rate of 1 symbol/s. 30

3.3 Spectral efficiencies of SOQPSK-MIL, SOQPSK-TG-PAM. 31

4.1 Block diagram of pragmatic coded modulation. 33

4.2 Block diagram of coded modulation. 34

4.3 Block diagram of pragmatic coded SOQPSK modulation. 35

4.4 Block diagram of serially concatenated coded SOQPSK modulation. 36

4.5 Capacity and pragmatic capacity of SOQPSK-MIL. 39

4.6 Capacity and pragmatic capacity of SOQPSK-TG-PAM. 40
4.7 Pragmatic capacities of SOQPSK-TG-PAM and differentially encoded SOQPSK-TG-PAM. 41

5.1 Block diagram of the serially concatenated coded SOQPSK-TG (SC-SOQPSK-TG)system. 44
5.2 Block diagram of the serially concatenated convolutional code (SCCC) pragmatic SOQPSK-TG (SCCC-SOQPSK-TG) system. 45
5.3 Block diagram of the low-density parity-check code (LDPC) pragmatic SOQPSK-TG (LDPC-SOQPSK-TG) system. 46
5.4 Information rates of SCCC-SOQPSK-TG, LDPC-SOQPSK-TG and SC-SOQPSK-TG for code rates of 1/2, 2/3, and 4/5 along with the spectral efficiency curve of SOQPSK-TG. 47

List of Tables

2.1 Relationship between the ternary branch symbol α_n, and the pseudo-symbols $\beta_{v,n}(\alpha_n)$ for SOQPSK. 20

5.1 Minimum required SNR values Eb/N_0 (dB) to achieve information rates, I_{LDPC}, I_{SC}, I_{SCCC}, equal to code rates, R, for SOQPSK-TG coding systems. Also given are the corresponding SNR values from the spectral efficiency curve of SOQPSK-TG. 48

Chapter 1

Background and Motivation

Shaped-offset quadrature phase-shift keying (SOQPSK) is a type of constant-envelope bandwidth-efficient continuous phase modulation (CPM) [1]. The constant signal envelope of SOQPSK makes it robust against the distortion introduced by non-linear power amplifiers. As a result, SOQPSK can be used in low-cost, high-power miniature transmitters since non-linear amplifiers are compact, inexpensive, and highly efficient in converting limited (i.e. battery) power into radiated power.

In this work, we focus on two versions of SOQPSK, namely military-standard SOQPSK (SOQPSK-MIL) [2] and aeronautical telemetry SOQPSK (SOQPSK-TG) [3], which differ in their frequency pulses. SOQPSK-TG achieves superior spectral containment compared to SOQPSK-MIL due to its partial response, which comes with a price of higher detection complexity. Optimal detection based on the CPM representation, as studied in [4], requires a 512-state trellis for SOQPSK-TG while it requires only a 4-state trellis for SOQPSK-MIL. However, reduced-complexity detection methods such as pulse amplitude modulation (PAM) [5,6] and pulse truncation (PT) [7,8] exist and result in 4-state detectors for SOQPSK-TG that perform within 0.1 dB of the optimal detector.

Recently, the ability has emerged to determine the theoretical performance limits of modulations with memory. In [9], a simulation-based technique to compute the capacity of

channels with memory (i.e. CPM) is proposed. In [10], a similar method is used to compute the capacity of CPM. Most recently, in [11] the best spectrally-efficient CPM modulations are investigated. However, despite the widespread use of SOQPSK, its channel capacity or the bound on its spectral efficiency has not been determined yet. The evaluation of the capacity of SOQPSK establishes a yardstick by which practical systems can be judged.

In [12], the authors define *bit-interleaved* (a.k.a. pragmatic) coded modulation as a system that consists of a binary encoder, a bit-interleaver, and an M-ary modulator. In [11], pragmatic CPM is defined as a pragmatic coded modulation where CPM is the modulation method. Furthermore, the pragmatic capacity of CPM is computed and maximized by altering the natural mapping between the CPM symbols and the trellis branches. In this work, we extend pragmatic CPM to SOQPSK modulation.

We first compute the capacities of SOQPSK-MIL, SOQPSK-TG, and PAM based SOQPSK-TG (SOQPSK-TG-PAM) by using the method proposed in [10]. The capacity of SOQPSK-TG-PAM is very close to the capacity of SOQPSK-TG despite having much lower detection complexity while the capacity of SOQPSK-MIL is slightly higher than the capacity of SOQPSK-TG. We compute the spectral efficiencies of SOQPSK-MIL and SOQPSK-TG-PAM, and show that SOQPSK-TG-PAM is more spectrally-efficient than SOQPSK-MIL while having the same detection complexity. The pragmatic capacity is defined and computed for SOQPSK-MIL and SOQPSK-TG-PAM. We show that the pragmatic capacity is maximized by the natural mapping of SOQPSK, and that any other mapping of the input bits onto the trellis branches results in lower pragmatic capacity. Simulation results of a low-density parity-check code (LDPC) SOQPSK-TG pragmatic scheme (LDPC-SOQPSK-TG), a serially concatenated convolutional code (SCCC) SOQPSK-TG pragmatic scheme (SCCC-SOQPSK-TG), and a serially concatenated coded SOQPSK-TG scheme where SOQPSK-TG is a constituent code (SC-SOQPSK-TG) are presented for coding rates of 1/2, 2/3 and 4/5. The LDPC scheme provides superior performance while performing within 1.05 dB of the capacity curve of SOQPSK-TG.

In the first part of this thesis, Chapter 2, we lay the groundwork necessary for the discussion of the contributions that led to this work. In Chapter 2 we describe SOQPSK modulation and present a CPM signal model for SOQPSK which includes the specifications of SOQPSK-MIL and SOQPSK-TG. We also briefly discuss the detection of SOQPSK, present the PAM representation of SOQPSK-TG, and give a summary of the BCJR [13] algorithm.

In the second part of this thesis, Chapter 3, Chapter 4 and Chapter 5, we present our results. In Chapter 3, the capacities and the spectral efficiencies of SOQPSK-MIL, SOQPSK-TG and SOQPSK-TG-PAM are computed. In Chapter 4, the pragmatic capacities of SOQPSK-MIL and SOQPSK-TG-PAM are computed and the optimality of the natural mapping of SOQPSK in the pragmatic sense is discussed. In Chapter 5, we compute the information rates of three SOQPSK-TG coding schemes: SCCC-SOQPSK-TG, LDPC-SOQPSK-TG, and SC-SOQPSK-TG.

Chapter 2

SOQPSK Signal Model and Detection

2.1 Introduction

Shaped offset quadrature phase-shift keying (SOQPSK)—the topic of this work—is a type of modulation that is robust against distortion introduced by non-linear components in the transmitter by virtue of its constant envelope. In this chapter, we provide a complete signal model for SOQPSK and discuss SOQPSK detection. We first discuss the relationship of SOQPSK to quadrature phase-shift keying (QPSK). We show that SOQPSK signals—unlike QPSK signals—have continuous phase; therefore, SOQPSK is a type of *continuous phase modulation* (CPM) [1]. We give a complete description of the CPM representation of SOQPSK which includes the specifications of the two standardized versions of SOQPSK, namely military-standard SOQPSK (SOQPSK-MIL) and aeronautical telemetry SOQPSK (SOQPSK-TG) [3]. We conclude the chapter with an overview of SOQPSK detection which includes a summary of the BCJR algorithm [13], its application to SOQPSK, and a summary of the pulse amplitude modulation (PAM) [5] based reduced complexity detection of SOQPSK-TG.

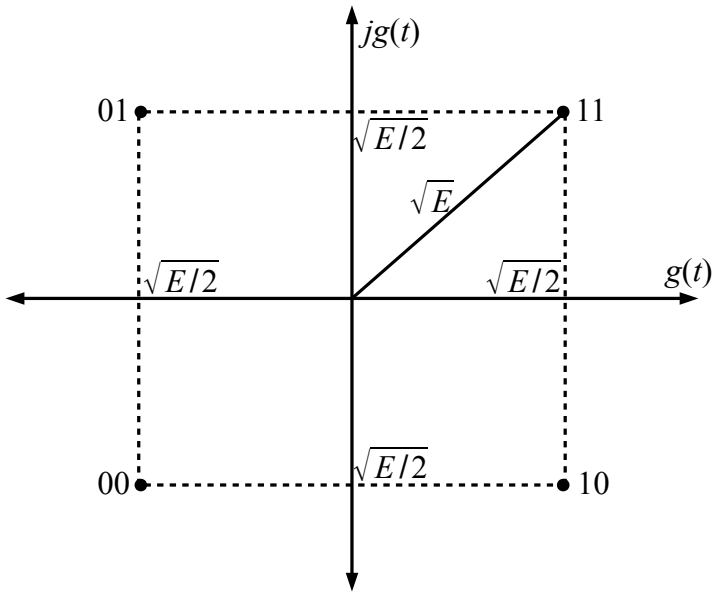

Figure 2.1: A QPSK signal constellation.

2.2 Offset QPSK

In *phase-shift keying* (PSK), information is conveyed by assigning distinct phase values to the carrier f_c. With M-ary PSK every m-bit sequence is mapped onto one of M complex baseband waveforms represented as

$$s_k(t) = \sqrt{E}g(t)\exp\left(\frac{j2\pi(k-1)}{M}\right), \quad k = 1, 2, \ldots, M-1 \tag{2.1}$$

where $M = 2^m$, $g(t)$ is the unit-energy signal pulse with a duration of T seconds and E is the *M-ary symbol energy* [14]. The baseband waveform $s_k(t)$ can also be expressed as the sum of an *in-phase* (I) and a *quadrature* (Q) component as in

$$s_k(t) = \underbrace{\sqrt{E}g(t)\cos\left(\frac{2\pi(k-1)}{M}\right)}_{I} + j\underbrace{\sqrt{E}g(t)\sin\left(\frac{2\pi(k-1)}{M}\right)}_{Q}, \quad k = 1, 2, \ldots, M-1. \tag{2.2}$$

In the case of $M = 4$ PSK modulation is called *quadrature* PSK (QPSK). Consider a QPSK modulation with the *constellation* and *binary* labeling shown in Figure 2.1. The

horizontal axis is $g(t)$ while the vertical axis is $jg(t)$. Note that an additional phase shift of $\pi/4$ is introduced to all waveforms so that the phase values of the constellation points are from the set $\{\pi/4, 3\pi/4, 5\pi/4, 7\pi/4\}$. We note that the first bit in each binary label determines the I component of the baseband signal while the second bit determines the Q component of the signal. Consider an infinitely long binary *antipodal* input sequence $\boldsymbol{a} = (\ldots, a_{-n}, \ldots, a_{-1}, a_0, a_1, \ldots, a_n, \ldots)$ where a binary '1' is represented by $+1$ and a binary '0' is represented by -1, i.e. $a_n \in \{-1, +1\}$. The QPSK modulator output for the bit sequence $\boldsymbol{a}$ can be written as a pulse train of the form

$$s(t; \boldsymbol{a}) = \sqrt{\frac{E}{2}} \sum_i g(t - iT)\big(a_{2i} + j a_{2i+1}\big) \tag{2.3}$$

where $i \in \mathbb{Z}$ is the discrete-time index. We note that any change to the phase of $s(t; \boldsymbol{a})$ occurs at integer multiples of T. At the instances when the values of both I and Q components, a_{2i} and a_{2i+1}, change simultaneously the phase of $s(t; \boldsymbol{a})$ changes by π radians. This abrupt change in the phase of the modulated signal causes the envelope of $s(t; \boldsymbol{a})$ to pass through the origin of the signal space shown in Figure 2.1. This is undesirable because it introduces *spectral regrowth* when a nonlinear amplifier is used. A version of QPSK known as *offset* QPSK (OQPSK) addresses the problem. With OQPSK the I and Q components of the standard QPSK are offset by $T/2$. The resulting OQPSK signal has the form

$$s(t; \boldsymbol{a}) = \sqrt{\frac{E}{2}} \sum_i g(t - iT) a_{2i} + j g(t - iT - T/2) a_{2i+1}. \tag{2.4}$$

The I and Q components of OQPSK never change simultaneously; consequently, the phase of $s(t; \boldsymbol{a})$ never changes by π radians and the envelope of the signal doesn't pass through zero. On the other hand, the phase of $s(t; \boldsymbol{a})$ changes more frequently than the standard QPSK. Every $T/2$ seconds the phase either advances or retards by $\pi/2$ radians, or remains the same. It is important to note that the phase of $s(t; \boldsymbol{a})$ is only allowed to change in one direction at multiples of $T/2$. For example if the phase of $s(t; \boldsymbol{a})$ is $\pi/4$ (constellation point

labeled with 11 in Figure 2.1) when an even indexed bit a_{2i} is to be modulated, the phase either remains the same with a_{2i} ='+1' or advances by $\pi/2$ with a_{2i} ='-1', but cannot retard by $\pi/2$. This is because even indexed input values $\{a_{2i}\}$ can only cause phase transitions across the vertical axis in Figure 2.1 while odd indexed input values $\{a_{2i+1}\}$ can only cause phase transitions across the horizontal axis. In other words, phase transitions are constrained by the current phase value of $s(t;\boldsymbol{a})$ and whether an even or an odd indexed bit is being modulated. If we are to focus mainly on the phase of the OQPSK signal given in (2.4) an alternative representation will be more useful [15]. If we assume a rectangular signal pulse $g(t)$ with fixed amplitude $\sqrt{1/T}$ we can write (2.4) as

$$s(t;\boldsymbol{a}) = \sqrt{\frac{E}{T}} \exp\left(\frac{\pi}{4} + \frac{\pi}{2}\int_{-\infty}^{t} \sum_{i} \alpha_i \delta(\tau - iT/2) d\tau\right). \tag{2.5}$$

where $\alpha_i \in \{-1, 0, +1\}$ and $\delta(\cdot)$ is the *Dirac delta function* [16] defined by

$$\delta(t) \triangleq 0, \qquad t \neq 0$$

and

$$\int_{-\infty}^{\infty} \delta(t) dt \triangleq 1. \tag{2.6}$$

The relationship between $\boldsymbol{a}$ and the *ternary* symbols $\{\alpha_i\}$ is non-trivial and will be discussed in the next section. The new representation of OQPSK in (2.5) encapsulates the possible phase shifts that occur every $T/2$ seconds through the symbol α_i and the Dirac delta function. It is important to note that although OQPSK prevents phase shift of π radians, it still results in a signal with high spectral lobes due to the abrupt changes in the phase of $s(t;\boldsymbol{a})$. In fact QPSK and OQPSK have the same power spectral density [14]. The abrupt jumps in the phase of the OQPSK signal $s(t;\boldsymbol{a})$ are eliminated by replacing the Dirac delta function in (2.5) by a continuous frequency shaping function with unit area whose duration is an integer multiple of $T/2$. This new type of modulation is called *shaped* OQPSK (SOQPSK)

modulation which is the subject of this work. Unlike OQPSK, the phase of an SOQPSK signal transitions among constellation points continuously as in continuous phase modulation (CPM) [1]. In other words, SOQPSK modulation is a type of CPM. In this section we only intended to introduce SOQPSK modulation and show that it is a type of CPM. In the next section we discuss in great detail how to represent an SOQPSK signal as a CPM signal.

2.3 CPM Signal Model

The complex baseband SOQPSK signal can be represented as a CPM signal of the form

$$s(t;\boldsymbol{\alpha}) \triangleq \sqrt{\frac{E_s}{T_s}} \exp\{j\psi(t;\boldsymbol{\alpha})\} \tag{2.7}$$

where $\boldsymbol{\alpha} \triangleq \{\alpha_i\}$ is the transmitted symbol sequence drawn from a *ternary* alphabet, i.e. $\alpha_i \in \{-1, 0, +1\}$. T_s is the *symbol interval* and E_s is the *energy per symbol*. The phase of the signal, $\psi(t;\boldsymbol{\alpha})$, is of the form

$$\psi(t;\boldsymbol{\alpha}) \triangleq 2\pi h \sum_i \alpha_i q(t - iT_s) \tag{2.8}$$

where $h = 1/2$ is the *modulation index*. The *phase response* $q(t)$ is defined as the time integral of the *frequency pulse* $f(t)$ that has a length of L symbol times, i.e. LT_s, and is given by the expression

$$q(t) \triangleq \begin{cases} 0 & t < 0 \\ \displaystyle\int_0^t f(\tau)\, d\tau & 0 \le t < LT_s \\ 1/2 & t \ge LT_s. \end{cases} \tag{2.9}$$

Based on the definitions made so far along with (2.9) and (2.8) it is easy to show that the SOQPSK signal description given in (2.7) is equivalent to the OQPSK signal given in (2.5) with a phase offset of $\pi/4$ if $\delta(\cdot)$ is replaced by $f(\cdot)$. The terms E and T in (2.4) are replaced

by $2E_s$ and $2T_s$ in (2.7) respectively. The integral is taken inside the sum in (2.7). Finally $\pi/2$ term in (2.4) is replaced by $2\pi h = \pi$ in (2.7) where the factor of 2 is due to the fact that the area under $\delta(\cdot)$ equals 1 while the area under $f(\cdot)$ equals 1/2.

If the duration of $f(t)$ is equal to the symbol interval T_s (i.e. $L = 1$) the signal is said to be *full-response* and if the duration is greater than T_s (i.e. $L > 1$) the signal is *partial-response*. Given that the modulation index is rational $h = K/l$, the phase in (2.8) may be expressed as

$$\psi(t;\boldsymbol{\alpha}) = \underbrace{2\pi h \sum_{i=n-L+1}^{n} \alpha_i q(t - iT_s)}_{\theta(t)} + \underbrace{\pi h \sum_{i=0}^{n-L} \alpha_i}_{\theta_{n-L}} \tag{2.10}$$

where $nT_s \leq t < (n+1)T_s$. We see in (2.10) that the phase of the transmitted signal is a function of not only the current symbol α_n but also all the past symbols; therefore, SOQPSK is a modulation scheme with *memory*. The *phase state* θ_{n-L} is the cumulative phase of all past phase responses that have reached the value 1/2 while the first expression in (2.10) is the phase increment due to the frequency pulses that are still in progress. The *phase state* is drawn from an alphabet of 4 values, $\theta_{n-L} \in \{0, \pi/2, \pi, 3\pi/2\}$, when taken modulo 2π.

According to the well-known Rimoldi decomposition [17] a CPM modulator can be represented as a *continuous phase encoder* (CPE) followed by a *memoryless modulator* (MM) as shown in Figure 2.2 where $\sum_p$ denotes a modulo p adder. The CPE is a time-invariant encoder and can be modeled as a *finite-state machine* (FSM). The state of the CPE denoted by S_n is uniquely determined by the *state vector* $\boldsymbol{s}_n = [P_{n-L}, \alpha_{n-L+1}, \alpha_{n-L+2}, \ldots, \alpha_{n-1}]$ where P_{n-L} is the *phase state index* from an alphabet of p values, $P_{n-L} \in \{0, 1, 2, \ldots, p-1\}$. The value of the state S_n captures the entire memory in the modulator due to all past symbols. From Figure 2.2 it can be seen that the phase state index is given by

$$P_{n-L} = \left(\sum_{i=0}^{n-L} \alpha_i \right) \bmod p \tag{2.11}$$

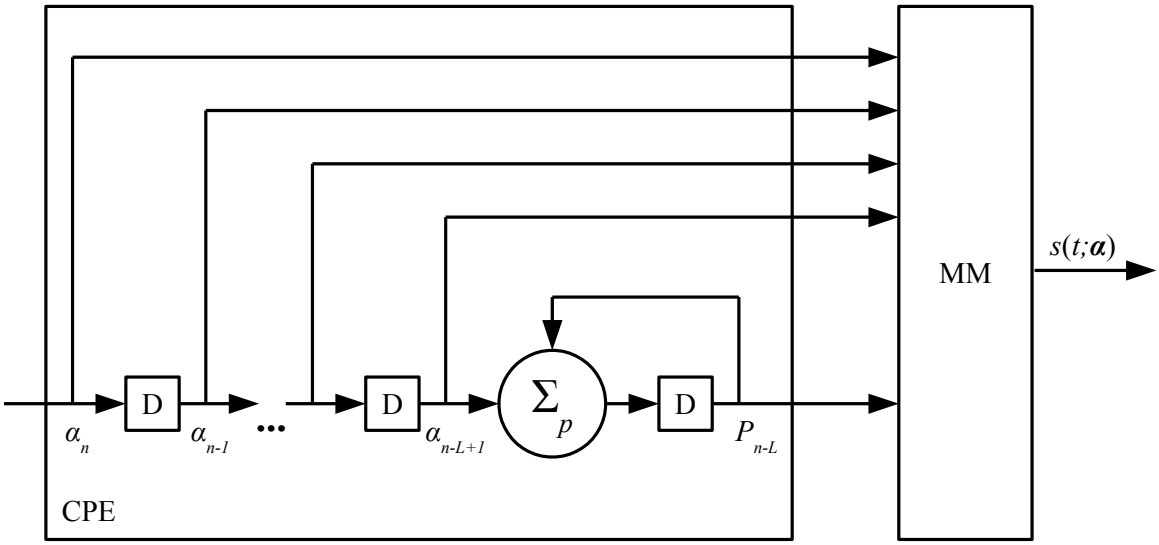

Figure 2.2: Rimoldi decomposition of the CPM modulator.

and related to the phase state by the expression

$$\theta_{n-L} = \frac{2\pi P_{n-L}}{p}. \tag{2.12}$$

The value of p is determined by the value of the rational modulation index h and when $K = 1$ it is given by $p = 2/h$. In the case of SOQPSK we have $p = 4$ from $h = 1/2$ which can also be seen from the phase state alphabet $\theta_{n-L} \in \{0, \pi/2, \pi, 3\pi/2\}$.

The state vector $\boldsymbol{s}_n$ can take pM^{L-1} distinct values given that $\{\alpha_i\}$ are M-ary symbols. The CPE passes the state vector $\boldsymbol{s}_n$ along with the current input symbol α_n to the MM, which uses them to determine which one of the pM^L waveforms of duration T_s to transmit. The state vectors $\{\boldsymbol{s}_n\}$ outputted by the CPE are such that the output of the MM has a continuous phase.

With SOQPSK—unlike CPM—the symbol sequence transmitted over the channel, $\boldsymbol{\alpha}$, is not the underlying bit sequence $\boldsymbol{a} \triangleq \{a_i\}$, $a_i \in \{0, 1\}$. The ternary symbol sequence $\boldsymbol{\alpha}$ is

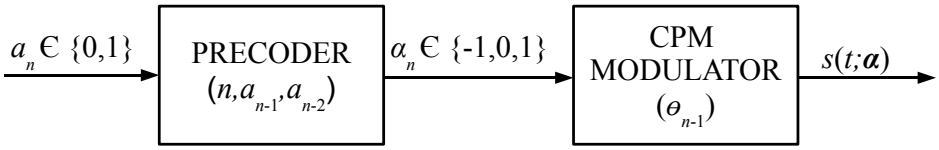

Figure 2.3: Transmitter model for SOQPSK.

derived from the original bit sequence $\boldsymbol{a}$ by the precoding operation defined by the rule [18]

$$\alpha_n(\boldsymbol{a}) = (-1)^{n+1}(2a_{n-1} - 1)(a_n - a_{n-2}). \tag{2.13}$$

Consequently the SOQPSK transmitter consists of a precoder followed by a ternary CPM modulator as shown in Figure 2.3 [4]. Due to the precoding operation, in any given symbol interval, α_i is drawn from one of the two *binary* alphabets: $\{-1, 0\}$ or $\{0, +1\}$ [18]. Thus, SOQPSK is viewed as a *constrained* ternary CPM. The role of the precoder is to orient the phase of the CPM signal so that it behaves like the phase of an offset-QPSK (OQPSK) signal driven by the symbol sequence $\boldsymbol{a}$.

There are two standardized versions of SOQPSK, namely military-standard SOQPSK (SOQPSK-MIL) and aeronautical telemetry SOQPSK (SOQPSK-TG) [3]. SOQPSK-MIL uses a full-response ($L = 1$) rectangular frequency pulse given by

$$f_{\text{MIL}}(t) \triangleq \begin{cases} \frac{1}{2T_s}, & 0 \le t < T_s \\ 0, & \text{otherwise.} \end{cases} \tag{2.14}$$

while SOQPSK-TG uses a longer ($L = 8$) and smoother frequency pulse given by

$$f_{\text{TG}}(t) \triangleq A\frac{\cos\left(\frac{\pi\rho Bt}{2T_s}\right)}{1 - 4\left(\frac{\rho Bt}{2T_s}\right)^2} \times \frac{\sin\left(\frac{\pi Bt}{2T_s}\right)}{\frac{\pi Bt}{2T_s}} \times w(t) \tag{2.15}$$

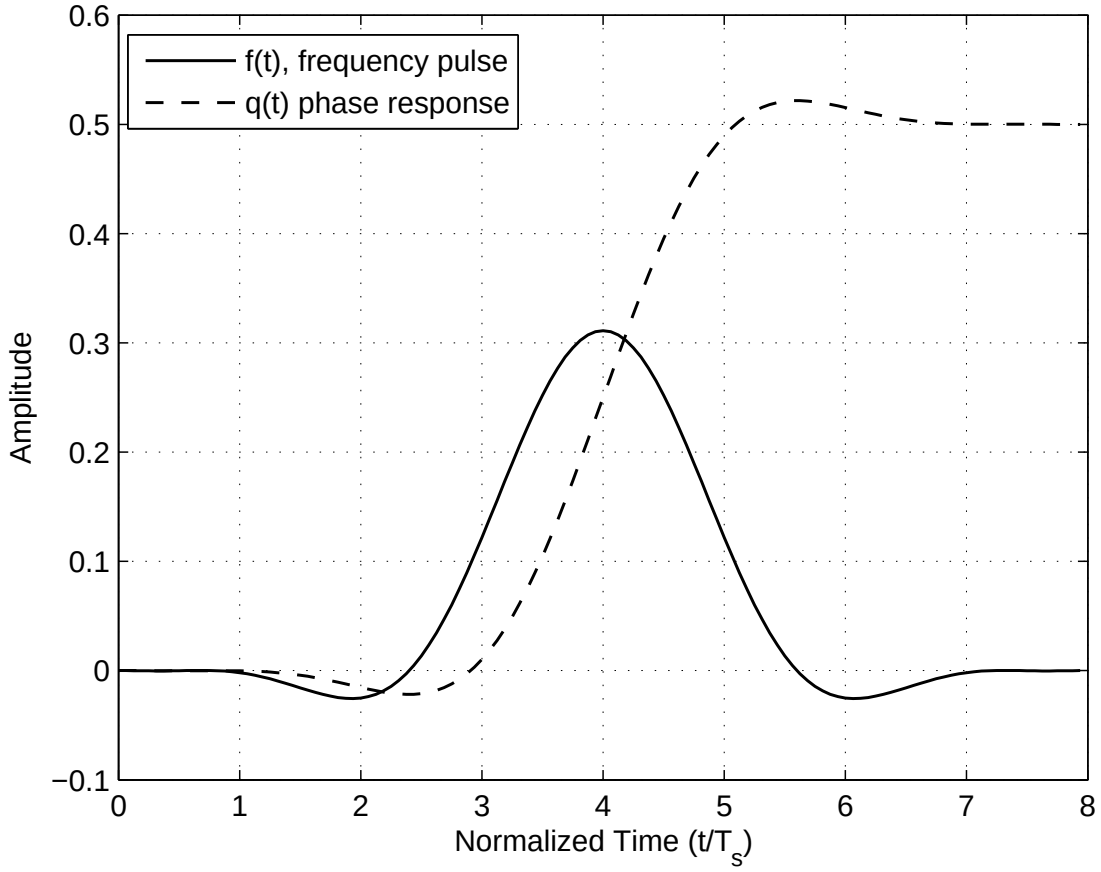

Figure 2.4: Frequency pulse and phase response of SOQPSK-TG.

where the window is

$$w(t) \triangleq \begin{cases} 1, & 0 \leq \left|\frac{t}{2T_s}\right| < T_1 \\ \frac{1}{2}+\frac{1}{2}\cos\left(\frac{\pi}{T_2}\left(\frac{t}{2T_s}-T_1\right)\right), & T_1 \leq \left|\frac{t}{2T_s}\right| \leq T_1+T_2 \\ 0, & T_1+T_2 < \left|\frac{t}{2T_s}\right|. \end{cases}$$

The constant A is chosen such that the area of the pulse is equal to 1/2 and $T_1 = 1.5$, $T_2 = 0.5$, $\rho = 0.7$ and $B = 1.25$. The frequency pulse $f(t)$ and the phase response $q(t)$ of SOQPSK-TG are shown in Figure 2.4. The longer and smoother frequency pulse of SOQPSK-TG compared to SOQPSK-MIL results in superior spectral containment.

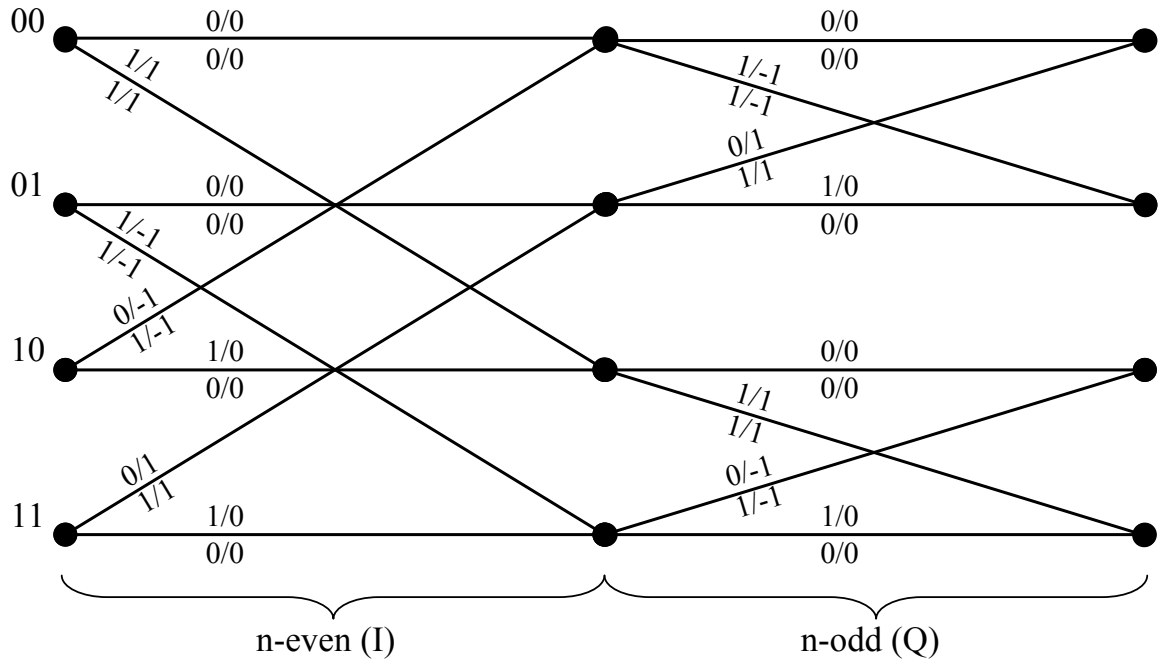

Figure 2.5: 4-state time-varying trellis for the precoder. States are labeled with the state vector S_n and branches are labeled with the input-bit/output-symbol pair a_n/α_n.

2.4 SOQPSK Trellis

In (2.13), we see that α_n is a function of three binary-valued state variables a_{n-1}, a_{n-2} and n-even/n-odd which leads to an 8-state trellis to describe the precoder in Figure 2.3 [19]. We follow the approach taken in [20] that handles the state variable n-even/n-odd by a time-varying two-section trellis shown in Figure 2.5. The state of the trellis denoted by S_n is determined by the variables a_{n-1}, a_{n-2} that results in $S_n \in \{00, 01, 10, 11\}$. The state variables a_{n-1} and a_{n-2} are ordered in such a way that the *in-phase* (I) bit (the input bit for the even symbol time) is always the most significant and the *quadrature* (Q) bit (the input bit for the odd symbol time) is always the least significant. So $S_n = a_{n-2}a_{n-1}$ for n-even and $S_n = a_{n-1}a_{n-2}$ for n-odd. The labels above each branch are the input-bit/output-symbol pairs a_n/α_n for the system in Figure 2.3. Also seen on the trellis are labels a_n/α_n below each branch that represent input-bit/output-symbol pairs of *differentially encoded* SOQPSK (DE-SOQPSK) [21]. With DE-SOQPQK the information symbols a_n are double-differentially

encoded prior to precoding by the rule

$$u_n = a_n \oplus u_{n-2}, \qquad a_n, u_n \in \{0, 1\} \tag{2.16}$$

where $\oplus$ is the modulo-2 operator with identity 0. The precoder in Figure 2.3 is preceded by the double differential encoder that takes the bit sequence $\boldsymbol{a}$ as input and outputs the bit sequence $\boldsymbol{u} = \{u_i\}$ which is fed to the precoder. The differential encoding rule in (2.16) can be summarized as "change phase on 1" because an input of $a_n = 1$ causes the output value u_n to change relative to the state value of u_{n-2}. Note the double-differential encoding is equivalent to differentially encoding in-phase and quadrature bits separately. The purpose of differential encoding is to remove the need to correct the phase offset in the receiver due to the channel impairements by transmitting information with phase shifts rather than absolute phases [22].

The advantage of the precoder trellis is that there is a *one-to-one mapping* between the phase states of the CPM modulator $\theta_{n-L} \in \{0, \pi/2, \pi, 3\pi/2\}$ and the precoder trellis states $S_n \in \{00, 01, 10, 11\}$ [4] given by

$$\begin{aligned} 00 &\leftrightarrow \frac{3\pi}{2}, \qquad 01 \leftrightarrow \pi, \\ 10 &\leftrightarrow 0, \qquad\quad 11 \leftrightarrow \frac{\pi}{2}. \end{aligned} \tag{2.17}$$

The phase states are simply a $\pi/4$-rotated version of the QPSK constellation given in Figure 2.1. Since SOQPSK-MIL is full response ($L = 1$) the state vector of the CPM modulator is uniquely determined by the phase state index, i.e. $\boldsymbol{s}_n = [P_{n-1}]$. It follows that the 4-state trellis in Figure 2.5 can be used to the describe the entire system in Figure 2.3. The optimal trellis-based detection of SOQPSK-MIL can be carried out on the precoder trellis without the need for a separate CPM trellis. In the case of SOQPSK-TG ($L = 8$) the state vector $\boldsymbol{s}_n$ takes $512 = 4 \times 2^7$ distinct values since α_i takes values from two binary alphabets: $\{-1, 0\}$ or $\{0, +1\}$. As we will discuss in the next section the pulse amplitude modulation (PAM) based

approximation of CPM [5,6] allows a 4-state trellis to be used for *near-optimal* detection of SOQPSK-TG [20] in which case the precoder trellis in Figure 2.5 again can be used instead of a separate CPM trellis.

2.5 SOQPSK Detection and BCJR Algorithm

Because SOQPSK is a modulation scheme with memory, the optimal detection of SOQPSK is carried out on the trellis that models the modulation. Trellis based SOQPSK detectors were first studied in [19]. Two trellis-based detection methods generally used are maximum likelihood sequence detection (MLSD) (i.e. Viterbi Algorithm) and maximum a posteriori (MAP) detection. The MAP detection of SOQPSK is accomplished with the BCJR [13] algorithm that can be used to compute the a posteriori (AP) probabilities of input bits. The BCJR [13] algorithm can also be used to compute the capacity and the pragmatic capacity of SOQPSK as we show in Chapter 3 and Chapter 4. Considering its importance we give a brief system-level desciption of this algorithm and show how it is applied to SOQPSK.

The BCJR [13] algorithm is applied to trellis encoders to compute the AP state transition probabilities. These probabilities then can be used to compute the AP probabilities of input bits or coded bits. To simplify our exposition we present the algorithm in the context of the rate 1/2 binary convolutional code [23] whose encoder and trellis are shown in Figure 2.6. The input to the encoder is a binary sequence denoted by $\boldsymbol{a} = (a_0, a_1, \ldots, a_n, \ldots)$ where $a_i \in \{0, 1\}$ and $i \in \mathbb{Z}^+$ is the discrete-time index. The input stream $\boldsymbol{a}$ passes through two digital filters and produces two binary output streams given by

$$c_n^{(1)} = a_n + a_{n-2} \quad \text{and} \quad c_n^{(2)} = a_n + a_{n-1} + a_{n-2} \tag{2.18}$$

which then are interleaved to produce coded stream $\boldsymbol{c} = (c_0^{(1)}, c_0^{(2)}, c_1^{(1)}, c_1^{(2)}, \ldots, c_n^{(1)}, c_n^{(2)}, \ldots)$ where $c_i^{(1)}, c_i^{(2)} \in \{0, 1\}$. For every input bit there are two output bits, so it is said that the rate of the code is 1/2. The encoder is a finite state-machine with memory elements

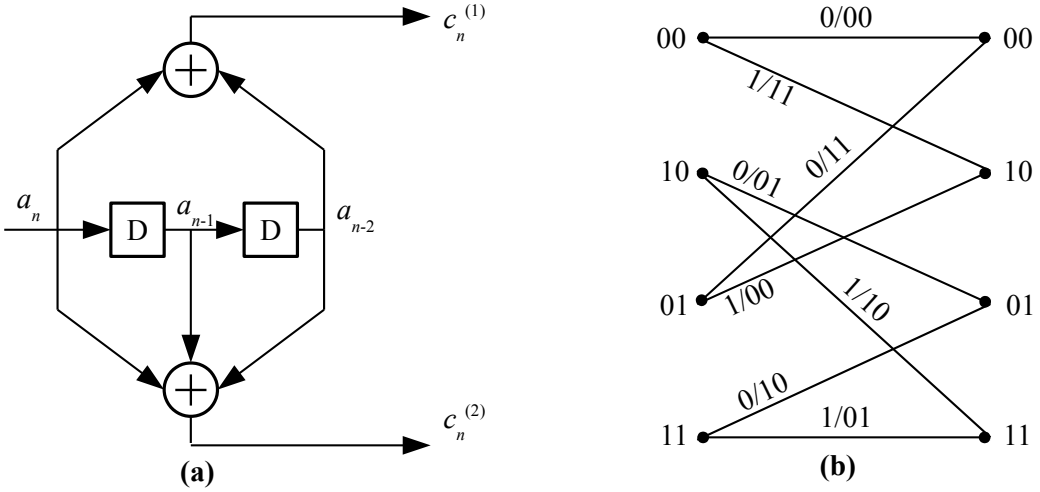

Figure 2.6: (5,7) feedforward convolutional code (a) Encoder (b) One-step trellis. 5 and 7 are the integer values of the bit vectors of the digital filter tap coefficients [101] and [111].

a_{n-1} and a_{n-2}. When the state diagram of the encoder is expanded over all time steps we get a trellis. Because the encoder is time-invariant the single section of the trellis shown in Figure 2.6 completely characterizes the code. The state of the trellis in the beginning of the n-th bit interval S_n is given by the ordered bit pair $a_{n-1}a_{n-2}$. The state of the encoder transitions to the next state $S_{n+1} = a_n a_{n-1}$ with the current input bit a_n and the encoder outputs the coded bits pairs (i.e. codewords) $c_n = c_n^{(1)} c_n^{(2)}$. The state dependent input-output relationship of the encoder is specified with the trellis branch labels $a_n/c_n^{(1)}c_n^{(2)}$ in Figure 2.6. The coded bit stream $\boldsymbol{c}$ is then modulated and transmitted over the channel.

In the rest of the description of the algorithm we adopt the notation used in a version of the algorithm called *soft-input soft-ouput* (SISO [24]) algorithm. When applied to this convolutional code the BCJR [13] algorithm takes two sequences of a priori probability distributions: one for the input bits denoted by $\mathbf{P}(\boldsymbol{a}; I) \triangleq \{P_i(a_i = a; I)\}$ and one for the codewords denoted by $\mathbf{P}(\boldsymbol{c}; I) \triangleq \{P_i(c_k = c; I)\}$ where $P_i(c_i = c; I) = P(c_i^{(1)} = c^{(1)}; I) \cdot P_i(c_i^{(2)} = c^{(2)}; I)$, and I implies *input*. The symbols a, c, $c^{(1)}$ and $c^{(2)}$ denote possible realizations of the input bits a_n, codewords $c_n = c_n^{(1)} c_n^{(2)}$, coded bits $c_n^{(1)}$ and $c_n^{(2)}$ respectively, i.e. $a, c^{(1)}, c^{(2)} \in \{0, 1\}$ and $c \in \{00, 01, 10, 11\}$. The algorithm outputs the sequence of AP probability distribu-

tions of state transitions denoted by $\mathbf{P}((\boldsymbol{s},\boldsymbol{s'});O) \triangleq \{P_i((S_i,S_{i+1})=(s,s');O\}$ where O implies *output*. The symbols s and s' denote possible realizations of the current state S_n and next state S_{n+1} while (s,s') denotes possible state transitions (S_n,S_{n+1}) on the trellis, i.e. $(s,s') \in \{(00,00),(00,10),(10,01),(10,11),(01,00),(01,10),(11,01),(11,11)\}$. It is trivial to compute AP probability distributions of the input bits $P_n(a_n=a;O)$ or coded bits $P_n(c_n^{(1)}=c^{(1)};O)$ and $P_n(c_n^{(2)}=c^{(2)};O)$ by simply summing up the AP state transition probabilites $P_n((S_n,S_{n+1})=(s,s');O)$ based on the bit labels of the state pairs (S_n,S_{n+1}). For example the AP probability of a_n being 1 is given by

$$P_n(a_n=1;O)=\sum_{(s,s')\in U_1} P_n((S_n,S_{n+1})=(s,s');O). \tag{2.19}$$

where $U_1=\{(00,10),(10,11),(01,10),(11,11)\}$ denotes the set of all state transitions (s,s') labeled with $a_n=1$ in Figure 2.6.

Before we continue with the application of the BCJR [13] algorithm to SOQPSK it is important to note the physical meanings of the input distribution $P_n(c_n=c;I)$ and the output distribution $P_n((S_n,S_{n+1})=(s,s');O)$. Assume that the coded bit stream $\boldsymbol{c}$ is modulated with BPSK and transmitted over an additive white Gaussian (AWGN) channel. On the receiver side we have a vector of noisy samples $\boldsymbol{r}=\{r_0^{(1)},r_0^{(2)},r_1^{(1)},r_1^{(2)},\ldots,r_n^{(1)},r_n^{(2)},\ldots\}$. The probability distribution $P_n(c_n=c;I)$ is simply the *likelihood* values of the codeword c_n, i.e. $P_n(c_n=c;I)=P(r_n^{(1)}|c_n^{(1)}=c^{(1)})\cdot P(r_n^{(2)}|c_n^{(1)}=c^{(1)})$ computed by the demodulator for all possible values of c. The AP probability distribution $P_n((S_n,S_{n+1})=(s,s');O)$ on the other hand is computed upon operating on the entire received vector $\boldsymbol{r}$ via the input $\mathbf{P}(\boldsymbol{c};I)$, with the knowledge of the code contraints and the a priori input bit distributions $\mathbf{P}(\boldsymbol{a};I)$, i.e. $P_n((S_n,S_{n+1})=(s,s');O)=P((S_n,S_{n+1})=(s,s')|\boldsymbol{r})$.

We now briefly discuss how to apply the BCJR [13] algorithm to SOQPSK. A baseband

SOQPSK signal received through an AWGN channel is modeled as

$$b(t;\boldsymbol{a}) \triangleq s(t;\boldsymbol{a}) + n(t) \tag{2.20}$$

where $n(t)$ is a complex white Gaussian noise process with two-sided power spectral density $N_0/2$ and the transmitted signal $s(t;\boldsymbol{a})$ is given by the equation (2.7). When we apply the algorithm to SOQPSK we view SOQPSK as a code with binary input sequence $\boldsymbol{a}$. The codewords are the length T_s waveforms associated with the SOQPSK trellis branches. Since the waveform associated with each trellis branch is unique we have as many codewords as branches. We denote branch waveforms by $s_k(t)$ where the branch index $k \in \{0, 1, 2, \ldots, K-1\}$. For ease of notation we refer to SOQPSK codewords by $\{k_i\}$. The algorithm expects the sequence of a priori probability distributions of the codewords denoted by $\mathbf{P}(\boldsymbol{k}; I) = \{P_i(k_i = k; I)\}$. The probability distribution $P_n(k_n = k; I)$ is simply the likelihood values of SOQPSK trellis branches and given by

$$P_n(k_n = k; I) = C \exp\left\{\frac{2d_{k,n}}{N_0}\right\}, \qquad 0 \le k \le K-1 \tag{2.21}$$

where C is a normalization constant while $d_{k,n}$ is known as *branch metric increment* (BMI) and computed by

$$d_{k,n} = \int_{nT_s}^{(n+1)T_s} b(t;\boldsymbol{a}) s_k^*(t) dt. \tag{2.22}$$

where $(\cdot)^*$ denotes the *complex conjugate*. The factor of 2 in the exponent in (2.21) has been omitted in the literature, so we provide a proof of in Appendix A. The algorithm outputs the sequence of AP state transition probabilities $\{P_i((S_i, S_{i+1}) = (s, s'); O)\}$ where $P_i((S_i, S_{i+1}) = (s, s'); O) = P((S_i, S_{i+1}) = (s, s') | b(t;\boldsymbol{a}))$.

We recall that SOQPSK-MIL trellis has 8 branches in each time step while SOQPSK-TG trellis has 1024 branches. The large size of the SOQPSK-TG trellis renders any trellis based SOQPSK-TG detection computationally infeasible. The detection of SOQPSK-TG is

generally carried out with reduced complexity detection methods such as pulse amplitude modulation (PAM) [5, 6] and pulse truncation (PT) [7, 8]. While both methods result in 4-state SOQPSK-TG detectors, the PAM based SOQPSK-TG detector has superior performance and performs within 0.1 dB of the optimal detector. The capacity of SOQPSK-TG with PAM detection (SOQPSK-TG-PAM) is of great interest—considering its relatively low detection complexity and *near-optimal* performance—and is computed in the next chapter. Here we give a brief summary of this detector.

With pulse amplitude modulation (PAM) decomposition approach, a CPM signal can be expressed as a linear combination of R amplitude modulated pulses in the form

$$s(t;\boldsymbol{\alpha}) = \sqrt{\frac{E_s}{T_s}} \sum_{v=0}^{R-1} \sum_{i} b_{v,i} g_v(t - iT_s), \qquad R = 2 \cdot 3^{L-1} \tag{2.23}$$

where $\{g_v(t)\}$ are the modulated pulses and $\{b_{v,i}\}$ are called the *pseudo-symbols* [6]. The values of psuedo-symbols $\{b_{v,n}\}$ are determined by the sequence of ternary symbol up to symbol time n $(\alpha_0, \alpha_1, \ldots, \alpha_n)$. In the case of SOQPSK-TG $R = 4374$ and the PAM representation is unmanageable. However most of the energy is concentrated in the first two pulses in the sum (2.23), called the *principal pulses* in [25]. An approximation based on the principal pulses is given by

$$s(t;\boldsymbol{\alpha}) \approx \sqrt{\frac{E_s}{T_s}} \sum_{v=0}^{1} \sum_{i} \exp\{j\theta_{i-1}\} \beta_{v,i} g_v(t - iT_s) \tag{2.24}$$

where $\theta_{i-1} \in \{0, \pi/2, \pi, 3\pi/2\}$ is the phase state and $\{\beta_{v,i}\}$ are simplified pseudo symbols. The simplified pseudo symbols $\{\beta_{v,n}\}$ only depend on the current symbol value $\{\alpha_n\}$ as shown in Table I while the memory of SOQPSK-TG is isolated in the phase state θ_{n-1}.

We note that the approximation in (2.24) is only a function of the phase state values $\{\theta_{i-1}\}$ and the ternary branch symbols $\{\alpha_i\}$ and independent of the pulse length L. Therefore, given the one-to-one mapping between the trellis states and the phase states in (2.17) the 4-state trellis in Figure 2.5 can be used for *suboptimal* detection of SOQPSK-TG.

Table 2.1: Relationship between the ternary branch symbol α_n, and the pseudo-symbols $\beta_{v,n}(\alpha_n)$ for SOQPSK.

α_n	$\beta_{0,n}(\alpha_n)$	$\beta_{1,n}(\alpha_n)$
-1	$\exp\{-j\pi/2\} = -j$	$\exp\{-j\pi/4\} = \frac{\sqrt{2}}{2}(1-j)$
0	1	$\cos(\pi/4) = \frac{\sqrt{2}}{2}$
$+1$	$\exp\{+j\pi/2\} = +j$	$\exp\{+j\pi/4\} = \frac{\sqrt{2}}{2}(1+j)$

We close the discussion of SOQPSK-TG-PAM with the definitions of PAM-based BMIs [4]. Again we denote the branch index by $k \in \{0, 1, 2, 3, 4, 5, 6, 7\}$. Associated with every value of k are a specific information bit $a_n(k)$, a ternary symbol $\alpha_n(k)$, and a phase state $\theta_{n-1}(k)$. The PAM-based branch metric increment is given by [4]

$$d_{k,n} = e^{-j\theta_{n-1}(k)} \sum_{v=0}^{1} y_v(n)[\beta_v(\alpha_n(k))]^*, \qquad 0 \le k \le 7 \tag{2.25}$$

where the matched filter outputs $y_v(n)$ are given by

$$y_v(n) = \int_{nT_s}^{(n+L+1-v)T_s} b(t;\boldsymbol{a}) g_v(t - nT_s)\, dt, \qquad 0 \le v \le 1. \tag{2.26}$$

As in the case of SOQPSK-MIL and SOQPSK-TG, SOQPSK-TG-PAM branch metric increments are used to compute the sequence of a priori probability distributions of codewords $\mathbf{P}(\boldsymbol{m}; I)$ from equation (2.21). Further discussion of the PAM based detection of CPM and SOQPSK signals can be found in [26], [27] and [6].

Chapter 3

Capacity and Spectral Efficiency of SOQPSK

3.1 Introduction

So far we have provided the signal model for SOQPSK, and discussed SOQPSK detection as well as the BCJR algorithm [13]. Starting with this chapter we will present our contributions, which include the computation of the capacity and the pragmatic capacity of SOQPSK, and information rates of various SOQPSK coding schemes.

Channel capacity is an important performance bound for any channel over which communication systems are designed. It is the maximum reliable transmission rate achievable over a communication channel [28] expressed in bits/channel use. It establishes a yardstick by which practical systems can be judged. Channel capacity can be computed for a specific modulation scheme to establish its performance bounds. Despite the widespread use of SOQPSK its capacity through an additive white Gaussian noise (AWGN) channel has not yet been computed. In this chapter we compute the channel capacities of SOQPSK-MIL, SOQPSK-TG and SOQPSK-TG-PAM. The results show that all three schemes have comparable capacities. We then take bandwidth considerations into account and compute the

spectral efficiencies of all three schemes, which is the maximum reliable transmission rate per second when only 1 Hz of bandwidth is available expressed in bits/second/Hz. The results show that SOQPSK-TG and SOQPSK-TG-PAM have much higher spectral efficiencies than SOQPSK-MIL.

3.2 Capacity of SOQPSK

The *capacity* of a channel is defined as the maximum *mutual information* between the channel input and the channel output over all possible input distributions [28] expressed as

$$C = \max_{P_{\boldsymbol{A}}(\boldsymbol{A})} \lim_{N\to\infty} \frac{1}{N} I(\boldsymbol{A};\boldsymbol{B}) \text{ [bits/channel use]} \tag{3.1}$$

with mutual information defined as

$$I(\boldsymbol{A};\boldsymbol{B}) = E_{\boldsymbol{A},\boldsymbol{B}}\left(\log_2 \frac{P_{\boldsymbol{A}|\boldsymbol{B}}(\boldsymbol{A}|\boldsymbol{B})}{P_{\boldsymbol{A}}(\boldsymbol{A})}\right) \tag{3.2}$$

where $E_X(g(X))$ denotes the expected value of $g(X)$, $\boldsymbol{A} = (A_1, A_2, \ldots, A_N)$ is the vector of the transmitted symbol sequence, and $\boldsymbol{B}$ is the corresponding vector of channel outputs, i.e. $\boldsymbol{A}$ and $\boldsymbol{B}$ are sufficient statistics for the transmitted waveform $s(t;\boldsymbol{A})$ and the received waveform $b(t;\boldsymbol{A})$ respectively. A_i is a discrete random variable (RV) that denotes the transmitted symbol during the i-th symbol period. For a given modulation scheme the input symbol alphabet is fixed. In other words $A_i \in \mathcal{A}$ for all i where $\mathcal{A}$ is a particular symbol alphabet. Note that A_i is used to denote both a RV and a realization of the same random variable from the set $\mathcal{A}$. The meaning of A_i will be clear from the context. We also assume that the input symbol distribution $P_{\boldsymbol{A}}(\boldsymbol{A})$ is fixed. Hence the maximization over the input distribution is dropped and the channel capacity is given as the limit

$$C = \lim_{N\to\infty} \frac{1}{N} I(\boldsymbol{A};\boldsymbol{B}). \tag{3.3}$$

The channel is assumed to be an AWGN channel with a two-sided noise power spectral density of $N_0/2$. With memoryless modulation schemes the channel output only depends on the current channel input and the channel capacity can be evaluated over a single symbol interval from (3.3) with $N = 1$. However, for modulation schemes with memory, the output waveform is a function of the entire input symbol sequence [14]; consequently, the mutual information in (3.2) cannot be analytically evaluated. In [10], the BCJR algorithm—described in Chapter 2—is used to approximate the capacity of CPM as a function of *a posteriori* (AP) state transition probabilities. We apply this algorithm to SOQPSK to compute its capacity. Below is a detailed explanation of this algorithm and how it is applied to SOQPSK.

By using the *chain rule of information* [28] (3.3) can be expressed as

$$\begin{aligned} C &= \lim_{N\to\infty} \frac{1}{N} \sum_{i=1}^{N} I(A_i; \boldsymbol{B}|A_1^{i-1}) \\ &= \lim_{N\to\infty} \frac{1}{N} \sum_{i=1}^{N} H(A_i|A_1^{i-1}) - \frac{1}{N} \sum_{i=1}^{N} H(A_i|\boldsymbol{B}, A_1^{i-1}) \end{aligned} \tag{3.4}$$

where A_i^j denotes the symbol sequence $(A_i, A_{i+1}, \ldots, A_j)$ and the *conditional entropy* is defined as

$$H(A_i|A_1^{i-1}) = -E_{A_i, A_1^{i-1}} \left(\log_2 P_{A_i|A_1^{i-1}}(A_i|A_1^{i-1}) \right) \text{ [bits]}. \tag{3.5}$$

For notational convenience we drop the subscripts of the expected values, and the probability mass and density functions. The capacity expression in (3.5) can be further simplified by replacing A_1^{i-1} by S'_{i-1} which is the state of the trellis that models the modulation scheme at the beginning of the i-th symbol interval. In addition to S'_{i-1} being uniquely determined by A_1^{i-1} this is due to the Markov property of the trellis that models the modulation scheme with memory. Given the current state of the trellis S'_{i-1} and the channel output vector $\boldsymbol{B}$, the current input symbol A_i is independent of the sequence of past input symbols A_1^{i-1}.

Hence the capacity expression can be further simplified to

$$\begin{aligned} C &= \lim_{N\to\infty} \frac{1}{N}\sum_{i=1}^{N} H(A_i|A_1^{i-1}) - \frac{1}{N}\sum_{i=1}^{N} H(A_i|\boldsymbol{B}, S'_{i-1}) \\ &= \lim_{N\to\infty} \frac{1}{N}\sum_{i=1}^{N} H(A_i|A_1^{i-1}) + \frac{1}{N}\sum_{i=1}^{N} E[\log_2(P(A_i|\boldsymbol{B}, S'_{i-1}))] \\ &= \lim_{N\to\infty} \frac{1}{N}\sum_{i=1}^{N} H(A_i|A_1^{i-1}) + \frac{1}{N}\sum_{i=1}^{N}\sum_{A_i\in\mathcal{A}} P(A_i|\boldsymbol{B}, S'_{i-1})\log_2(P(A_i|\boldsymbol{B}, S'_{i-1})). \end{aligned} \tag{3.6}$$

The first expression in (3.6) is a function of the joint probability distribution of the input symbol sequence $\boldsymbol{A}$. For independent and uniformly distributed input symbols it is equal to the number of bits per input symbol, m, and the capacity is called the *symmetric information rate* [29]. $P(A_i|\boldsymbol{B}, S'_{i-1})$ in the second expression is the AP probability distribution of the input symbols over the branches leaving the known state S'_{i-1} during the symbol interval i. This probability distribution is very difficult to obtain analytically, but it can be estimated by simulations with the use of the BCJR algorithm described in Chapter 2. This is done by grouping the state transitions from the state S'_{i-1} according to the input symbols occuring on them S'_{i-1} and normalizing the total probability to 1. Let U_{A_i} be the set of all state transitions labeled with A_i. Then we have

$$P(A_i|\boldsymbol{B}, S'_{i-1}) = \frac{\sum_{(S'_{i-1},S_i)\in U_{A_i}} P(S'_{i-1}, S_i|\boldsymbol{B})}{\sum_{(S'_{i-1},S_i)} P(S'_{i-1}, S_i|\boldsymbol{B})} \tag{3.7}$$

where (S'_{i-1}, S_i) denotes a state transition from state S'_{i-1} to state S_i while $P(S'_{i-1}, S_i|\boldsymbol{B})$ is the AP probability associated with the transition. Here it is important to note that S'_{i-1} is a known fixed state while S_i is any state that can be reached from S'_{i-1} on the trellis in a single step. It follows that the numerator in (3.7) is the total probability of the state transitions from state S'_{i-1} occuring with the transmission of A_i. The denominator is the

total probability of the state transitions from state S'_{i-1}, and is used to guarantee that

$$\sum_{A_i \in \mathcal{A}} P(A_i|\boldsymbol{B}, S'_{i-1}) = 1. \tag{3.8}$$

In order to compute (3.7) the transmitted symbol sequence $\boldsymbol{A}$ must be known so that it can used to determine the trellis state sequence vector $\boldsymbol{S} = (S'_0, S'_1, \ldots, S'_{N-1})$. For a specific realization of the input symbol sequence $\boldsymbol{A}$ and the corresponding channel output sequence $\boldsymbol{B}$ AP state transition probabilities $P(S_{i-1}, S_i|\boldsymbol{B})$ for all state transitions for all i can be computed with the BJCR algorithm. With all state transition probabilities available and the state sequence vector $\boldsymbol{S}$ known $P(A_i|\boldsymbol{B}, S'_{i-1})$ can be computed through (3.7) and can be used in (3.6) to compute the channel capacity.

In the case of SOQPSK the input symbol alphabet is binary. There are two branches leaving every state of the trellis; one labeled with binary '1' and the other one labeled with binary '0'. Given that the input symbols are independent and uniformly distributed the first expression in (3.6) is equal to 1. If we let S_i^1 denote the state that can be reached from the known state S'_{i-1} with the transmission of binary '1' and let S_i^0 denote the state that can be reached from the known state S'_{i-1} with the transmission of binary '0' and let

$$p_i^1 \triangleq P(S'_{i-1}, S_i^1|\boldsymbol{B}) \tag{3.9}$$

$$p_i^0 \triangleq P(S'_{i-1}, S_i^0|\boldsymbol{B}) \tag{3.10}$$

$$p_i \triangleq \frac{p_i^1}{p_i^1 + p_i^0} \tag{3.11}$$

the approximated channel capacity of SOQPSK is given by the expression

$$C_{\text{SOQPSK}} = 1 - \lim_{N\to\infty} \frac{1}{N} \sum_{i=1}^{N} H(p_i) \tag{3.12}$$

where the *entropy* $H(p_i)$ is defined as

$$H(p_i) = -p_i \log_2 p_i - (1 - p_i) \log_2 (1 - p_i). \tag{3.13}$$

Note that the channel capacity of SOQPSK doesn't depend on the specific mapping between the binary input symbols and the trellis branches. There are only two distinct mappings because we only consider the trellis branches leaving a known state. Switching from one mapping to the other one is equivalent to replacing p_i by $(1 - p_i)$ in (3.13) which results in the same entropy and consequently the same channel capacity.

So far we have shown that the capacity of SOQPSK can be computed by applying the BCJR algorithm— described in Chapter 2—to an infinitely long SOQPSK channel output waveform $\boldsymbol{B}$ with the knowledge of the input symbol sequence $\boldsymbol{A}$. Hence, we use a simulation-based time-average to compute the capacity of SOQPSK. We intend to compute the capacity of SOQPSK through an AWGN channel for various values of signal to noise ratio (SNR) per information bit E_b/N_0 where E_b is the energy per information bit and $N_0/2$ is the value of the two-sided noise spectral density. The value of E_b cannot be controlled during simulations since the amount of information per transmitted symbol is not known in advance. During the simulations we can only control the SNR per modulated symbol E_s/N_0. The relationship between E_b/N_0 and E_s/N_0 is very straightforward. Let's assume that capacity for an E_s/N_0 value is computed and its value is denoted by $C_{\text{SOQPSK}}(E_s/N_0)$. The energy per information bit E_b is equal to the energy per symbol E_s normalized by the average amount of information bits in each symbol, i.e. the computed channel capacity. This follows from the well-known relationship $E_c = RE_b$ where E_c is the energy per coded bit and R is the *code rate* for a coding scheme [23]. Therefore once the SOQPSK capacity is computed for an E_s/N_0 value, the SNR per information bit E_b/N_0 is computed from

$$E_b/N_0 \left(C_{\text{SOQPSK}}(E_s/N_0), E_s/N_0\right) = \frac{E_s/N_0}{C_{\text{SOQPSK}}(E_s/N_0)}. \tag{3.14}$$

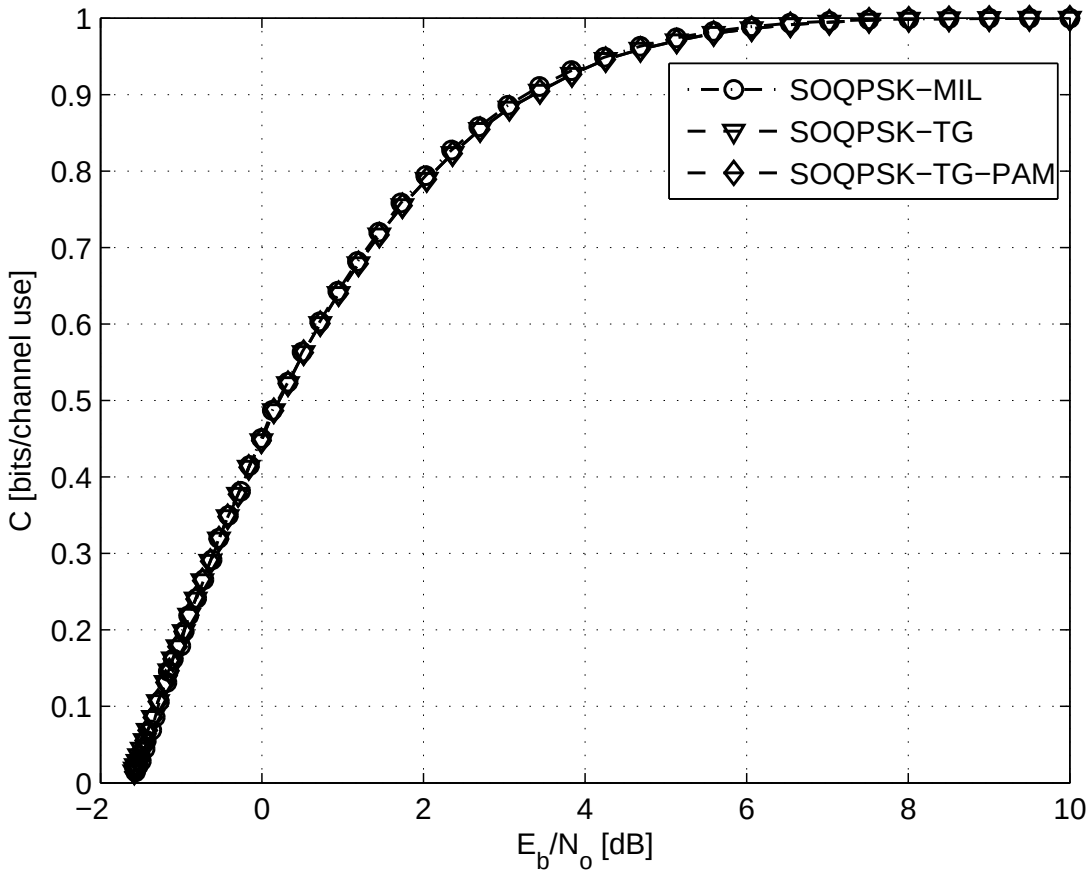

Figure 3.1: Channel capacities of SOQPSK-MIL, SOQPSK-TG, and SOQPSK-TG-PAM.

We computed the channel capacities of all three SOQPSK schemes, i.e. SOQPSK-MIL, SOQPSK-TG and SOQPSK-TG-PAM, for a large set of E_s/N_0 values. For each E_s/N_0 value we generated a channel input sequence $\boldsymbol{A}$ of independent and uniformly distributed bits. The input sequence $\boldsymbol{A}$ is modulated and white Gaussian noise samples are added to the output of the modulator with variance adjusted according to the value of E_s/N_0, which gives us the channel output vector $\boldsymbol{B}$. The state sequence $\boldsymbol{S}$ is computed from $\boldsymbol{A}$ and used along with $\boldsymbol{B}$ to compute the channel capacity by the algorithm desribed above, and the E_b/N_0 value is computed from (3.14). The capacity curves of SOQPSK-MIL, SOQPSK-TG and SOQPSK-TG-PAM are shown in Figure 3.1. Note that the capacities of SOQPSK-MIL and SOQPSK-TG are computed on full (i.e. exact) trellises, a 4-state trellis for SOQPSK-MIL and a 512-state trellis for SOQPSK-TG, while the capacity of SOQPKS-TG-PAM is computed on

the reduced (i.e. approximate) 4-state trellis. In other words, the capacity of SOQPSK-TG-PAM is the capacity of SOQPSK-TG constrained on the PAM-based suboptimal detection method. It is also important to note that SOQPSK-MIL and SOQPSK-TG-PAM have the same detection complexity, which makes it fair to compare their performances.

Looking at Figure 3.1 we observe that the capacity of SOQPSK-MIL is slightly greater than the capacities of SOQPSK-TG and SOQPSK-TG-PAM. The capacity of SOQPSK-TG is greater than the capacity of SOQPSK-TG-PAM which is expected considering the suboptimality of the PAM-based SOQPSK-TG detection. However, the difference is slim which can be explained by the the fact that PAM-based SOQPSK-TG detectors perform within 0.1 dB the optimal SOQPSK-TG detector [30]. The capacity for all three schemes approach 1 as E_b/N_0 increases since SOQPSK is a binary modulation. The capacities of all three schemes approach 0 as E_b/N_0 approaches approximately -1.6 dB. This is consistent with the fact that for any communication system operating over an AWGN channel the minimum required value of E_b/N_0 is -1.6 dB [14]. Below this limit information cannot be transmitted reliably.

3.3 Spectral Efficiency of SOQPSK

The channel capacity computed in bits/channel as a function of SNR is a good measure of performance limits of communications systems but doesn't take into account an extremely important and scarce resource for communication systems: bandwidth. It is crucial to investigate the maximum achievable rate with SOQPSK modulation over a finite bandwidth. The *spectral efficiency* of a modulation scheme is defined as the maximum achievable transmission rate per second when only 1 Hz of bandwidth is available and is given in bits/s/Hz. Fortunately the previously computed channel capacities of SOQPSK schemes can be used to compute the respective spectral efficiencies. To compute the spectral efficiency of SOQPSK we follow the approach taken in [11].

We define the *normalized symbol rate* R_s as the symbol rate when only 1 Hz of bandwidth is available, i.e. the number of times per second the channel can be used if only 1 Hz of bandwidth is available. Therefore, R_s is given in channel use/s/Hz. Naturally R_s is specific to the modulation scheme used and needs to be computed for both SOQPSK-TG and SOQPSK-MIL. Once R_s is computed the spectral efficiency can be computed by the equation

$$S_{\text{SOQPSK}}[\text{bits/s/Hz}] \triangleq C_{\text{SOQPSK}}[\text{bits/channel use}] \cdot R_s \text{ [channel use/s/Hz]} \tag{3.15}$$

where S_{SOQPSK} denotes the spectral efficiency of SOQPSK. The capacity of SOQPSK is bounded by 1 for SOQPSK being a binary modulation scheme, which results in

$$S_{\text{SOQPSK}} \leq R_s. \tag{3.16}$$

For a specific scheme for example SOQPSK-MIL C_{SOQPSK} will be replaced by the capacity of SOQPSK-MIL and R_s will be replaced by that of SOQPSK-MIL. For SOQPSK-TG-PAM C_{SOQPSK} will be replaced by the capacity of SOQPSK-TG-PAM and and R_s will be replaced by that of SOQPSK-TG. In order to proceed we need to compute the normalized symbol rates R_s of SOQPSK-MIL and SOQPSK-TG. R_s is computed from the *power spectral density* (PSD) of each modulation scheme for a transmission rate of 1 symbol per second. We denote the PSD by $S_{SS}(f)$. From the PSD the *normalized bandwidth* denoted by B_s is computed. The bandwidth B_s is defined as the one that contains 99.9 % of the signal power, hence it satisfies the equation

$$\frac{\int_{-B_s}^{B_s} S_{SS}(f)\,\mathrm{d}f}{\int_{-\infty}^{\infty} S_{SS}(f)\,\mathrm{d}f} = 0.999. \tag{3.17}$$

Once B_s is computed from the PSD, the normalized symbol rate is computed as

$$R_s = \frac{1}{B_s}. \tag{3.18}$$

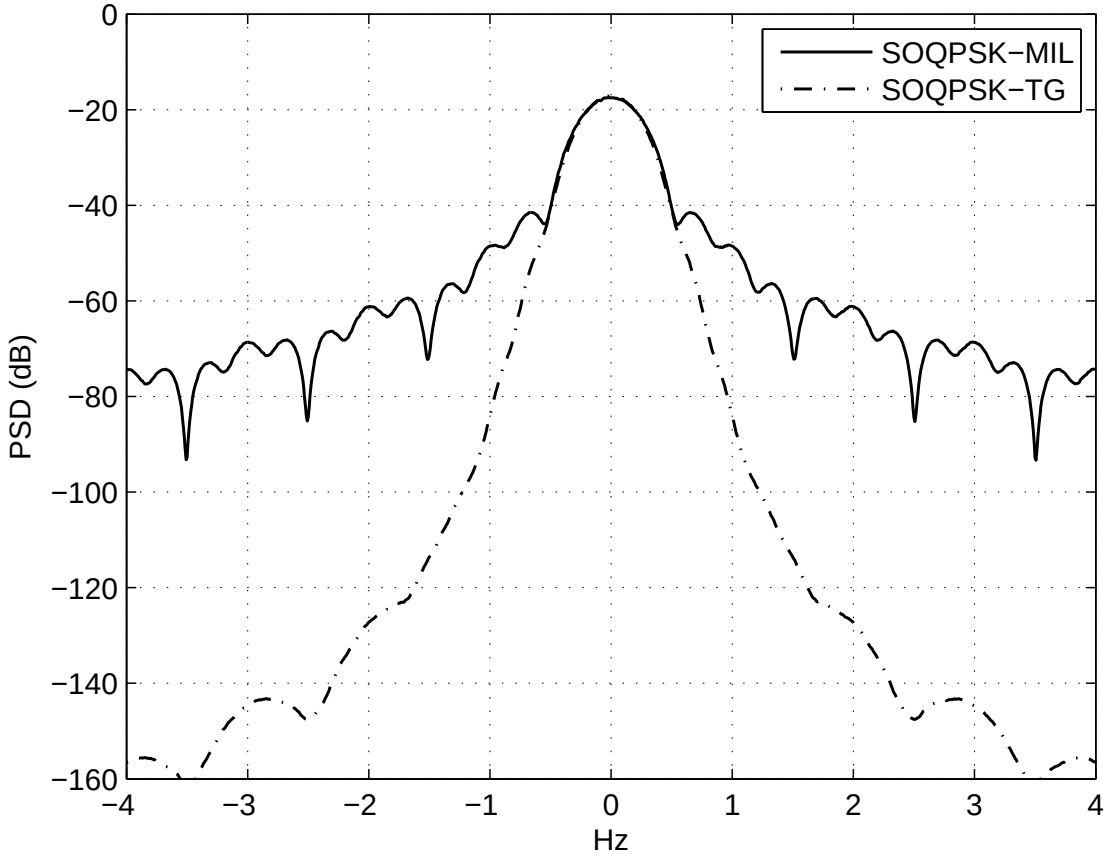

Figure 3.2: Power spectral densities of SOQPSK-MIL, SOQPSK-TG for a transmission rate of 1 symbol/s.

We follow a simulation-based approach (i.e. *Welch periodogram method* [31]) to compute the PSDs of SOQPSK-TG and SOQPSK-MIL. The resulting PSDs for SOQPSK-MIL and SOQPSK-TG are shown in Figure 3.2. We observe that the PSD of SOQPSK-TG is significantly more compact than that of SOQPSK-MIL due to its longer and smoother frequency pulse.

By using the PSDs computed and equations (3.17), (3.18) we computed the normalized symbol rate R_s for SOQPSK-TG and SOQPSK-MIL. Normalized symbol rate R_s of SOQPSK-TG is equal to 1.00 while it is equal to 0.56 for SOQPSK-MIL. From (3.16) these normalized symbol rate values imply that the spectral efficiency of SOQPSK-MIL is bounded by 0.56 even if we let the SNR per information bit E_b/N_0 increase without bounds. On the

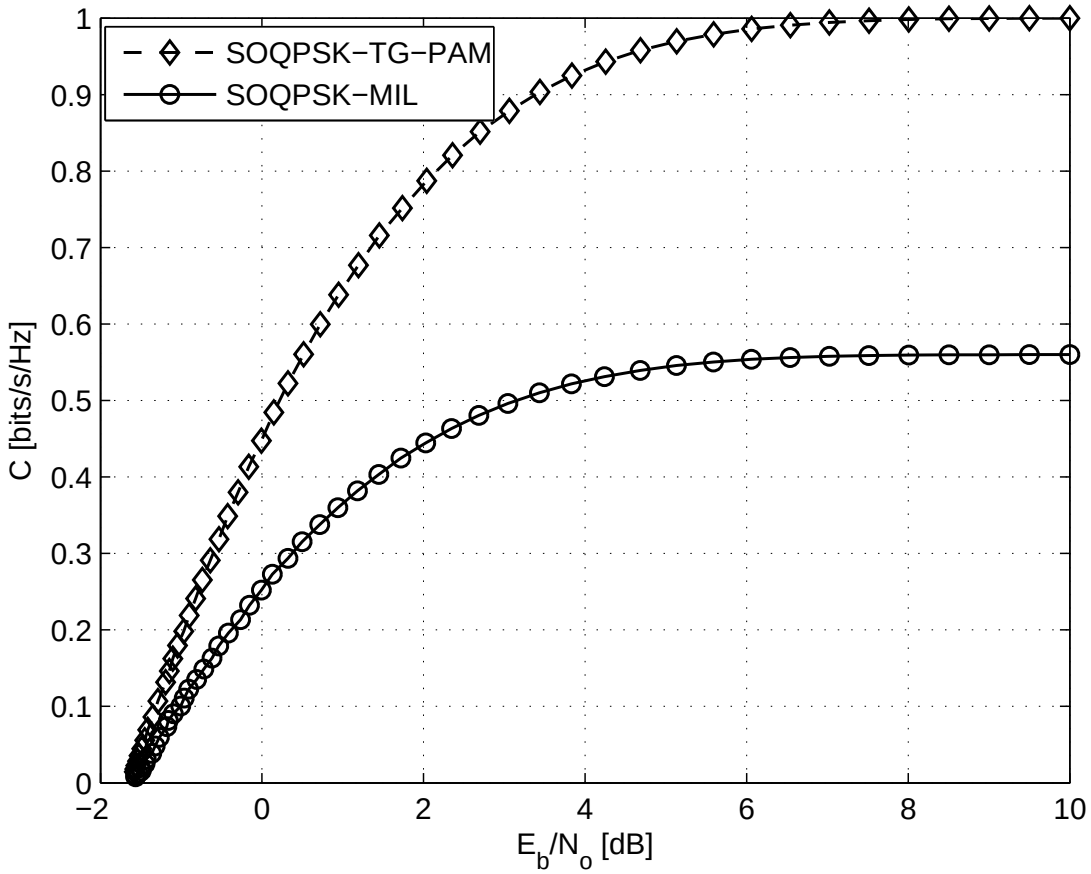

Figure 3.3: Spectral efficiencies of SOQPSK-MIL, SOQPSK-TG-PAM.

other hand we can reliably transmit at any rate up to 1.00 bits/s/Hz with an SOQPSK-TG system.

We generated spectral efficiency curves for SOQPSK-TG-PAM and SOQPSK-MIL shown in Figure 3.3. Spectral efficiency curve for SOQPSK-TG is omitted primarily because it is slightly higher than that of SOQPSK-TG-PAM despite much higher detection complexity. Taking into account the power spectrums of SOQPSK-TG and SOQPSK-MIL results in a more accurate performance comparison between SOQPSK-TG-PAM and SOQPSK-MIL. We conclude that SOQPSK-TG-PAM is spectrally a lot more efficient than SOQPSK-MIL despite having the same detection compexity.

Chapter 4

Pragmatic Capacity of SOQPSK

4.1 Introduction

We computed the capacity of SOQPSK in the previous chapter. In this chapter we evaluate the *pragmatic capacity* [12] of SOQPSK, the channel capacity of SOQPSK when SOQPSK is constrained to be used as a modulation scheme rather than a constituent code in a serially concatenated coding scheme [32]. We show that the pragmatic capacities of SOQPSK-MIL and SOQPSK-TG-PAM are slightly lower than their respective channel capacities. We also investigate how the pragmatic capacity of SOQPSK varies with different mappings between the input symbols and the branches of the trellis that models SOQPSK modulation.

4.2 Pragmatic Coded Modulation

Pragmatic coded modulation (also known as *bit-interleaved coded modulation*) consists of a binary encoder concatenated with a memoryless modulator through a binary interleaver [12]. As shown in Figure 4.1 the bit sequence at the output of the binary encoder $\boldsymbol{a} = (a_1, a_2, \ldots, a_N)$ is interleaved and the interleaved bit sequence $\boldsymbol{a'}$ is divided into subsequences of m bits denoted by $\boldsymbol{a'_i} = (a'_{i,1}, a'_{i,2}, \ldots, a'_{i,m})$. i.e. $\boldsymbol{a'} = (a'_1, a'_2, \ldots, a'_N)$. Then the sequence of subsequences $\boldsymbol{a'}$ is mapped onto a signal $s(t; \boldsymbol{a'})$ by a memoryless M-ary modulator where $M = 2^m$.

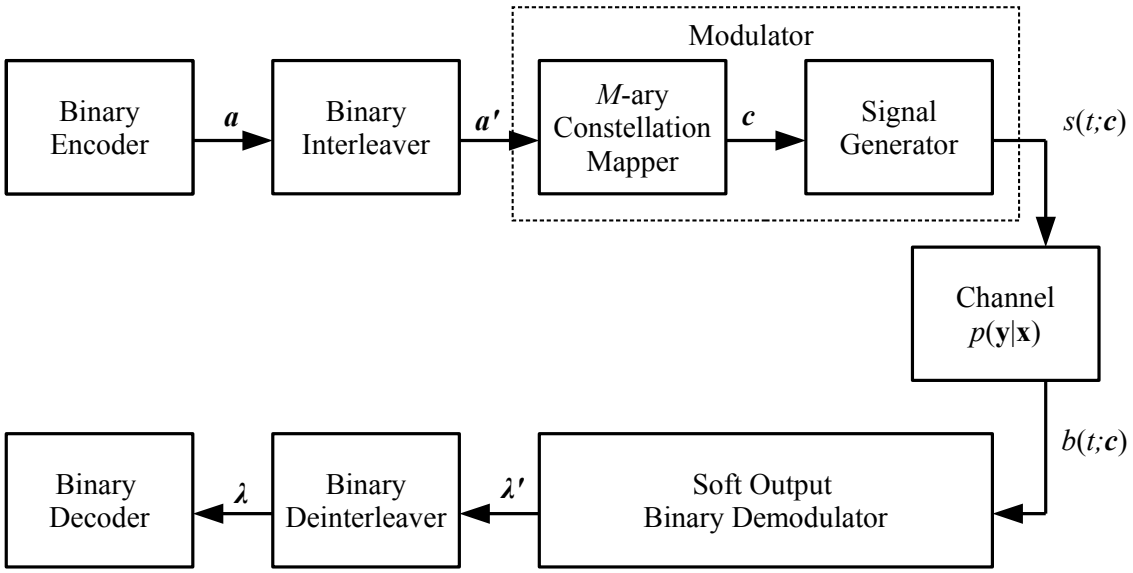

Figure 4.1: Block diagram of pragmatic coded modulation.

The modulation operation can be divided into two operations such as constellation mapping and signal generation. Constellation mapping refers to the mapping of the m-bit subsequences $\boldsymbol{a'}$ onto a sequence of constellation points (i.e. M-ary symbols) $\boldsymbol{c} = (c_1, c_2, ..., c_{N/m})$ in the constellation of an M-ary modulator denoted by $\mathcal{C}$. Signal generation refers to the conversion of the sequence of constellation points $\boldsymbol{c}$ into the physical signal $s(t; \boldsymbol{c})$.

In pragmatic coded modulation encoding and constellation mapping operations are separated by a binary interleaver. Therefore, these two operations are independent, which eliminates the need for joint optimization. The demodulator in a pragmatic scheme computes the *log likelihood ratio* (LLR) of each transmitted bit given by

$$\lambda'_i = \log \frac{P(a'_i = 1|b(t; \boldsymbol{c}))}{P(a'_i = 0|b(t; \boldsymbol{c}))} \tag{4.1}$$

where $b(t; \boldsymbol{c})$ is the channel output. These LLRs are deinterleaved and fed into the binary decoder which assumes them to come from m memoryless independent binary channels. The binary decoder uses the LLRs to estimate the underlying information sequence.

Pragmatic coded modulation is an alternative to *coded modulation* (e.g. trellis coded

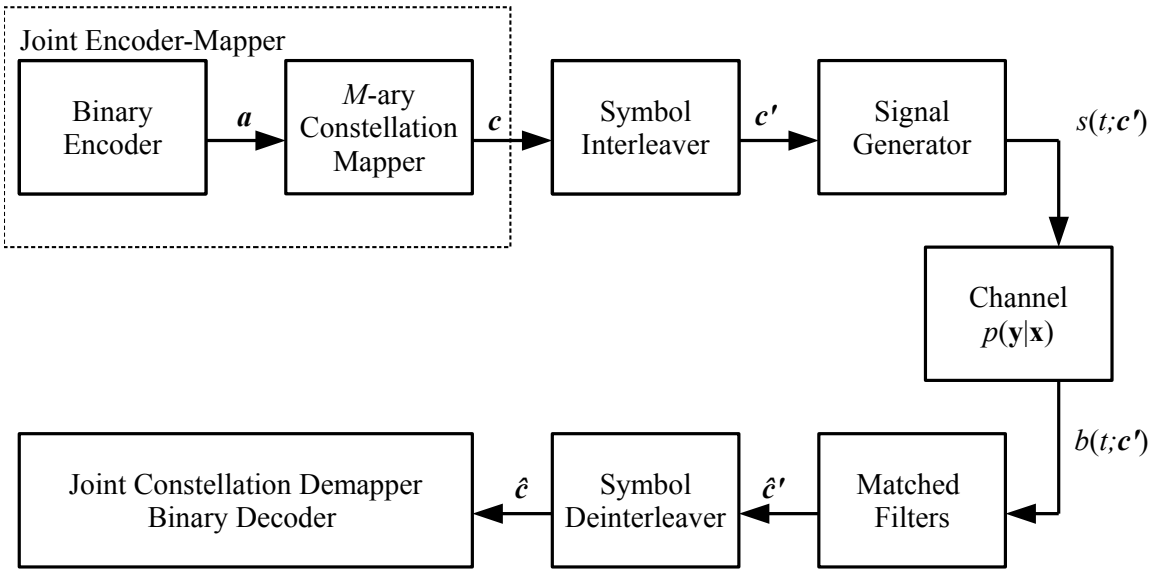

Figure 4.2: Block diagram of coded modulation.

modulation (TCM) [23]) shown in Figure 4.2 where the encoding and the constellation mapping operations are done jointly so that only constrained sequences of M-ary constellation points are transmitted. A rate k/m coded modulation scheme takes a sequence of information bits, k bits in each symbol interval T_s, and outputs a sequence of M-ary constellation points $\boldsymbol{c}$, one M-ary symbol in each symbol interval T_s. The symbol sequence $\boldsymbol{c}$ is then interleaved and the interleaved sequence $\boldsymbol{c'}$ is converted into the physical signal $s(t;\boldsymbol{c'})$. For example a rate 2/3 TCM encoder consists of a binary convolutional encoder that takes 2 bits in each symbol interval and outputs an 8-phase-shift keying (8-PSK) symbol defined on a specific 8-PSK constellation. On the receiver side the received signal $s(t;\boldsymbol{c'})$ is projected onto the signal space of the M-ary constellation $\mathcal{C}$ by matched filters to generate the sequence of sufficient statistics $\hat{\boldsymbol{c}}'$. The sequence $\hat{\boldsymbol{c}}'$ is then deinterleaved and fed into the joint demapper/decoder to recover the underlying information sequence.

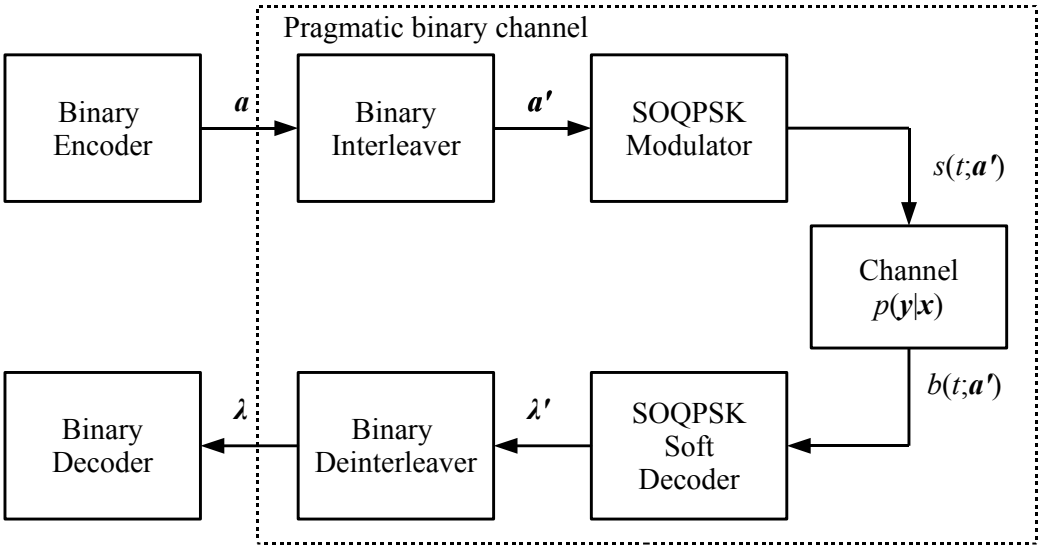

Figure 4.3: Block diagram of pragmatic coded SOQPSK modulation.

4.3 Pragmatic Coded SOQPSK

In [11], the use of CPM in a pragmatic coded modulation scheme is considered. In Chapter 2, we showed that an SOQPSK modulator can be decomposed into a continuous-phase encoder (CPE) and a memoryless modulator (MM). In the pragmatic coded modulation context the CPE is the analog of the constellation mapper and the MM is the analog of the signal generator. The difference between a pragmatic coded SOQPSK scheme and a coded SOQPSK scheme lies on whether the CPE is separated from the MM and treated as a code on the receiver side or CPE and MM together are treated as a memoryless modulator.

In a pragmatic SOQPSK scheme SOQPSK (i.e. CPE and MM together) is treated as a memoryless binary modulation as shown in Figure 4.3. A soft output SOQPSK decoder computes the LLRs, $\boldsymbol{\lambda'}$, from the a posteriori (AP) bit probabilities $\{P(a'_i = 1|b(t;\boldsymbol{a'}))\}$ by using the BCJR algorithm described in Chapter 2. Then the LLRs are deinterleaved and fed into the binary decoder that assumes them to be generated by a memoryless demodulator. The interleaver, the modulator, the channel, the soft output decoder and the deinterleaver can together be modeled a *pragmatic binary channel*.

In the coded SOQPSK scheme the CPE is used as a constituent code in a serially con-

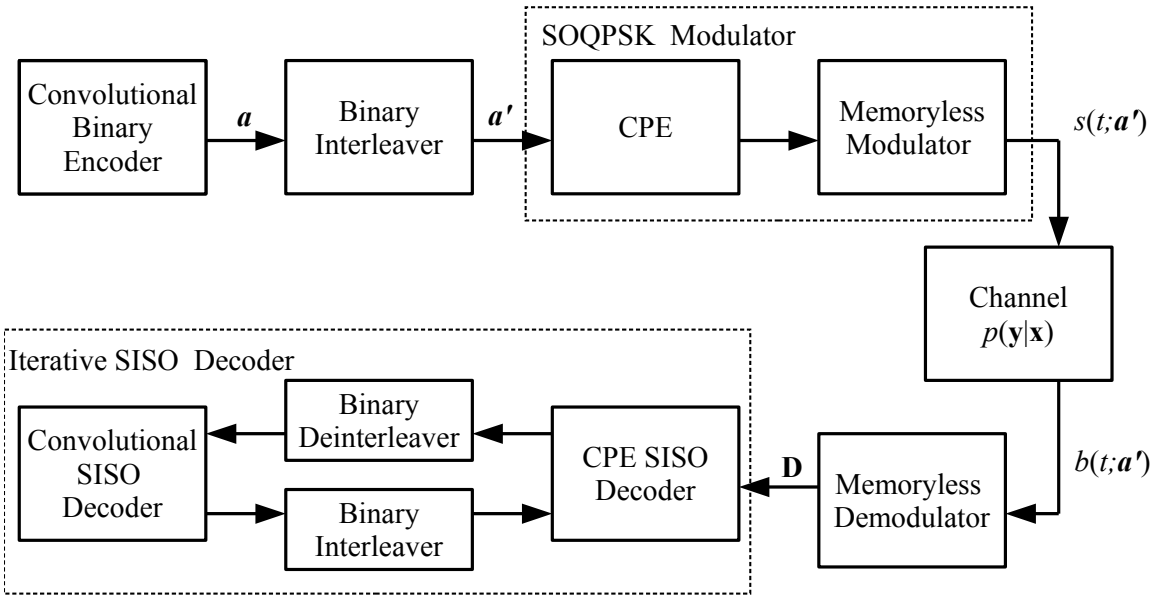

Figure 4.4: Block diagram of serially concatenated coded SOQPSK modulation.

catenated code [32] (SCC) as shown in Figure 4.4. An SCC consists of two constituent codes separated by a binary interleaver. We refer to the coding scheme in Figure 4.4 as serially concatenated coded SOQPSK (SC-SOQPSK). With an SC-SOQPSK on the transmitter side the output of a binary convolutional encoder $\boldsymbol{a}$ is first interleaved and then modulated by an SOQPSK modulator viewed as an encoder. On the receiver side the memoryless demodulator generates the matrix of branch metric increments denoted by $(\mathbf{D})_{kn} = d_{k,n}$. The elements of $\mathbf{D}$ are used by the iterative *soft-input soft-output* (SISO) [24] decoder to recover the underlying information sequence. Iterative decoding is a distinct feature of SCCs where the decoding operation is carried out iteratively between two SISO decoders by exchanging appropriate probabilities (i.e. soft inputs/outputs). Note that the main difference between coded SOQPSK and pragmatic coded SOQPSK is that in coded SOQPSK iterations take place between the SOQPSK decoder (i.e. CPE SISO decoder) and another binary decoder.

4.4 Pragmatic Capacity of SOQPSK

The channel capacity of SOQPSK—computed in Chapter 3—applies to any SOQPSK system without constraints. It is the highest transmission rate achievable for any communication system that employs SOQPSK modulation. An important question we intend to answer in this section is the performance limits of pragmatic SOQPSK schemes where SOQPSK is treated as a memoryless modulation as opposed to a code.

In a pragmatic scheme the modulator, along with the channel, is modeled as a set of m parallel independent and memoryless binary channels [12]. The capacity of this *pragmatic channel* is called the *pragmatic capacity* and is defined as the sum of the capacities of these m equivalent binary channels. Let $\boldsymbol{A} = (A_1, A_2, \ldots, A_N)$ be the binary input vector to the pragmatic channel and let $\boldsymbol{A}_i = (A_{i,1}, A_{i,2}, \ldots, A_{i,m})$ be the m-bit subsequence transmitted during the ith symbol period, i.e. $\boldsymbol{A} = (\boldsymbol{A}_1, \boldsymbol{A}_2, \ldots, \boldsymbol{A}_{N/m})$. Note that A_i is a binary random variable. The pragmatic capacity is given by

$$C_{\mathrm{P}} = \lim_{N\to\infty} \frac{1}{N/m} \sum_{i=1}^{N/m} \sum_{j=1}^{m} I(A_{i,j}; \boldsymbol{B}) \tag{4.2}$$

where $\boldsymbol{B}$ is the output of the pragmatic channel. Since SOQPSK is a binary modulation scheme, (4.2) reduces to

$$C_{\text{P-SOQPSK}} = \lim_{N\to\infty} \frac{1}{N} \sum_{i=1}^{N} I(A_i; \boldsymbol{B}) \tag{4.3}$$

which can be further simplified to

$$C_{\text{P-SOQPSK}} = \lim_{N\to\infty} \frac{1}{N} \sum_{i=1}^{N} H(A_i|A_1^{i-1}) - \frac{1}{N} \sum_{i=1}^{N} H(A_i|\boldsymbol{B}). \tag{4.4}$$

At this point it is important to show that the pragmatic capacity of SOQPSK is upper-

bounded by the capacity of SOQPSK. Because conditioning reduces entropy [28] we have

$$\frac{1}{N}\sum_{i=1}^{N} H(A_i|\boldsymbol{B}) \geq \frac{1}{N}\sum_{i=1}^{N} H(A_i|\boldsymbol{B}, A_1^{i-1}) \tag{4.5}$$

which is equivalent to

$$-\frac{1}{N}\sum_{i=1}^{N} H(A_i|\boldsymbol{B}) \leq -\frac{1}{N}\sum_{i=1}^{N} H(A_i|\boldsymbol{B}, A_1^{i-1}). \tag{4.6}$$

By adding $\frac{1}{N}\sum_{i=1}^{N} H(A_i|A_1^{i-1})$ to both sides and taking the limit we get

$$\lim_{N\to\infty} \frac{1}{N}\sum_{i=1}^{N} H(A_i|A_1^{i-1}) - \frac{1}{N}\sum_{i=1}^{N} H(A_i|\boldsymbol{B}) \leq \lim_{N\to\infty} \frac{1}{N}\sum_{i=1}^{N} H(A_i|A_1^{i-1}) - \frac{1}{N}\sum_{i=1}^{N} H(A_i|\boldsymbol{B}, A_1^{i-1}) \tag{4.7}$$

where the left hand side is the pragmatic capacity of SOQPSK and the right hand side is the capacity of SOQPSK. Therefore, we conclude that

$$C_{\text{P-SOQPSK}} \leq C_{\text{SOQPSK}} \tag{4.8}$$

which is expected since the pragmatic capacity of SOQPSK is the capacity of SOQPSK with a constraint.

Assuming that the elements of $\boldsymbol{A}$ are independent and uniformly distributed, and using the definition of entropy, (4.4) reduces to

$$\begin{aligned} C_{\text{P-SOQPSK}} &= 1 + \lim_{N\to\infty} \frac{1}{N}\sum_{i=1}^{N} E[\log_2(P(A_i|\boldsymbol{B}))] \\ &= 1 + \lim_{N\to\infty} \frac{1}{N}\sum_{i=1}^{N}\sum_{A_i=0}^{1} P(A_i|\boldsymbol{B})\log_2(P(A_i|\boldsymbol{B})). \end{aligned} \tag{4.9}$$

The a posteriori (AP) distribution $P(A_i|\boldsymbol{B})$ is difficult to obtain analytically, hence we use a simulation based approach to estimate $P(A_i|\boldsymbol{B})$. We note that $P(A_i|\boldsymbol{B})$ is the AP dis-

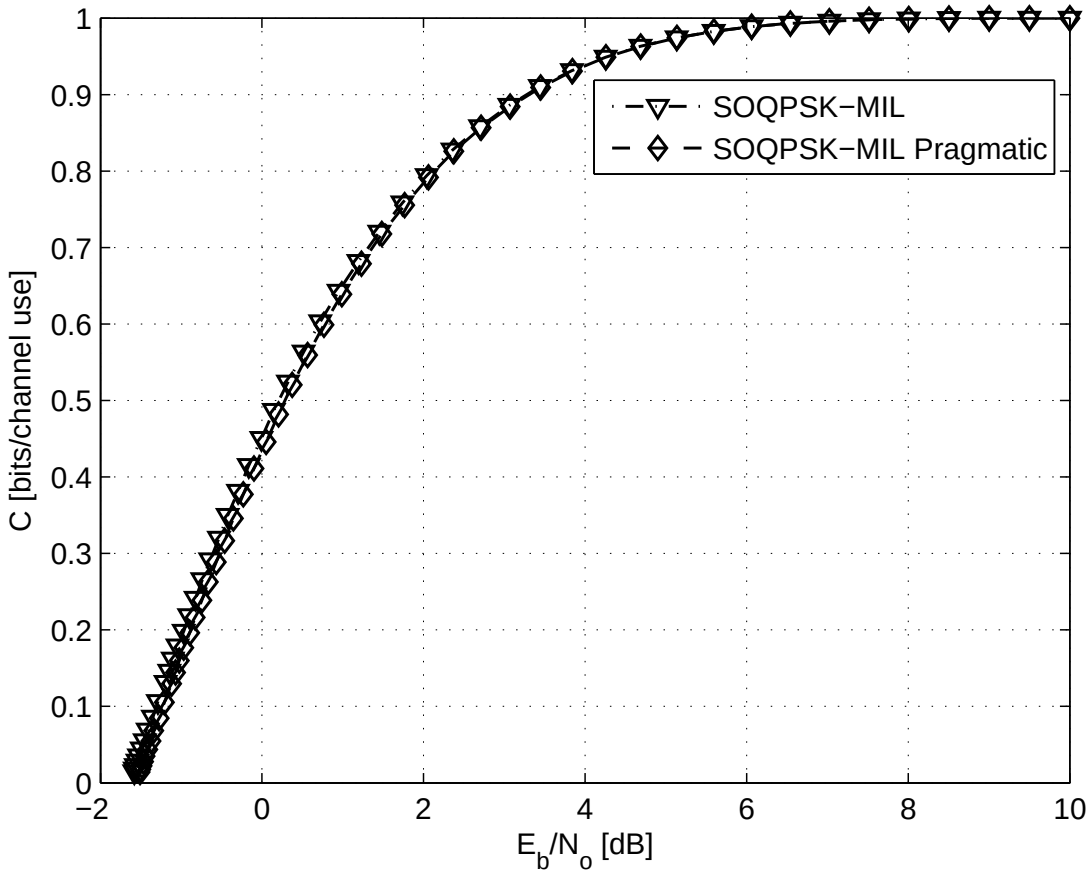

Figure 4.5: Capacity and pragmatic capacity of SOQPSK-MIL.

tribution of the binary input symbol A_i and can be computed by using the BCJR algorithm—described in Chapter 2— from an input sequence $\boldsymbol{A}$ and the corresponding channel output sequence $\boldsymbol{B}$. Therefore, we use the BCJR algorithm over long simulations to estimate the pragmatic capacity of SOQPSK. We note that the pragmatic capacity of SOQPSK is simply the average information rate at the output of the soft-output SOQPSK decoder shown in Figure 4.3. It is also important to note that in the pragmatic capacity estimation of SOQPSK the AP probabilities of all trellis branches are used while for the capacity estimation of SOQPSK we only used the AP probabilities of the branches leaving known state sequence $\boldsymbol{S}$. Therefore, the pragmatic capacity of SOQPSK—unlike the channel capacity of SOQPSK—is a function of the mapping between the input symbols and the trellis branches.

We computed the pragmatic capacities of SOQPSK-MIL and SOQPSK-TG-PAM for a

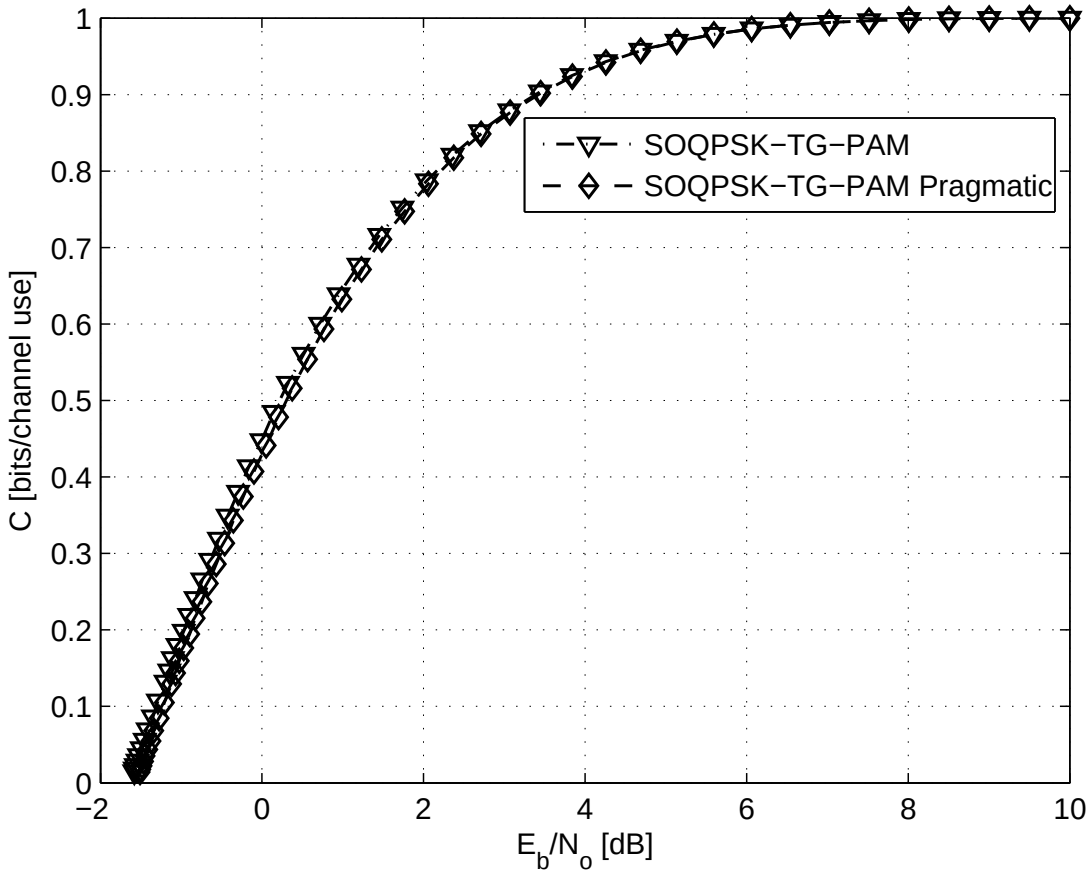

Figure 4.6: Capacity and pragmatic capacity of SOQPSK-TG-PAM.

large number of SNR per information bit E_b/N_0 values. The simulation method is identical to that used to compute the channel capacity of SOQPSK in Chapter 3. Figure 4.5 and Figure 4.6 show the capacity and the pragmatic capacity curves of SOQPSK-MIL and SOQPSK-TG-PAM respectively. In both plots we observe that the pragmatic capacity is very close to but slightly lower than the channel capacity. This result is very important since it implies that the performance improvement achieved by using SOQPSK as an encoder as opposed to a memoryless modulator operating with an independent binary encoder is very slim. A performance very close to the channel capacity of SOQPSK can be achieved with a binary code operating on the LLRs outputted by a pragmatic binary channel where SOQPSK is the modulation scheme.

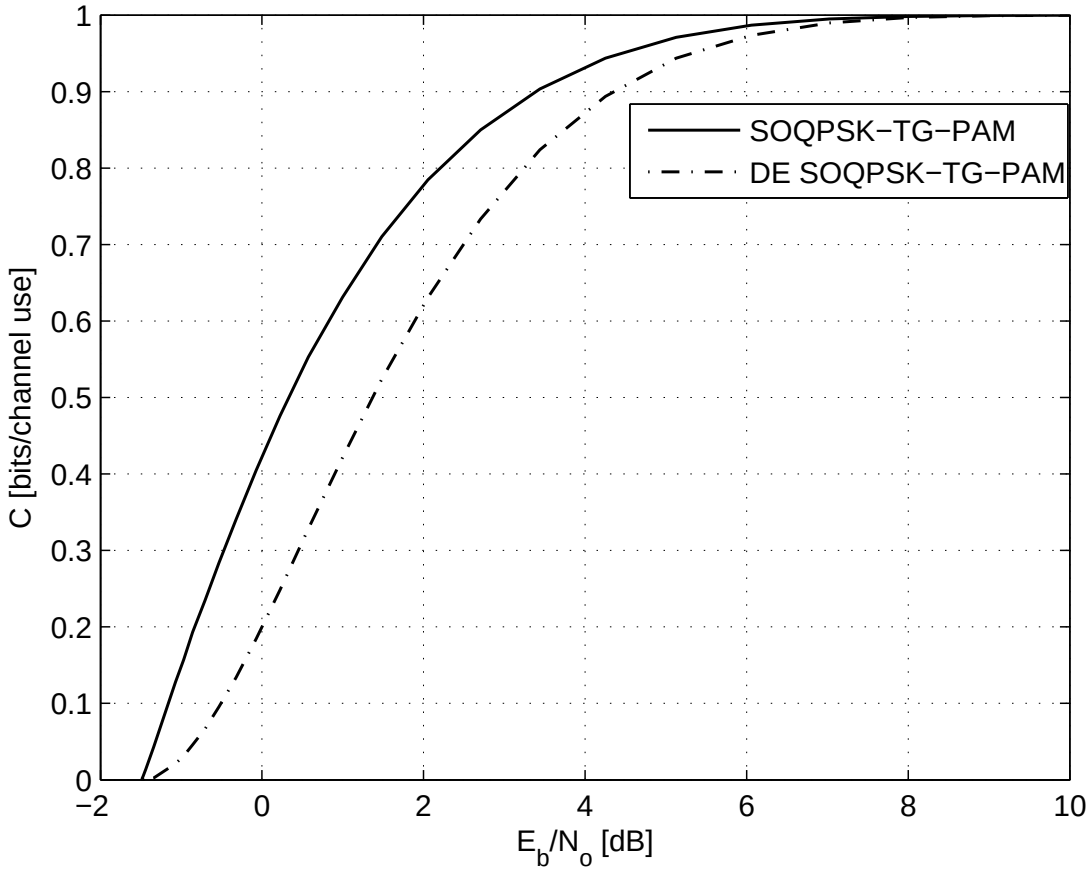

Figure 4.7: Pragmatic capacities of SOQPSK-TG-PAM and differentially encoded SOQPSK-TG-PAM.

4.5 Optimal Mapping for Pragmatic Capacity of SOQPSK

As we mentioned earlier the pragmatic capacity of SOQPSK is a function of the mapping between the input symbols and the branches of the trellis that models the SOQPSK modulation. In this section we investigate the optimality of the natural mapping of SOQPSK exhaustively, by computing the pragmatic capacity of SOQPSK for all possible mappings. We only permute the labels of the branches with the same starting state since the modulator cannot have two different outputs for the same binary input at a given state. It is very important to note that the pragmatic capacity is computed on the receiver side without the knowledge of the transmitted symbol sequence $\boldsymbol{A}$. Therefore, for SOQPSK-TG-PAM we only

consider possible mappings on the 4-state trellis used for SOQPSK-TG-PAM rather than the 512-state trellis that models the transmitted SOQPSK-TG signal. It follows that for both SOQPSK-MIL and SOQPSK-TG-PAM we consider all possible mappings on the SOQPSK precoder trellis. The precoder trellis is time-varying and alternates between odd-time and even-time trellis sections. Therefore, by only permuting the labels of the branches with the same starting state we get a total of $2^4 \times 2^4 = 256$ distinct mappings.

We computed the pragmatic capacity of SOQPSK-TG-PAM and SOQPSK-MIL for all distinct mappings, and observed that the pragmatic capacity is maximum for the natural mapping in both cases. Therefore, we conclude that the natural mapping of SOQPSK is optimal in the pragmatic capacity sense. The most common mapping of SOQPSK other than the natural mapping is the differentially encoded SOQPSK (DE-SOQPSK) [21]. We couldn't include plots for all distinct mappings, but we wanted show how differential encoding affects the pragmatic capacity of SOQPSK. In Figure 4.7 we see that the pragmatic capacity of SOQPSK-TG-PAM is significantly higher than that of differentially encoded SOQPSK-TG-PAM.

Chapter 5

Information Rates of SOQPSK Coding Schemes

5.1 Introduction

We have computed the capacity and the pragmatic capacity of SOQPSK in the previous chapters. In this chapter we compute the *information rates* of three SOQPSK coding schemes, i.e. a serially concatenated convolutional code (SCCC) SOQPSK-TG pragmatic scheme (SCCC-SOQPSK-TG), a low-density parity-check code (LDPC) SOQPSK-TG pragmatic scheme (LDPC-SOQPSK-TG), and a serially concatenated coded SOQPSK-TG-PAM (SC-SOQPSK-TG), for three coding rates: 1/2, 2/3, 4/5. Information rate of a coding scheme expressed in bits/s/Hz is the average mutual information between the information bits and the channel outputs scaled by $R_s \cdot R$, i.e. the product of the normalized symbol rate R_s and the code rate R, computed at the output of the channel decoder. SOQPSK-TG is used over SOQPSK-MIL in these coding schemes because of its significantly higher spectral efficiency.

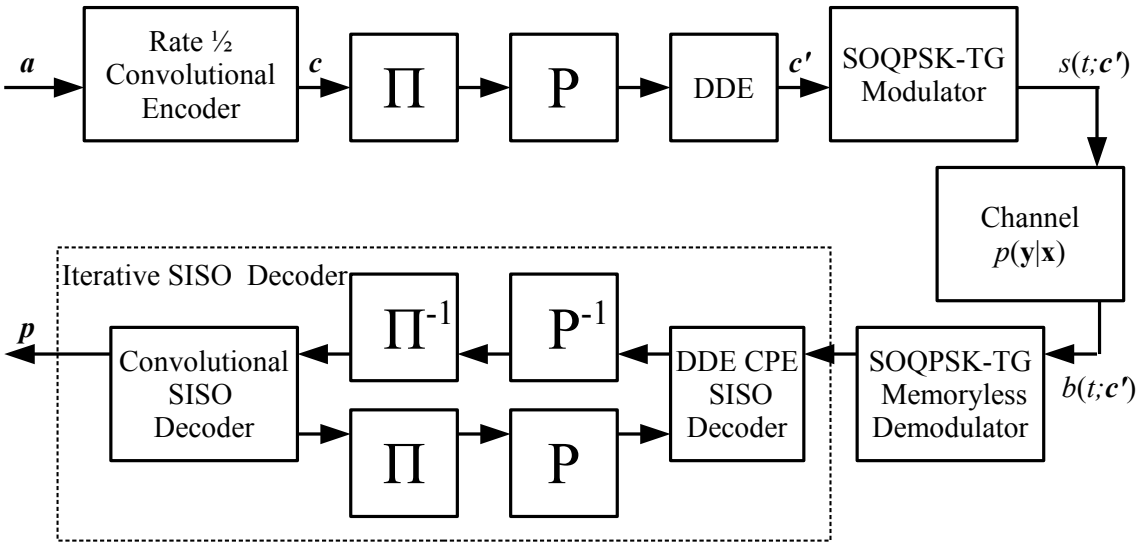

Figure 5.1: Block diagram of the serially concatenated coded SOQPSK-TG (SC-SOQPSK-TG)system.

5.2 SOQPSK-TG Coding Schemes

In this section we give brief system level descriptions of the SOQPSK-TG coding systems under consideration. All three systems take a binary information sequence $\boldsymbol{a} \triangleq \{a_i\}$ and output the a posteriori (AP) probabilities of the information bits denoted by $\boldsymbol{p} \triangleq \{p_i\}$ where $p_i = P(a_i = 1|b(t;\boldsymbol{c'}))$, $b(t;\boldsymbol{c'})$ is the AWGN channel output and $\boldsymbol{c'}$ is the modulated coded bit sequence. In addition, all three systems use PAM based reduced complexity SOQPSK-TG detection. The first coding system we describe is the serially concatenated coded SOQPSK-TG (SC-SOQPSK-TG) system shown in Figure 5.1. The information bit sequence $\boldsymbol{a}$ is encoded by the rate 1/2 convolutional encoder described in Chapter 2 (see Figure 2.6). The coded bit sequence $\boldsymbol{c}$ is first interleaved by a binary interleaver and then punctured by the blocks labeled with $\boldsymbol{\Pi}$ and $\mathbf{P}$ respectively in Figure 5.1. Puncturing is the operation of deleting a proportion of the coded bits to increase the code rate R. A puncturing rate of $R_{\mathrm{P}} = \frac{N_{\mathrm{O}}}{N_{\mathrm{I}}}$ implies keeping N_{O} bits for every N_{I} coded bits and deleting the rest. To achieve the coding rates R =1/2, 2/3 and 4/5 we use puncturing rates of $R_{\mathrm{P}} = 1$, 3/4, 5/8 respectively

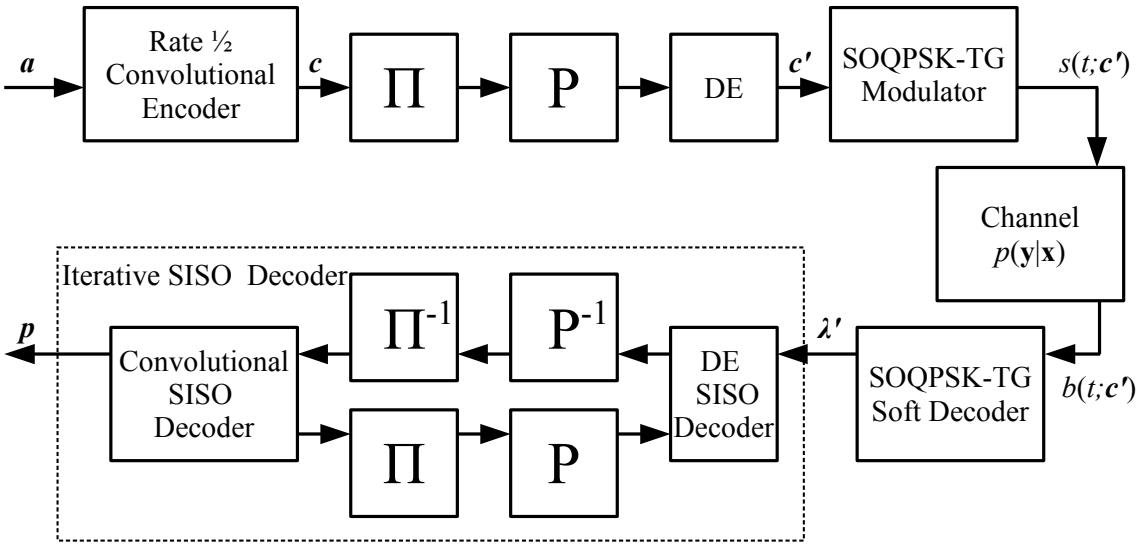

Figure 5.2: Block diagram of the serially concatenated convolutional code (SCCC) pragmatic SOQPSK-TG (SCCC-SOQPSK-TG) system.

with puncturing patterns given in [33]. The interleaved and punctured coded bit sequence $\boldsymbol{c'}$ is first double differentially encoded (DDE) by the rule (2.16) to introduce recursion to the code which is known improve the code performance [32]. Then it is modulated by an SOQPSK-TG modulator and sent over the AWGN channel. On the receiver side the channel output $b(t;\boldsymbol{c'})$ is used by an iterative soft-input soft-output (SISO [24]) decoder to produce the AP probability sequence $\boldsymbol{p}$.

The second SOQPSK-TG coding system we consider is a pragmatic coded SOQPSK-TG system with an SCCC encoder (SCCC-SOQPSK-TG) shown in Figure 5.2. Two constituent codes of the SCCC are the rate 1/2 code used in SC-SOQPSK-TG and a differential encoder (DE). The operation of this sytem is very similar to SC-SOQPSK-TG. The main difference is that the iterative soft-input soft-output (SISO [24]) decoder operates on the log likelihood ratios (LLR), $\boldsymbol{\lambda'}$, (see equation (4.1)) of the coded bit sequence $\boldsymbol{c'}$ generated by the soft SOQPSK-TG decoder, but no iterations take place between the SOQPSK-TG decoder and and the convolutional code SISO decoder.

The third SOQPSK-TG coding system is a pragmatic coded SOQPSK-TG system with

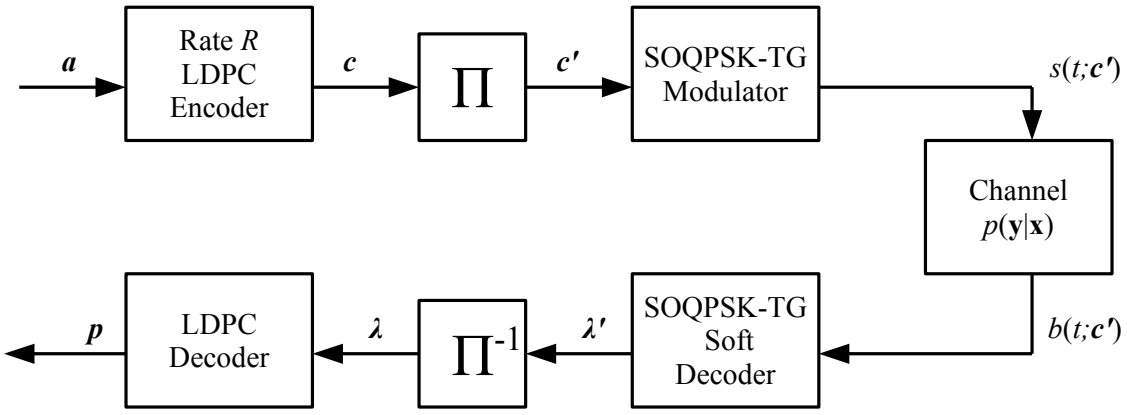

Figure 5.3: Block diagram of the low-density parity-check code (LDPC) pragmatic SOQPSK-TG (LDPC-SOQPSK-TG) system.

an LDPC code (LDPC-SOQPSK-TG) shown in Figure 5.3. Like the SCCC-SOQPSK-TG system the LDPC decoder operates on the LLRs outputted by the soft SOQPSK-TG decoder. We used the rate 1/2, 2/3 and 4/5 LDPC codes described in [34] that are known for their strong performance. On the receiver side we used the iterative decoding algorithm for binary LDPC codes given in [23] (Chapter 15 p. 648).

5.3 Simulation Results

In this section we compute the information rates of the coding systems described in the previous section. The information rate of a coding system [11] is given by

$$I \triangleq R \cdot R_s \lim_{N\to\infty} \frac{1}{N} \sum_{i=1}^{N} I(a_i; \boldsymbol{b}) \tag{5.1}$$

where $\boldsymbol{b}$ is the vector of channel outputs, i.e. a sufficient statistic for $b(t; \boldsymbol{c'})$, R_s is the normalized symbol rate of SOQPSK-TG described in Chapter 3, and the mutual information is computed at the output of the channel decoder. We note that the mutual information is given in bits/channel use, while the product $R \cdot R_s$ is the normalized symbol rate of the information bits given in channel use/s/Hz. When the term $\lim_{N\to\infty} \frac{1}{N} \sum_{i=1}^{N} I(a_i; \boldsymbol{b})$ is

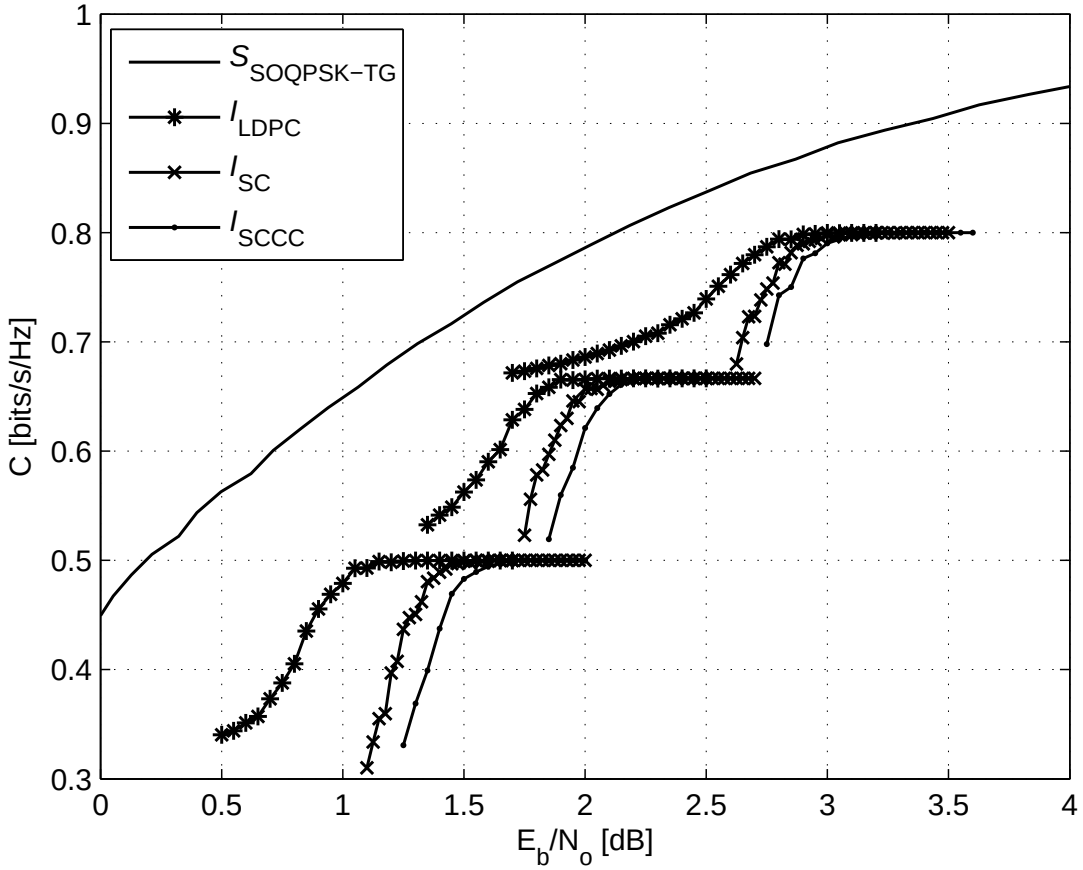

Figure 5.4: Information rates of SCCC-SOQPSK-TG, LDPC-SOQPSK-TG and SC-SOQPSK-TG for code rates of 1/2, 2/3, and 4/5 along with the spectral efficiency curve of SOQPSK-TG.

identically equal to 1, which implies perfect reliability, the information rate is equal to the code rate R since $R_s = 1.00$ for SOQPSK-TG. In other words, the equality $I = R$ implies that R bits of information can be reliably transmitted per second over 1 Hz, i.e. a spectral efficiency of R bits/s/Hz is achieved. Therefore, while we compute the information rates of coding schemes at various SNR per information bit E_b/N_0 values, we are interested in the minimum E_b/N_0 value required to have $I = R$. By noting that the conditional probability distribution $P(a_i|\boldsymbol{b})$ at the decoder output is a Bernoulli distribution with parameter p_i the

Table 5.1: Minimum required SNR values Eb/N_0 (dB) to achieve information rates, I_{LDPC}, I_{SC}, I_{SCCC}, equal to code rates, R, for SOQPSK-TG coding systems. Also given are the corresponding SNR values from the spectral efficiency curve of SOQPSK-TG.

R	$S_{\text{SOQPSK-TG}}$(dB)	I_{LDPC}(dB)	I_{SC}(dB)	I_{SCCC}(dB)
1/2	0.20	1.25	1.40	1.90
2/3	1.10	2.05	2.20	2.45
4/5	2.15	3.00	3.10	3.45

information rate expression can be further simplified to

$$I = R\bigg(1 + \lim_{N\to\infty} \frac{1}{N} \sum_{i=1}^{N} p_i \log_2 p_i + (1-p_i)\log_2(1-p_i)\bigg). \tag{5.2}$$

It follows that the information rate can be computed through simulations. During simulations all three systems use input blocks of 4096 bits for fairness purposes.

We computed the information rates I_{SCCC}, I_{LDPC} and I_{SC} of the coding systems SCCC-SOQPSK-TG, LDPC-SOQPSK-TG and SC-SOQPSK-TG respectively for code rates of 1/2, 2/3 and 4/5. The information rates are shown in Figure 5.4 where the spectral efficiency curve of SOQPSK-TG, labeled with $S_{\text{SOQPSK-TG}}$, serves as a reference. From Figure 5.4 we observe that the information rates of LDPC-SOQPSK-TG reach the code rates at lower SNR values than the other two coding schemes. However, it is difficult to locate the SNR values at which the information rates equal to the code rates are achieved. Hence in Table 5.1 we give these SNR values for all three schemes along with the SNR values from the spectral efficiency curve of SOQPSK-TG. In Table 5.1 we see that the LDPC scheme outperforms the other schemes while performing within 1.05 dB from the SOQPSK-TG spectral efficiency curve for all coding rates. We also observe that the coded SC-SOQPSK-TG systems outperform the pragmatic SCCC systems SCCC-SOQPSK-TG because of the iterations between the soft-output SOQPSK decoder and the convolutional SISO decoder.

Chapter 6

Conclusions and Future Work

We computed the channel capacities of SOQPSK-MIL and SOQPSK-TG over an AWGN channel through simulations by using the BCJR algorithm. The results showed that the capacity of SOQPSK-MIL is slightly higher than the capacity of SOQPSK-TG. We also computed the capacity of SOQPSK-TG constrained on a PAM based reduced complexity detection method (SOQPSK-TG-PAM). We showed that the capacity of SOQPSK-TG-PAM is very close to the capacity of SOQPSK-TG despite having much lower detection complexity. We then computed the spectral efficiency of SOQPSK-MIL and SOQPSK-TG-PAM. The results showed that SOQPSK-TG-PAM is spectrally much more efficient than SOQPSK-MIL despite having the same detection complexity, and with an SOQPSK-TG-PAM system information can be transmitted at rates up to 1 bit/s/Hz.

We then computed the pragmatic capacities of SOQPSK-MIL and SOQPSK-TG-PAM. The results showed that the pragmatic capacities of both SOQPSK-MIL and SOQPSK-TG-PAM are very close to their respective capacities. Therefore, the performance improvement achieved by using SOQPSK as a constituent code as opposed to a memoryless modulator operating with an independent binary encoder is very slim. We then showed that the natural mapping of SOQPSK between the input bits and the SOQPSK trellis branches maximizes the pragmatic capacity. Any other mapping such as differential encoding, reduces the pragmatic

capacity.

Finally we computed the information rates of three SOQPSK-TG coding schemes: a pragmatic coded SOQPSK-TG with an SCCC, a pragmatic coded SOQPSK-TG with an LDPC and a serially concatenated coded SOQPSK-TG. The results showed that the LDPC scheme, while superior to the other two schemes, performs within 1.05 of the spectral efficiency curve of SOQPSK-TG for code rates of 1/2, 2/3 and 4/5.

In [35] authors design an irregular repeat-accumulate code for minimum-shift keying (MSK), a type of binary CPM, that performs within 0.45 dB of the capacity curve of MSK. Our future work would be to follow the same approach with SOQPSK-TG, and design codes that approach the capacity curve of SOQPSK-TG further than the coding schemes we considered in this work.

Appendix A

Proof of Equation (2.21)

In the BCJR algorithm [13] the probability $P_n(k_n = k; I)$ is defined as the likelihood of the k-th codeword during the n-th symbol interval given by

$$P_n(k_n = k; I) = P(r_n|k_n) \tag{A.1}$$

where r_n is the channel observation. With SOQPSK the branch waveforms $\{s_k(t)\}$, $k \in \{0, 1, 2, \ldots, K-1\}$, are the codewords. We write the received SOQPSK signal $b(t; \boldsymbol{a})$ as a sequence of length-T_s observations, i.e. $b(t; \boldsymbol{a}) = (b_0(t), b_1(t), \ldots, b_n(t), \ldots)$. Now we have

$$P_n(k_n = k; I) = P(b_n(t)|s_k(t)) \tag{A.2}$$

which is the probability of observing $b_n(t)$ given that $s_k(t)$ is transmitted over an AWGN channel. In other words, we assume the relationship

$$b_n(t) = s_k(t) + n_n(t) \tag{A.3}$$

where $n_n(t)$ is the white Gaussian noise process $n(t)$ in the interval $nT_s \leq t < (n+1)T_s$. Now assume that we derived an orthonormal basis $\{\phi_j(t), 1 \leq j \leq V\}$ for the branch waveforms

$\{s_k(t)\}$. We denote the projection of the observation $b_n(t)$ onto the signal space $\{\phi_j(t)\}$ by the vector $\boldsymbol{b}_n$ and the projection of $s_k(t)$ by $\boldsymbol{s}_k$. The projection of $n_n(t)$ is denoted $\boldsymbol{n}_n$ where each element of $\boldsymbol{n}_n$ is a Gaussian random variable with variance $N_0/2$ from Karhunen-Loeve expansion [14]. Therefore the conditional probability $P(b_n(t)|s_k(t))$ can be expressed as

$$\begin{aligned} P(b_n(t)|s_k(t)) &= P(\boldsymbol{b}_n|\boldsymbol{s}_k) \\ &= \left(\frac{1}{\sqrt{\pi N_0}}\right)^V \exp\left(-\frac{||\boldsymbol{b}_n - \boldsymbol{s}_k||^2}{N_0}\right) \end{aligned} \tag{A.4}$$

which can be further simplified to

$$\begin{aligned} P(\boldsymbol{b}_n|\boldsymbol{s}_k) &= \left(\frac{1}{\sqrt{\pi N_0}}\right)^V \exp\left(-\frac{||\boldsymbol{b}_n||^2 + ||\boldsymbol{s}_k||^2 - 2\boldsymbol{b}_n \cdot \boldsymbol{s}_k}{N_0}\right) \\ &= \left(\frac{1}{\sqrt{\pi N_0}}\right)^V \exp\left(-\frac{||\boldsymbol{b}_n||^2 + ||\boldsymbol{s}_k||^2}{N_0}\right) \exp\left(\frac{2\boldsymbol{b}_n \cdot \boldsymbol{s}_k}{N_0}\right) \end{aligned} \tag{A.5}$$

where $\boldsymbol{b}_n \cdot \boldsymbol{s}_k$ denotes the *vector dot product* of $\boldsymbol{b}_n$ and $\boldsymbol{s}_k$. Since SOQPSK is a constant envelope signal the waveform energy $||\boldsymbol{s}_k||^2$ is constant for all k. In addition the term $||\boldsymbol{b}_n||^2$ doesn't depend on k. These two terms along with $\left(\frac{1}{\sqrt{\pi N_0}}\right)^V$ can be combined into an arbitrary constant C. Therefore, we have

$$P(\boldsymbol{b}_n|\boldsymbol{s}_k) = C \exp\left(\frac{2\boldsymbol{b}_n \cdot \boldsymbol{s}_k}{N_0}\right). \tag{A.6}$$

where

$$\boldsymbol{b}_n \cdot \boldsymbol{s}_k = \int_{nT_s}^{(n+1)T_s} b(t;\boldsymbol{a}) s_k^*(t) dt. \tag{A.7}$$

From (A.7) and (2.22) we have $\boldsymbol{b}_n \cdot \boldsymbol{s}_k = d_{k,n}$. It follows that

$$P_n(k_n = k; I) = C \exp\left(\frac{2d_{k,n}}{N_0}\right), \qquad 0 \le k \le K-1 \tag{A.8}$$

which completes the proof.

References

[1] J. B. Anderson, T. Aulin, and C.-E. Sundberg, *Digital Phase Modulation.* New York: Plenum Press, 1986.

[2] D. I. S. Agency, "Department of Defense interface standard, interoperability standard for single-access 5-kHz and 25-kHz UHF satellite communications channels," tech. Rep. MIL-STD-188-181B, Department of Defense, Mar. 1999.

[3] *IRIG Standard 106-04: Telemetry Standards*, Range Commanders Council Telemetry Group, Range Commanders Council, White Sands Missile Range, New Mexico, 2004, (Available on-line at http://www.ntia.doc.gov/osmhome/106.pdf).

[4] E. Perrins and M. Rice, "Reduced-complexity approach to iterative detection of SO-QPSK," *IEEE Trans. Commun.*, vol. 55, no. 7, pp. 1354–1362, Jul. 2007.

[5] P. A. Laurent, "Exact and approximate construction of digital phase modulations by superposition of amplitude modulated pulses (AMP)," *IEEE Trans. Commun.*, vol. 34, no. 2, pp. 150–160, Feb. 1986.

[6] E. Perrins and M. Rice, "PAM representation of ternary CPM," *IEEE Trans. Commun.*, vol. 56, no. 12, pp. 2020–2024, Dec. 2008.

[7] T. Aulin, C.-E. Sundberg, and A. Svensson, "Viterbi detectors with reduced complexity for partial response continuous phase modulation," in *Proc. National Telecommun. Conf., NTC'81*, New Orleans, LA, Nov./Dec. 1981, pp. A7.6.1–A7.6.7.

[8] A. Svensson, C.-E. Sundberg, and T. Aulin, "A class of reduced-complexity Viterbi detectors for partial response continuous phase modulation," *IEEE Trans. Commun.*, vol. 32, no. 10, pp. 1079–1087, Oct. 1984.

[9] D. Arnold, H.-A. Loeliger, P. Vontobel, A. Kavcic, and W. Zeng, "Simulation-based computation of information rates for channels with memory," *IEEE Trans. Inform. Theory*, vol. 52, no. 8, pp. 3498–3508, Aug. 2006.

[10] K. Padmanabhan, S. Ranganathan, S. P. Sundaravaradhan, and O. M. Collins, "General CPM and its capacity," in *Proc. Int. Symp. Inf. Theory*, Adelaide, Australia, Sep. 2005, pp. 750–754.

[11] A. Perotti, A. Tarable, S. Benedetto, and G. Montorsi, "Capacity-achieving CPM schemes," *IEEE Trans. Inf. Theory*, vol. 56, no. 4, pp. 1521–1541, Apr. 2010.

[12] G. Caire, G. Taricco, and E. Biglieri, "Bit-interleaved coded modulation," *IEEE Trans. Inf. Theory*, vol. 44, no. 3, pp. 927–946, May 1998.

[13] L. Bahl, J. Cocke, F. Jelinek, and J. Raviv, "Optimal decoding of linear codes for minimizing symbol error rate," *IEEE Trans. Inform. Theory*, vol. 20, no. 2, pp. 284–287, Mar. 1974.

[14] J. Proakis, *Digital Communications*. New York: McGraw-Hill, 2008.

[15] T. Hill, "An enhanced, constant envelope, interoperable shaped offset QPSK (SOQPSK) waveform for improved spectral efficiency," in *Proc. Int. Telemetering Conf.*, San Diego, CA, Oct. 2000.

[16] B. P. Lathi, *Linear Systems and Signal*. New York: Oxford, 2005.

[17] B. E. Rimoldi, "A decomposition approach to CPM," *IEEE Trans. Inform. Theory*, vol. 34, no. 3, pp. 260–270, Mar. 1988.

[18] M. Simon, *Bandwidth-Efficient Digital Modulation With Application to Deep-Space Communication.* New York: Wiley, 2003.

[19] L. Li and M. Simon, "Performance of coded OQPSK and MIL-STD SOQPSK with iterative decoding," *IEEE Trans. Commun.*, vol. 52, no. 11, pp. 1890–1900, Nov. 2004.

[20] E. Perrins and M. Rice, "Simple detectors for shaped-offset QPSK using the PAM decomposition," in *Proc. IEEE Global Telecommun. Conf.*, St. Louis, Missouri, Nov./Dec. 2005, pp. 408–412.

[21] M. K. Simon, "Multiple-bit differential detection of offset QPSK," *IEEE Trans. Commun.*, vol. 51, no. 6, pp. 1004–1011, Jun. 2003.

[22] E. Perrins, "Everything you wanted to know about double differential encoders but were afraid to ask," in *Proc. Int. Telemetering Conf.*, Oct. 2006.

[23] T. K. Moon, *Error Correction Coding: Mathematical Methods and Algorithms.* Wiley-Interscience, 2005.

[24] S. Benedetto, D. Divsalar, G. Montorsi, and F. Pollara, "A soft-input soft-output APP module for iterative decoding of concatenated codes," *IEEE Commun. Lett.*, vol. 1, no. 1, pp. 22–24, Jan. 1997.

[25] U. Mengali and M. Morelli, "Decomposition of M-ary CPM signals into PAM waveforms," *IEEE Trans. Inform. Theory*, vol. 41, no. 9, pp. 1265–1275, Sep. 1995.

[26] G. K. Kaleh, "Simple coherent receivers for partial response continuous phase modulation," *IEEE J. Sel. Areas Commun.*, vol. 7, no. 12, pp. 1427–1436, Dec. 1989.

[27] E. Perrins and M. Rice, "A new performance bound for PAM-based CPM detectors," *IEEE Trans. Commun.*, vol. 53, no. 10, pp. 1688–1696, Oct. 2005.

[28] T. M. Cover and J. A. Thomas, *Elements of Information Theory.* New Jersey: Wiley, 2006.

[29] X. Zhang and M. P. Fitz, "Symmetric information rate for continuous phase channel and BLAST architecture with CPM MIMO system," in *Proc. IEEE Int. Conf. Commun.*, vol. 5, May 2003, pp. 3051–3055.

[30] T. Nelson, E. Perrins, and M. Rice, "Near optimal common detection techniques for shaped offset qpsk and feher's qpsk," *Communications, IEEE Transactions on*, vol. 56, no. 5, pp. 724 –735, May 2008.

[31] J. G. Proakis and D. K. Manolakis, *Digital Signal Processing.* New Jersey: Prentice Hall, Apr. 2006.

[32] S. Benedetto, D. Divsalar, G. Montorsi, and F. Pollara, "Serial concatenation of interleaved codes: Performance analysis, design, and iterative decoding," *IEEE Trans. Inform. Theory*, vol. 44, no. 5, pp. 909–926, May 1998.

[33] Y. Yasuda, K. Kashiki, and Y. Hirata, "High-rate punctured convolutional codes for soft decision Viterbi decoding," *IEEE Trans. Commun.*, vol. 32, no. 3, pp. 315–319, Mar. 1984.

[34] Consultive Committee for Space Data Systems (CCSDS),, "Low density parity check codes for use in near-Earth and deep space applications (131.1-O-2 Orange Book)," Sep. 2007.

[35] M. Valenti, S. C. Cheng, and D. Torrieri, "Coherent and Multi-symbol Noncoherent CPFSK: Capacity and Code Design," in *Proc. IEEE Military Commun. Conf.*, Orlando, FL, Oct. 2007.