POLITECNICO DI MILANO
Corso di Laurea Magistrale in Ingegneria delle Telecomunicazioni
Dipartimento di Elettronica e Informazione

MODELING THE FILTERED AND SAMPLED CONTINUOUS-TIME SIGNAL AFFECTED BY WIENER PHASE NOISE

Supervisor: **Prof. Maurizio Magarini**
Co-Supervisor : **Prof. Dr. sc. techn. Gerhard Kramer**

Tesi di Laurea di:
Silvio Mandelli, matricola 765369

Anno Accademico 2011-2012

To my parents

Abstract

This work concerns the study of the modeling of the sampled signal after the receiver matched filter with transmission over a Wiener Phase noise channel.
In the literature regarding phase noise (for example in [BMS12, DMR00, AB11, MAV$^+$11]) is usually considered a symbol-time model for the sampled signal, whose discrete-time phase noise after the receive filtering is considered to be a Wiener process; this is done by assuming that one has slow phase variation in one time symbol.
The contribution of this thesis is the study of the difference between the symbol-time model used in the literature, which in this thesis work will be called Symbol-spaced Phase Noise Model (SPNM), and the model (called Complete Model) for the sampled signal after the receive filter obtained by considering the continuous-time signals and mathematically deriving the expression for the signal at the sampler. Limits of validity of the SPNM will be proved; moreover the features of the phase noise of the Complete Model which differ from the features of the SPNM will be pointed out.
This document is organized as follows: after an introductory chapter about the notations used, the symbol-time model (SPNM) is presented together with an application found in the literature [AB11] where carrier recovery with Pilot Aided Transmission is performed for a channel that as a Symbol-spaced Phase Noise Model. In Chapter 3 the continuous-time signal at the input of the receiver is considered; the continuous-time signal at the sampler, obtained after matched filtering of the received signal affected by Wiener Phase Noise, is then derived. A mismatch between the Complete Model obtained by sampling the continuous-time signal and the SPNM is found.
In Chapter 4 the power of this mismatch is studied with simulations, with a particular emphasis on the phase noise of the Completed Model: indeed it is observed that the receive filtering introduces memory in the phase process.
The results of Chapter 4 arise the need to extend to the Complete Model the derivations done in [BMS12] for the SPNM. Therefore in

Chapter 5 some changes and integrations to [BMS12] are proposed in order to compute the capacity bounds with a symbol-spaced model that is different from the SPNM; in particular it has a phase noise with innovation process with memory greater than one.

Sommario

Questo lavoro si occupa dello studio della modellazione del segnale campionato dopo il filtro adattato al ricevitore, nel caso in cui la trasmissione avviene su un canale soggetto a rumore di fase di Wiener.
Quando si studia il rumore di fase in letteratura (per esempio in [BMS12, DMR00, AB11, MAV$^+$11]) si considera spesso un modello a tempo di simbolo per il segnale campionato, il cui rumore di fase tempo-discreto dopo il filtraggio è considerato essere un processo di Wiener, a patto di ammettere variazioni di fase lente durante un tempo di simbolo.
Il contributo di questo lavoro di tesi è di studiare la differenza tra il modello usato in letteratura a tempo di simbolo, che in questo documento chiameremo Symbol-spaced Phase Noise Model (SPNM), e il modello (chiamato completo) del segnale campionato dopo il filtro in ricezione ottenuto considerando tutti i segnali tempo-continui al ricevitore e derivando matematicamente l'espressione del segnale al campionatore. Verranno proposti anche dei limiti sotto i quali il modello SPNM è valido e evidenziate le caratteristiche del rumore di fase del modello completo che si discostano dal SPNM.
Questo documento si articola come segue: dopo aver introdotto le notazioni, il modello a tempo di simbolo (SPNM) viene presentato nel capitolo 2 insieme a un esempio trovato in letteratura [AB11] in cui si effettua il recupero di portante con trasmissione di simboli pilota servendosi di questo modello. Nel capitolo 3 è considerato il segnale tempo-continuo all'ingresso del ricevitore; viene quindi illustrata la derivazione del segnale tempo-continuo al campionatore inserendo prima un rumore di fase di Wiener e poi filtrando il segnale per il filtro adattato. Si osserva quindi uno scostamento tra il modello completo ottenuto campionando questo segnale tempo-continuo e il SPNM.
Nel capitolo 4, grazie a simulazioni, si studia la potenza di questo scostamento e in particolare le caratteristiche del rumore di fase del modello completo: si può notare che il il processo di fase presenta memoria introdotta dal filtro in ricezione. Le considerazioni che scaturiscono dall'osservazione dei risultati nel capitolo 4 fanno sorgere la necessità di estendere al modello completo il lavoro svolto in [BMS12]

per il SPNM). Quindi nel capitolo 5 viene ripreso [BMS12], a cui vengono proposti i cambiamenti e le integrazioni necessarie per calcolare i limiti della capacità con un modello a tempo di simbolo che, diversamente da quanto proposto nell'articolo, ha rumore di fase con memoria maggiore di uno.

Contents

List of Figures

Acronyms

AWGN Additive White Gaussian Noise

BER Bit Error Rate

CM Current Mismatch

DTFN Discrete-Time Frequency Noise

DSTM Discrete Symbol-Time Model

Hp Hypothesis

HWHM Half-Width Half-Maximum

i.i.d. independent and identically distributed

ISI Inter-Symbol Interference

LB Lower Bound

MSE Mean Square Error

NPC Normalized Pearson Coefficient

NSR Noise-to-Signal Ratio

PN Phase Noise

PLL Phase-Locked Loop

PSD Power Spectral Density

QAM Quadrature Amplitude Modulation

QPSK Quadrature Phase-Shift Keying

SPNM Symbol-spaced Phase Noise Model

SPVC Slow Phase Variation Condition

Th Thesis

UB Upper Bound

USF Up-Sampling Factor

Chapter 1

Introduction on notations

In this chapter we report some notations that will be often encountered in this work. Additional notation may be introduced and explained within each chapter.

The imaginary unit is denoted with $j = \sqrt{-1}$. Given a complex number w, we denote its real and imaginary part as $\Re\{w\}$ and $\Im\{w\}$, respectively, and its argument with

$$\arg(w) = \angle(w) = \arctan \frac{\Im\{w\}}{\Re\{w\}}.$$

The complex conjugate of w is denoted with $w^\star$. The complex conjugate of the inverse of w is indicated with

$$w^{-\star} = \frac{1}{w^\star}.$$

Given a discrete random variable x we denote its expected value with $E\{x\}$.

A sequence is denoted either by $\{s_k\}$, where the subscript indicates that the sequence is indexed by k, or by a polynomial, i.e. the (bilateral) z-transform of the sequence

$$\mathcal{Z}\{s(k)\}(z) = S(z) = \sum_{k=-\infty}^{\infty} s(k)\ z^{-k}.$$

The auto-correlation of the sequence $\{s(k)\}$ is denoted with $\{\psi_s(k)\}$,

that has z-transform

$$\begin{aligned}\Psi_s(z) &= \sum_{k=-\infty}^{\infty} \psi_s(k)\ z^{-k} \\ &= \sum_{k=-\infty}^{\infty} z^{-k} \sum_{i=-\infty}^{\infty} s^\star(i) \cdot s(i+k) \\ &= S(z) \cdot S^\star(z^{-\star}),\end{aligned}$$

when

$$\sum_{i=-\infty}^{\infty} |s(i)|^2 < \infty.$$

If $\{s(k)\}$ is a stationary random sequence then the z-transform of the auto-correlation is

$$\Psi_s(z) = \sum_{k=-\infty}^{\infty} z^{-k} \sum_{i=-\infty}^{\infty} E\{s^\star(i) \cdot s(i+k)\}.$$

The inverse z-transform of $S(z)$ is denoted with

$$\mathcal{Z}^{-1}\{S(z)\}(k) = \{s(k)\}.$$

We denote with

$$[S(z)]_{-j}^{-i} = \sum_{k=i}^{j} s(k)\ z^{-k}$$

the sum of the monomials of $S(z)$ with degree between $-j$ and $-i$. If we want to consider the i-th element of a sequence $\{s_k\}$ we write s_i and the notation for $\{s_i, s_{i+1}, ..., s_{k-1}, s_k\}$ will be s_i^k.

Matrices and vectors are reported in boldface and, unless otherwise stated, a vector is assumed to be a column vector.

We denote that a variable X is Gaussian distributed with mean μ and variance σ^2 with

$$X \sim N(\mu, \sigma^2).$$

Chapter 2

Carrier Recovery in presence of Phase Noise

In this chapter we recall some basic concepts of an used model in digital communications with multiplicative noise: the Symbol-spaced Phase Noise Model (SPNM). We also propose some of the analysis considered in [AB11] for determining the optimal Carrier Recovery filter in order to estimate the phase after the receive filter and the sampler in a communication system, under the assumptions of the SPNM.

2.1 First Order Wiener Process

In the literature of digital communications, a trusted model to study the performance of carrier recovery is the SPNM [MAV+11]:

$$y_k = x_k e^{j\theta_k} + w_k, \tag{2.1}$$

where $\{x_k\}$ is the sequence of modulated symbols with unitary average power and $\{w_k\}$ is a zero mean complex additive white Gaussian noise (AWGN) sequence with variance $\sigma_w^2 = \frac{1}{SNR}$. The time-varying phase noise $\{\theta_k\}$ is usually modeled as a first order Wiener process

$$\theta_{k+1} = \theta_k + \sigma_{PN}\nu_k, \tag{2.2}$$

where σ_{PN} is a constant and $\{\nu_k\}$ is a sequence of real independent and identically distributed Gaussian random variable with zero mean and unitary variance. In many papers is underlined that the standard deviation of the innovation term σ_{PN} must not be too high or the SPNM could be not valid anymore [MD97]. For small σ_{PN} the spectrum of the continuous-time complex exponential $e^{j\theta(t)}$ affected by random

phase walk, from which is generated the sampled phase noise sequence $e^{j\theta_k}$, is the Lorentzian function [DMR00]

$$L_\theta(f) = \frac{4\sigma_{PN}^2 T}{\sigma_{PN}^4 + 16\pi^2 f^2 T^2}, \tag{2.3}$$

where T is the symbol interval. The Lorentzian function is parametrized by its Half-Width Half-Maximum(HWHM) frequency

$$f_{HWHM} = \frac{\sigma_{PN}^2}{4\pi T} \tag{2.4}$$

In this way the model defined in (2.1) and (2.2) describes the channel by only two parameters: SNR and the standard deviation σ_{PN} of the innovation in the phase noise process. In [MAV+11] some experiments have been done in order to validate this model for the phase in lightwave WDM transmissions. As is well known, the performance of coherent detection is susceptible to phase noise [MD97]. In the wireless context scenario, phase noise is introduced by local oscillators used for up-conversion and down-conversion. In the optical scenario laser's phase noise is usually described as a Wiener process [FV88], and the Wiener model has been recently proposed in [MAV+11] also for the phase noise accumulated during nonlinear propagation, at least for the cases studied in that paper. Recent papers [Tay09, PHN09, LCY+11] address the problem of coherent demodulation in the presence of Wiener phase noise. In [AN09] this model is used in order to compute the Bit Error Rate (BER) of different modulations in optical transmissions impaired by Laser Phase Noise.

2.2 Carrier Recovery

In Digital Communications the task of Carrier Recovery is to track the phase sequence $\{\theta_k\}$ of (2.1) and (2.2). This is achieved with different approaches: the most common are Blind techniques and Pilot-Aided techniques. Blind techniques try to estimate the carrier phase $\{\theta_k\}$ without knowing the transmitted symbols; they usually exploit the knowledge of the statistical properties of the signal, the transmitted constellation, or both. Pilot-Aided techniques use the informations used in the Blind ones, but also they insert in the random pattern of transmitted symbols some Pilot sequences of symbols known both at the transmitter and at the receiver. Then the receiver uses the information given by this knowledge to estimate the phase. These techniques reduces the information rate. but are more robust than

other techniques when the phase noise is high.
In [AB11] is proposed a way to recover the phase noise impairment of the SPNM with a Pilot-Aided Transmission. In the paper is shown how to recover the phase with the assumption of having independency between the transmitted data, the phase noise and the additive noise; let the pilot rate be M^{-1}, meaning that one pilot symbol is inserted after $M-1$ payload symbols periodically, so the transmitted Pilot Sequence, with unitary power, is

$$B(z) = \sum_k a(kM) z^{-kM} \tag{2.5}$$

Multiplying the input sequence y_k by the complex conjugate of the zero-padded pilot sequence, one obtains the zero-padded sequence represented by the z-transform

$$L(z) = \sum_k l(kM) z^{-kM} \tag{2.6}$$

$$l(kM) = e^{j\theta(kM)} + w'(kM) \tag{2.7}$$

where $w'(kM)$ is the complex AWGN that is statistically equivalent to $w(kM)$. The channel is then characterized by its *pilot signal-to-noise ratio*

$$SNR_p = (E\{|w'(k))|^2)\})^{-1} \, . \tag{2.8}$$

The argument of (2.6) is

$$\phi(kM) = arg\{l(kM)\} = \theta(kM) + n(kM) \tag{2.9}$$

where $n(kM)$ is the tangential part of the complex noise $w'(kM)$. At high-to-intermediate SNR_p is approximated as real AWGN with zero mean and power $E\{n^2(kM)\} = (2 \cdot SNR_p)^{-1}$. If we define the m-th estimate of the phase $\hat{\Theta}_m(z) = H_m(z) \cdot \Phi(z)$, where

$$H_m(z) = \sum_{k=\delta}^{\infty} h(kM+m) z^{-kM-m} \tag{2.10}$$

In this way, the input sequence $\{\phi(-\infty), ..., \phi(-M), \phi(0)\}$ is used to get the estimate of the phase $\hat{\theta}(\delta M + m)$. Note that, for $\delta < 0$, $H_m(z)$ is non-causal. The cost-function to be minimized in order to find out the formula for the filters $H_m(z)$ is the Mean Square Error

$$MSE_m = E\{(\theta(m) - \hat{\theta}(m))^2\} \tag{2.11}$$

Since the sequence $\{y(k)\}$ is cyclostationary with period M, the sequence $\{\hat{\theta}(m)\}$ is cyclostationary also with period M as well, so we

can restrict the range of m to $m = 1, 2, ..., M$. Now, translating the continuous-time approach of [BS50] to a discrete-time scenario, the minimization of the m-th MSE, one finds that the optimal m-th transfer function is

$$H_m(z) = \alpha^2(1 - P(z))[(1 - P^*(z^{-*}))\Psi_{\theta_m}(z)]_{-\infty}^{-\delta M - m} \tag{2.12}$$

where $\Psi_{\theta_m}(z)$ is the z-transform of the zero-padded autocorrelation ψ_θ

$$\Psi_{\theta_m}(z) = \sum_{k=-\infty}^{+\infty} \psi_\theta(kM + m)z^{-kM-m} \ . \tag{2.13}$$

The transfer function $1 - P(z)$ in (2.12) corresponds to the causal monic and minumum phase whitening filter obtained from the spectral factorization

$$(1 - P(z))\alpha^2(1 - P^*(z^{-*})) = \frac{1}{\Psi_\phi(z)} \tag{2.14}$$

where

$$\Psi_\phi(z) = \Psi_{\theta_M}(z) + (2 \cdot SNR_p)^{-1} \tag{2.15}$$

$$1 - P(z) = 1 - \sum_{k=1}^{\infty} p(kM)z^{-kM} \tag{2.16}$$

$$\alpha^2 = exp\left\{ \int_0^1 log(\frac{1}{\Psi_\theta(e^{j2\pi f})}) \ df \right\} \tag{2.17}$$

These formulae hold when the following requirements are satisfied

$$\int_0^1 \frac{1}{\Psi_\theta(e^{j2\pi f})} df < \infty \tag{2.18}$$

$$\int_0^1 log(\frac{1}{\Psi_\theta(e^{j2\pi f})}) df > -\infty \ . \tag{2.19}$$

The first condition implies finite power and the second guarantees causality and rules out spectra that have zeros over a set of points of non-null measure over the unit circle, such as bandlimited spectra. Then, in the cited paper [AB11], the authors perform the analysis by taking two different cases, the first with $\delta \to \infty$, that is the infinite length filter (*unconstrained*), and the second with $\delta = 0$, that is the causal filter (*constrained*). Using (2.15) in (2.14), one gets the *unconstrained filter* that is used in the case of feed-forward synchronization,

$$H_m(z) = \frac{\Psi_{\theta_m}(z)}{\Psi_{\theta_M}(z) + (2 \cdot SNR_p)^{-1}} \quad . \tag{2.20}$$

If for some reasons one needs to have a feed-back implementation of the estimation filter, δ must be greater or equal to zero. For the minimum delay scenario, that is $\delta = 0$, the sequence $\{y(-\infty), ..., y(-M), y(0)\}$ is used to determine $\{\hat{\theta}(1), \hat{\theta}(2), ..., \hat{\theta}(M)\}$. The sub-filter $H_M(z)$ is the one that originates the sub-sequence $\{\hat{\theta}(kM)$ from the past samples of the noisy input phase. This is the linear predictor found in [SM08] in the context of carrier recovery based on Phase-Locked Loop

$$H_M(z) = P(z) \ . \tag{2.21}$$

The last step to determine a closed form for the filter is to substitute $\Psi_\theta(z)$ with the z-transform of the autocorrelation function of the process θ. From (2.1) and (2.2), one can write that

$$\Psi_\theta(z) = \frac{\sigma^2_{PN}}{(1 - z^{-1})(1 - z)} \ . \tag{2.22}$$

Starting from this expression of $\Psi_\theta(z)$, the z-spectrum of the input sequence can be written as

$$\Psi_\phi(z) = \frac{M\sigma^2_{PN}}{(1 - z^{-M})(\ -z^M)} + \frac{1}{2 \cdot SNR_p} \ , \tag{2.23}$$

and the spectral factor $1 - P(z)$ as

$$1 - P(z) = \frac{1 - z^{-M}}{1 - z_\phi z^{-M}}, \tag{2.24}$$

where $z_\phi = \lambda - \sqrt{\lambda^2 - 1}$, and $\lambda = 1 + M\sigma^2_{PN} SNR_p > 1, \ \ |z_\phi| < 1$.
The last step is to substitute those formulas both in the *unconstrained* filter and the the *causal* one. In the case of $\delta \to \infty$ one gets the optimal *unconstrained* filter

$$H_m(z) = (1-z_\phi)\frac{z^{-[m]}}{1 - z_\phi z^{-M}} + (\frac{[m]}{M} + \frac{z_\phi}{1 - z_\phi})\frac{z^{M-[m]}(1 - z^{-M})(1 - z_\phi)^2}{(1 - z_\phi z^{-M})(1 - z_\phi z^M)} \ , \tag{2.25}$$

where $[m] = m(modM)$. For $z_\phi \to 1$, the (2.25) becomes after some rearrangements

$$H_m(z) \approx \frac{z^{M-[m]}(1 - z_\phi)^2}{(1 - z_\phi z^{-M})(1 - z_\phi z^M)} \ . \tag{2.26}$$

In the case of *causal* filtering, the optimal m-th subfilter with delay $\delta = 0$ is

$$H_m(z) = (1 - z_\phi)\frac{z^{-m}}{1 - z_\phi z^{-M}} \ . \tag{2.27}$$

2.3 Summary

In this chapter we have presented the Symbol-spaced Phase Noise Model. The derivation of the model is given in the first section. In the other section it is shown how to build a Carrier Recovery System in order to have an estimate of the Phase Noise impairment of the SPNM, exploiting the information given by the Pilot Symbols. It is worth emphasizing that all the mathematical consideration of the section 2.2 come from the assumption that the signal after the receive filter and the sampler is of the form (2.1).
As we saw, in the literature the SPNM is considered valid when the phase varies slowly. Problems on the validity of the model may arise when the phase has fast variations. So one can ask if the SPNM is still a usable model for digital communications when the the standard deviation of the innovation of the phase noise σ_{PN} is big. A more accurate analysis of the SPNM in these cases will be done in the next chapters of this document.

Chapter 3

Symbol-spaced Phase Noise Model: mathematical proof of the Model Mismatching

In this chapter we deepen in the Symbol-spaced Phase Noise Model and we investigate, with mathematical derivations, its validity for digital communications. The model of the received filtered and sampled signal of a continuous-time transmission with Wiener phase noise is first derived. Then the model obtained is compared with the SPNM. In the last subsection the two models are compared using a frequency-domain approach and simulations.

3.1 Derivation of the received sampled continuous-time signal

In digital communication systems the oscillators demodulates the signal from radio-frequency to a baseband signal before any receiver processing; therefore the phase noise impairment due to the noise in the oscillators affects the signal before the receive matched filter. Let the continuous-time signal at the input of the receive system be

$$r(t) = s(t) + w(t) \tag{3.1}$$

$$s(t) = \sum_k a_k h(t - KT) e^{j2\pi f_c t} \,, \tag{3.2}$$

where $\{a_k\}$ is the sequence of modulated symbols, $h(t)$ is the square-root Raised Cosine Transmit Filter with roll-off factor α with group delay $D_g = 0$, and w(t) is a zero mean complex additive white Gaussian noise (AWGN) sequence with variance σ_w^2. Those filters are shaped in

a way which satisfies the Nyquist Criterion [Pro01]:

$$\int_{-\infty}^{\infty} h(\tau - iT)h^*(\tau - iT)d\tau = 1 \ , \text{ and}$$
$$\int_{-\infty}^{\infty} h(\tau - kT)h^*(\tau - iT)d\tau = 0 \ , \ \ i \neq k \ . \tag{3.3}$$

Then the signal $r(t)$ is demodulated through the optical oscillator, that have a phase $\beta(t) = 2\pi f_c t - \phi(t)$; in this way it introduces a multiplicative phase noise $\phi(t)$ in the baseband signal

$$r_b(t) = \sum_k a_k h(t - KT)e^{j\phi(t)} + w(t)e^{j\phi(t)} \ , \tag{3.4}$$

where $\phi(t)$ is the continuous-time Wiener process

$$\phi(t) = \sigma \int_{-\infty}^{t} n(t) \ , \tag{3.5}$$

where, $n(t)$ is a zero mean and unitary power Gaussian process and σ a constant.
Without loss of generality we can substitute $w'(t) = w(t)e^{j\phi(t)}$, that has the same statistical properties of $w(t)$. The signal $r_b(t)$ is then filtered by the filter $h^*(-t)$, matched to the transmit filter, in order to obtain

$$y(t) = \sum_k a_k \int_{-\infty}^{\infty} h(\tau - kT)h^*(\tau - t)e^{j\phi(\tau)}d\tau + n(t) \ , \tag{3.6}$$

where

$$n(t) = w'(t) * h^*(-t) \ . \tag{3.7}$$

By sampling the signal at time instants $t = iT$ one obtains the sequence

$$y_i = \sum_k a_k \int_{-\infty}^{\infty} h(\tau - kT)h^*(\tau - iT)e^{j\phi(\tau)}d\tau + n(iT) \ . \tag{3.8}$$

Note that, for $h(t)$ satisfying the Nyquist criterion, $n(iT)$ is white and Gaussian with Power Spectral Density (PSD) N_0 and then, by isolating the term $i = k$, the (3.8) becomes

$$y_i = a_i \int_{-\infty}^{\infty} h(\tau - iT)h^*(\tau - iT)e^{j\phi(\tau)}d\tau$$
$$+ \sum_{k \neq i} a_k \int_{-\infty}^{\infty} h(\tau - kT)h^*(\tau - iT)e^{j\phi(\tau)}d\tau + n_i \ . \tag{3.9}$$

It is important to point out that when the phase noise is a constant, without loss of generality we can assume $\phi = 0$, obtaining the Nyquist Criterion:

$$\begin{aligned} y_i &= a_i \int_{-\infty}^{\infty} h(\tau - iT)h^*(\tau - iT)d\tau \\ &+ \sum_{k \neq i} a_k \int_{-\infty}^{\infty} h(\tau - kT)h^*(\tau - iT)d\tau + n_i \\ &= a_i + n_i \ . \end{aligned}$$

When the phase noise $\phi(t)$ is not a constant the integral in (3.8) does not satisfy the Nyquist criterion anymore: in particular

$$\begin{aligned} &\int_{-\infty}^{\infty} h(\tau - iT)h^*(\tau - iT)e^{j\phi(\tau)}d\tau = \rho_i e^{j(\phi(iT)+\chi_i)} \neq 1 \cdot e^{j\phi(iT)} \ , \text{ and} \\ &\int_{-\infty}^{\infty} h(\tau - kT)h^*(\tau - iT)e^{j\phi(\tau)}d\tau \neq 0 \ , \quad i \neq k \ . \end{aligned} \tag{3.10}$$

However when $\phi(t)$ has small variations within the span of the filters $h(t)$ it can be shown that

$$\rho_i = 1 + \epsilon_{\rho_i} \approx 1 \ , \text{ and } \ \chi_i \approx 0. \tag{3.11}$$

Substituting (3.11) in (3.10), and using it in (3.9), we obtain

$$\begin{aligned} y_i &= a_i e^{j\phi(iT)} \cdot (1 + \epsilon_{\rho_i})e^{j\chi_i} \\ &+ \sum_{k \neq i} a_k \int_{-\infty}^{\infty} h(\tau - kT)h^*(\tau - iT)e^{j\phi(\tau)}d\tau + n_i \ . \end{aligned} \tag{3.12}$$

If we define

$$\text{ISI}_i = \sum_{k \neq i} a_k \int_{-\infty}^{\infty} h(\tau - kT)h^*(\tau - iT)e^{j\phi(\tau)}d\tau \tag{3.13}$$

since ISI_i (Inter-Symbol Interference) is due to the symbols a_k, with $k \neq i$, we can finally write the Complete Model of the sampled and filtered received signal

$$y_i = a_i e^{j\phi(iT)} \cdot (1 + \epsilon_{\rho_i})e^{j\chi_i} + \text{ISI}_i + n_i \ . \tag{3.14}$$

Writing the filtered and sampled received sequence $\{y_i\}$ in this way, one can easily compare it with the Symbol-spaced Phase Noise Model. First of all, one can observe that the additive Gaussian noise sequence $\{n_i\}$ is the same of the SPNM. Considering the signal part, (3.14)

shows that an ISI term appears. Moreover it can be pointed out that there is a perturbation, both in the phase and in the magnitude of the transmitted symbol. Both this phenomena can be explained by observing that in the integrals in (3.10) there is a phase term ruled by the Wiener process $\phi(t)$, that is not constant in the integrals' domains $(-\infty, +\infty)$.
A mismatch between the SPNM and the sequence $\{y_i\}$, that is the Complete Model (3.14), after the sampler and the receive filter has been found out. In particular the SPNM seems to be an approximation of the Complete Model. In the next section we will focus on this mismatch.

3.2 Complete Model Features

This section consists in two subsections: first the difference between the SPNM and the filtered and sampled receive sequence $\{y_i\}$ of (3.14) impaired by Wiener Phase Noise is considered. Then a condition in order to have agreement between the sequence $\{y_i\}$ and the SPNM is derived. Then a demonstration of the Model Mismatch in the frequency domain is shown.

3.2.1 Difference between the SPNM and the filtered and sampled received signal

As it is shown in (3.14), the sequence $\{y_i\}$ of the Mismatched Model is different from the SPNM in (2.1). The aim of this subsection is to find a condition that makes the Complete Model be equivalent to the SPNM.
Previously we said that the perturbation of the current symbol a_i and the ISI term are due to the non-constant nature of the phase term $\phi(t)$. To be more precise one could say that, in order to remove this two impairments, $\phi(t)$ must be a constant when the product $h(\tau - kT)h^*(\tau - iT)$ assumes values not equal to zero or approximatively not close to zero.
Let us define the Slow Phase Variation Condition (SPVC): SPVC holds if the phase term $\phi(t)$ can be considered as a constant equal to $\phi(iT)$ when the product $h(\tau - kT)h^*(\tau - iT)$ has not negligible values. If this

condition holds, the equations in (3.10) can be rewritten as:

$$\begin{aligned}
&\int_{-\infty}^{\infty} h(\tau - iT)h^*(\tau - iT)e^{j\phi(\tau)}d\tau \\
&= e^{j\phi(iT)} \cdot \int_{-\infty}^{\infty} h(\tau - iT)h^*(\tau - iT) = e^{j\phi(iT)}, \text{ and} \\
&\int_{-\infty}^{\infty} h(\tau - kT)h^*(\tau - iT)e^{j\phi(iT)}d\tau \\
&= e^{j\phi(iT)} \cdot \int_{-\infty}^{\infty} h(\tau - kT)h^*(\tau - iT) = 0, \quad i \neq k.
\end{aligned} \tag{3.15}$$

Substituting (3.15) in (3.9), instead of the Mismatched Model, one obtains

$$y_i = a_i e^{j\phi(iT)} + n_i \ . \tag{3.16}$$

Since a continuous-time Wiener process $\phi(t)$ sampled at time instants $t = iT$, is a discrete-time Wiener process (see Appendix A) with innovation variance $\overline{\sigma}^2 = \sigma^2 T$, we can define the discrete-time sequence $\phi_i = \phi(iT)$, in order to obtain

$$y_i = a_i e^{j\phi_i} + n_i \ , \tag{3.17}$$

that is the SPNM, with $\theta_i = \phi_i$.
In this subsection the proof that the Symbol-spaced Phase Noise Model is an approximation of the Mismatched Model has been given. In Chapter 4, dedicated to simulations, we will try to measure some parameters of this mismatching.

3.2.2 Model Mismatch in the Frequency-Domain

The mismatch between the SPNM and the sequence $\{y_i\}$ in the (3.14), could be also observed from a frequency domain point of view.
As we have seen in this chapter, the AWGN noise does not have difference between the two models. Accordingly, in this subsection the additive noise $w(t)$ is not considered in order to have simpler mathematical formulae. Let us recall equation (3.4),

$$\bar{r}_b(t) = \sum_k a_k h(t - KT)e^{j\phi(t)} = s_b(t) \cdot e^{j\phi(t)} \ , \tag{3.18}$$

where $w(t) = 0$, and $s_b(t)$ is the baseband demodulated $s(t)$. If we take the Fourier Transform of $\bar{r}_b(t)$, under the hypothesis of independent and identically distributed a_i, we obtain

$$\bar{R}_b(f) = \sigma_a^2 H(f) * L_\phi(f) \ , \tag{3.19}$$

where $L_\phi(f)$ is the one from (2.3). The multiplication by the complex exponential phase noise in the time domain becomes a convolution in the frequency domain due to the duality of the Fourier Transform. When the phase noise process $\phi(t)$ is slow compared to the band of the signal, $\phi((i+1)T) - \phi(iT) \ll \pi/2$, the effect of this convolution is negligible since (3.19) becomes the convolution of $\sigma_a^2 H(f)$ with a function that is similar to an impulse in the frequency domain. If the bandwidth of the phase noise increases and becomes comparable with the bandwidth of the signal, the convolution produces a spectral broadening of the signal in respect to its original spectrum. To clarify the idea, here there are some figures as an anticipation of the Chapter 4 (dedicated to the simulations). Simulations on the spectra of the signals $s_b(t)$, $\phi(t)$ and $\bar{r}_b(t)$ are computed.

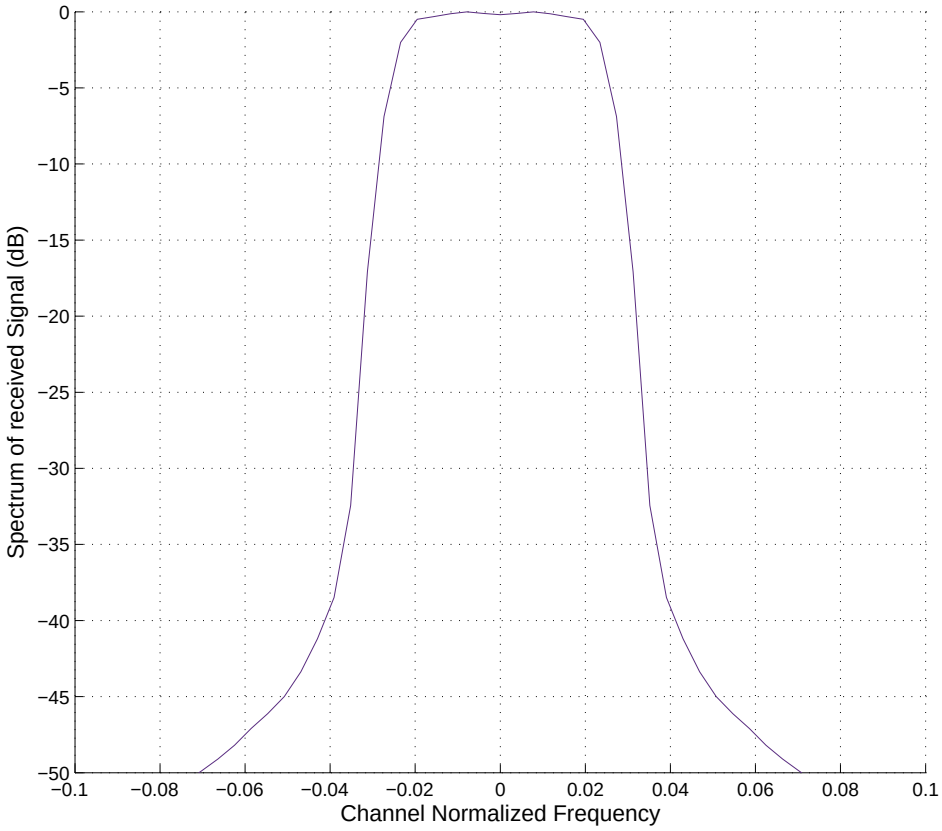

Figure 3.1: Simulated Spectrum of the Signal $s_b(t) = \sum_k a_k h(t - KT)$ at the input of the receiver

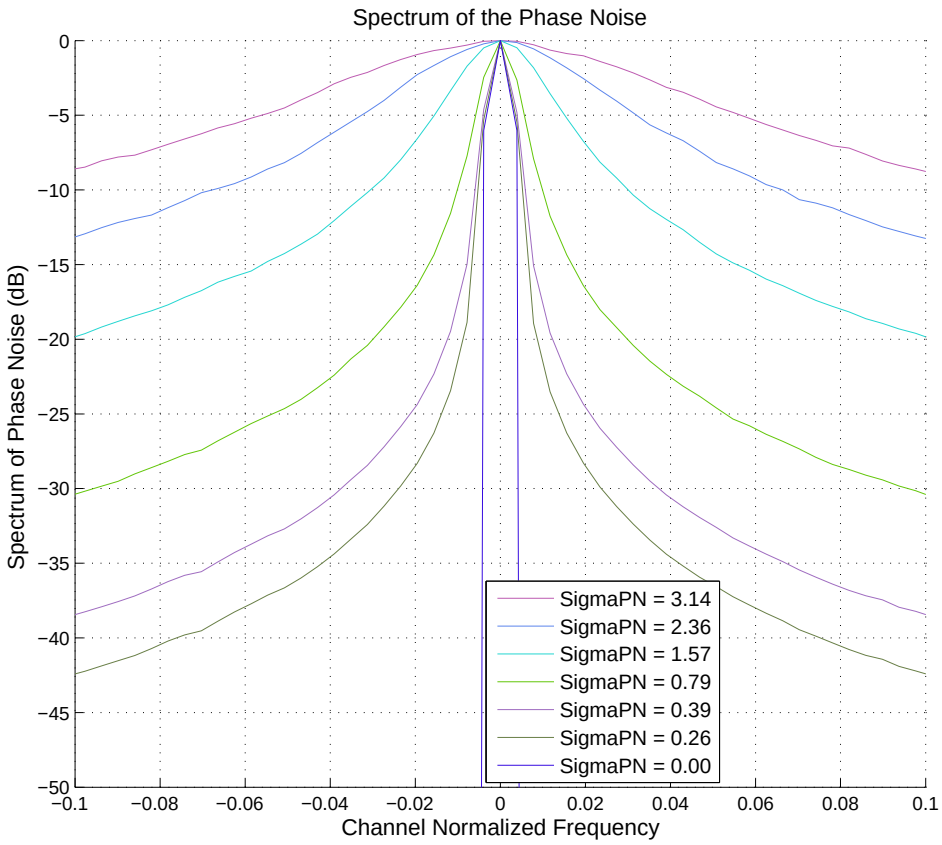

Figure 3.2: Simulated Spectrum of the complex exponential Phase Noise $e^{j\phi(t)}$ ruled by the Wiener Process $\phi(t)$ with different values for σ_{PN}

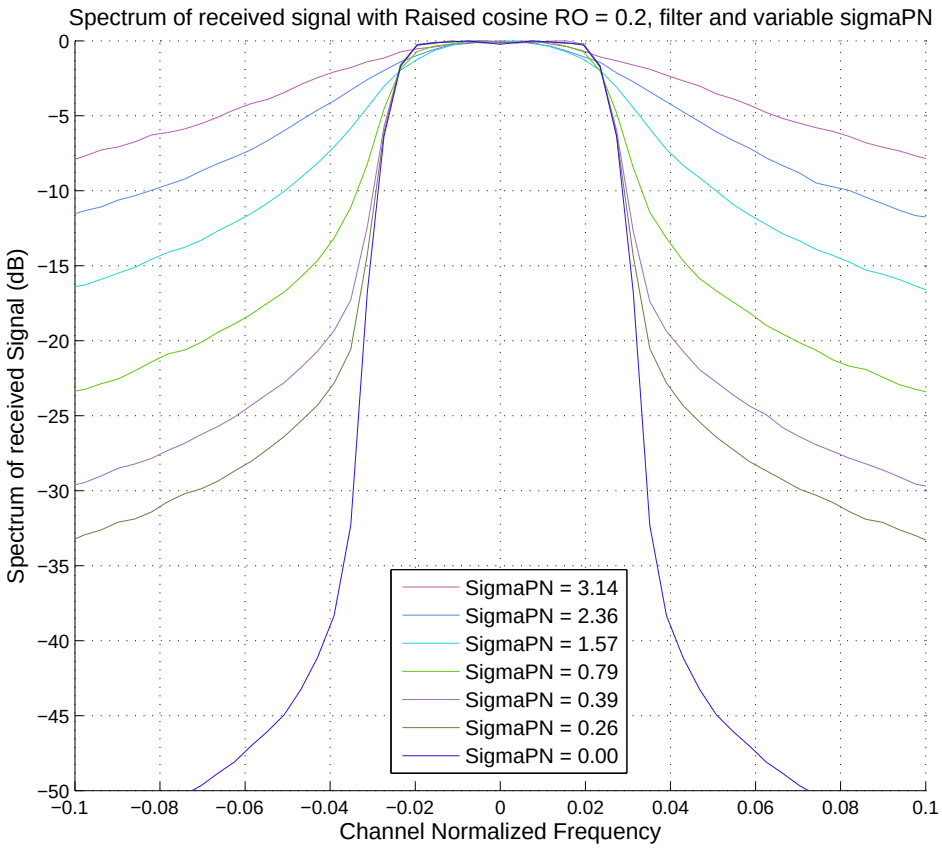

Figure 3.3: Simulated Spectrum of the received signal impaired by the Laser Phase Noise $\bar{r}_b(t) = \sum_k a_k h(t - KT)e^{j\phi(t)}$ compared with the spectrum of the receive matched filter $h^*(-t)$

In Figure 3.1 the spectrum of the received signal $|H(f)|^2$ without phase noise is reported. In Figure 3.2 is reported the spectrum of the complex exponential phase noise. It can be noted that the spectrum becomes wider with increasing values of σ_{PN}. Then, as one can observe from Figure 3.3, the received signal spectrum $\bar{R}_b(f)$ impaired by phase noise is not matched with the receive filter spectrum $|H(f)|^2$ obtained with $\sigma_{PN} = 0$. This leads to the same Model Mismatch of the Section 3.2.1, seen in a frequency domain approach: a filter at the receiver not matched with the transmitter violates the Nyquist Criterion, producing ISI; the random perturbation on the actual symbol a_i is due to the fact that the complex exponential Phase Noise is a random variable and not a deterministic process.

3.3 Summary

In this chapter we have shown a model mismatch between the Symbol-spaced Phase Noise Model and the received sampled sequence $\{y_i\}$ impaired by Wiener Phase Noise. This Mismatch could be proved both in the time and in the frequency domain. Moreover a condition for the Phase Noise process $\phi(t)$ is pointed out in order to obtain the SPNM from the Complete Model of the sampled and filtered signal (3.14).

Chapter 4

Simulations

In this chapter the simulation setup used in the thesis is given, together with the conclusions these simulations suggest. Since the aim of this thesis is not to investigate the impact of the AWGN, that has no mismatch in the two models, when it is not specified differently the additive noise is omitted from the simulations. In this way one can observe better the impairments due only to the phase noise. It is worth noting that there is no fear of strange phenomena of cross-talking of the noises, since we saw that in both models the AWGN remains additive with the same statistical properties. Considering the transmit and the receive filters $h(t)$ and $h^*(-t)$, both will be Square-root Raised Cosine Filters, unless otherwise specified.

The first section of this chapter explains how the continuous-time signals are handled in the simulations by up-sampling the signals in the channel. Since the Model Mismatch arises different issues, the remaining part of the chapter is divided into many sections: simulation of signal spectra, computation of the Mean Square Error (MSE) due to the Model Mismatch, pointing out the features of the phase noise impairment after the sampler. The sections dealing with simulation will be divided in three steps: aim and setup of the simulation, results and conclusions.

4.1 Up-Sampled Signal Model

As we saw in the last chapter, if we analyze the signals in the continuous-time domain, we discover that the SPNM is not always valid. However, to run simulations that allow to investigate this phenomena we cannot use continuous-time processes, since a calculator cannot process them. The trick one can do to avoid this problem is to process oversampled

signals at rate higher than the symbol frequency. Let's define the Up-Sampling Factor USF of our simulations as

$$USF = \frac{f_{sampling}}{f_{symbol}} \,, \tag{4.1}$$

where $f_{sampling}$ is the sampling frequency and $f_{symbol} = (T_{symbol})^{-1} = (T_s)^{-1}$ is the symbol rate. In this way one can sample the continuous time signal at the input of the receiver (3.4) to obtain

$$\begin{aligned} r_b\left(n\frac{T_s}{M}\right) &= \sum_k a_k h\left(n\frac{T_s}{M} - KT\right) \cdot e^{j\phi\left(n\frac{T_s}{M}\right)} + w'\left(n\frac{T_s}{M}\right) \\ r_{b,n} &= \sum_k a_k h_{n-kM} \cdot e^{j\phi_m} + w'_m \,, \end{aligned} \tag{4.2}$$

where the discrete-time processes of the second row in (4.2) is the continuous-time processes of the first row sampled at time $t = n\frac{T_s}{M}$. By setting the USF to values greater than 10 one begins to separate the replica in the spectrum enough to say that the continuous-time phenomenon is well simulated [PM92, AMR11]. Then, the remaining thing to do is to sample also the receive filter $h^*(-t)$ at $t = nT_s/M$, filter the received signal with the up-sampled filter and down-sample the obtained sequence; in this way one gets $y_i = y'_{nM}$, that is the same of (3.8).
Notice that if one wants to up-samples a discrete-time Wiener process $\{\theta_i\}$ with innovation variance σ^2_{PN} by a factor M, a discrete-time Wiener process $\{\theta^{UP}_n\}$ with innovation variance $\sigma^2_{UP} = \sigma^2_{PN}/M$ is obtained. A similar effect happens for the AWGN: if the white-sequence $\{n_i\}$ with variance σ^2_n is up-sampled by a factor M, the variance of the new sequence $\{n^{UP}_n\}$ must be $\sigma^2_{n^{UP}} = \sigma^2_n \cdot M$, in order to have the same values of the Power Spectral Density of the noise, since up-sampling a signal widens the frequency domain without aliasing.
In this section we have seen how continuous-time signals can be simulated in the next sections. When it is not specified, the Up-Sampling Factor of the simulation will be $M = 20$.

4.2 Spectral Broadening

In this section the simulation done to produce the results of the Subsection 3.2.2 are described. The aim of this section is to compute the spectrum of the signals $s_b(t)$ and $\bar{r}_b(t)$ of the (3.18). In order to compute the desired spectra, one must generate the transmitted constellation, the filters and set the parameters of the channel, like the variance of

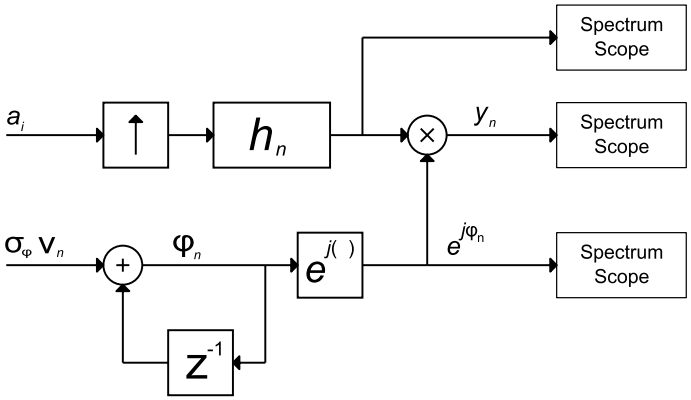

Figure 4.1: Block Diagram of the Spectral Broadening Analysis simulation

the Wiener Process.
Following the block diagram in Figure 4.1, in this set of simulations the parameters are

$$a_k \in \text{QPSK Modulation}, \quad E|a_k|^2 = 1, \tag{4.3}$$

$$h_n = \text{Square-root Raised Cosine Filter, roll-off } = 0.25, \tag{4.4}$$

$$\phi_n = \phi_{n-1} + \sigma_\phi \nu_n \ , \quad \nu_n \sim N(0,1), \tag{4.5}$$

$$\sigma_\phi^2 = \frac{\sigma_{PN}^2}{M} \ . \tag{4.6}$$

Observe that the choice of the modulation a_k does is not have relevant in our result if the sequence is i.i.d. since if we admit independence the spectrum of the sequence $\{a_k\}$ is white. σ_{PN} is the standard deviation of the innovation term of the phase noise at each symbol. Then the simulation is setup in with a Matlab script that follows the idea of the block diagram shown in Figure 4.1. The results and the comments of these simulation are in section 3.3.

4.3 Phase Noise after the Sampler

Recalling the Complete Model in (3.14), would be interesting to emphasize the difference between the received signal and the Symbol-spaced Phase Noise Model. With simulations it is easy to generate the sequence (3.14) without the additive noise. Now, if we ignore the introduced perturbation in the absolute value in respect to the SPNM, one can write that

$$y_i = a_i e^{j\theta_i} \ , \tag{4.7}$$

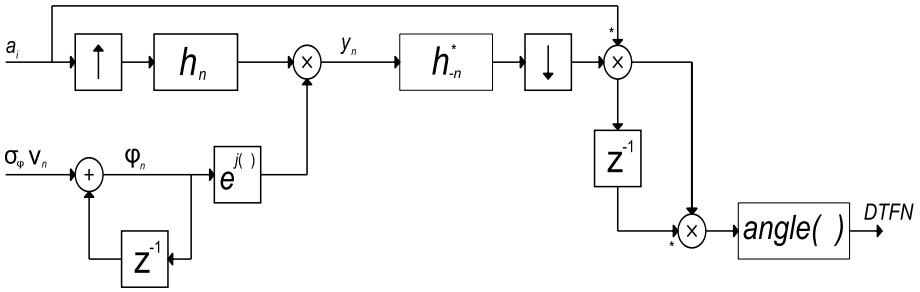

Figure 4.2: Block Diagram of the Phase Noise after the Sampler simulation

where $\{\theta_i\}$ is a new Phase Noise sequence that takes into account both the phase part of the ISI and $e^{j\chi_i}$ of (3.14). Now, if one can find some limits under which the sequence $\{\theta_i\}$ is a Wiener Process and the magnitude perturbation of the symbol is lower than the angle one, the SPNM can be validated. The aim of this section is to simulate $\{\theta_i\}$ and to point out when it can be considered a Wiener Process.

4.3.1 Simulation of the Frequency Noise Error

As we have seen above, the aim of this section is to understand if the Phase Noise after the Sampler is still a Wiener Process. First of all we must extract the sequence $\{\theta_i\}$ by simulations. In order to proof that the Phase Noise is a Wiener Process, one must proof that the sequence $\{\nu_i\} = \{\theta_i - \theta_{i-1}\}$ is White and Gaussian. Following the block diagram in Figure 4.2, one takes the received y_i and multiplies it by a_i^* in order to obtain $z_i = |a_i|^2 e^{j\theta_i}$ that does not depend of the angle of the transmitted data. Another step must be done in order to extract this "Discrete-Time Frequency Noise" (DTFN): if one takes z_{i-1}^* and multiply it by z_i, obtains $d_i = |a_i|^2 e^{j\nu_i}$, then if the angle of d_i is taken one have obtained ν_i. Intuitively this sequence has zero mean, but to be more precise in the following will be considered the zero mean sequence $\{\nu_i - E[\nu_i]\}$.
As we have seen before, in the next subsections we must find out if the sequence $\{\nu_i\}$ is White and Gaussian.

4.3.2 Proof of Gaussianity and Whiteness: setup

The index used to proof the Gaussianity of the DTFN is the Kurtosis of the sequence $\{\nu_i\}$

$$K_\nu = \frac{E[\nu_i^4]}{(E[\nu_i^2])^2} \ . \tag{4.8}$$

As it is known from statistics the Kurtosis of a Gaussian variable is 3, and the more that variable is distributed like a Gaussian function, closer to 3 is its Kurtosis.
The indicator considered to test the whiteness of the sequence is the Normalized Pearson Coefficient (NPC) at different time step p

$$\rho_p = \frac{Cov[\nu_i, \nu_{i-p}]}{Var[\nu_i]}. \tag{4.9}$$

The NPCs are the cross-correlation terms at time p, normalized by the variance. So we can say that the sequence is white if the NPCs are zero, or almost white if they are very close to zero.
The simulation parameters are set as follows:

$$a_k \in \text{constant, QPSK Modulation, 16-QAM, } \; E|a_k|^2 = 1, \tag{4.10}$$
$$h_n = \text{Square-root Raised Cosine Filter}, \alpha = \text{roll-off} \tag{4.11}$$
$$\alpha = (0, 0.1, 0.2, 0.3, 0.4, 0.5),$$
$$\phi_n = \phi_{n-1} + \sigma_\phi \nu_n \; , \;\; \nu_n \sim N(0,1), \tag{4.12}$$
$$\sigma_\phi^2 = \frac{\sigma_{PN}^2}{M} \; . \tag{4.13}$$

In these simulations we are using different modulation formats. This is because we guess that we will observe different behavior of the sequence $\{\theta_i\}$ with the choice of the constellation transmitted since that sequence depends also on an ISI term. Same thing is about the choice of different roll-off factors: since the tails of the filters decay faster if the roll-off is higher, the ISI term contribution will be lower at higher roll-off.

4.3.3 Proof of Gaussianity: results

Simulations are ran with different transmitted constellations: constant, QPSK, 16-QAM. The Kurtosis of the DTFN is plotted versus σ_{PN} in different figures for each constellation for different values of the roll-off parameter of the raised cosine filters.

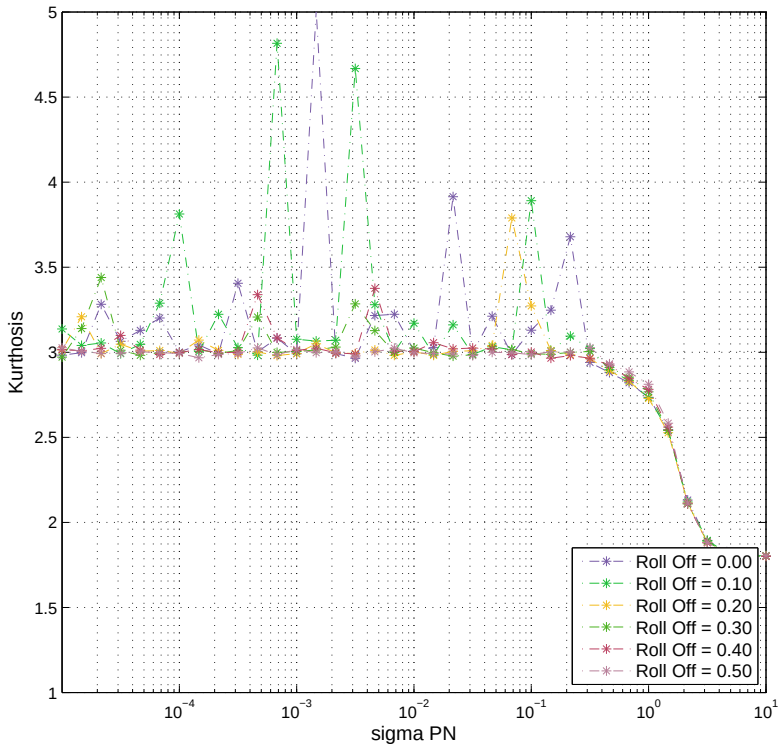

Figure 4.3: Kurtosis of the DTFN versus σ_{PN} with different roll-offs. Transmitted constellation: Constant

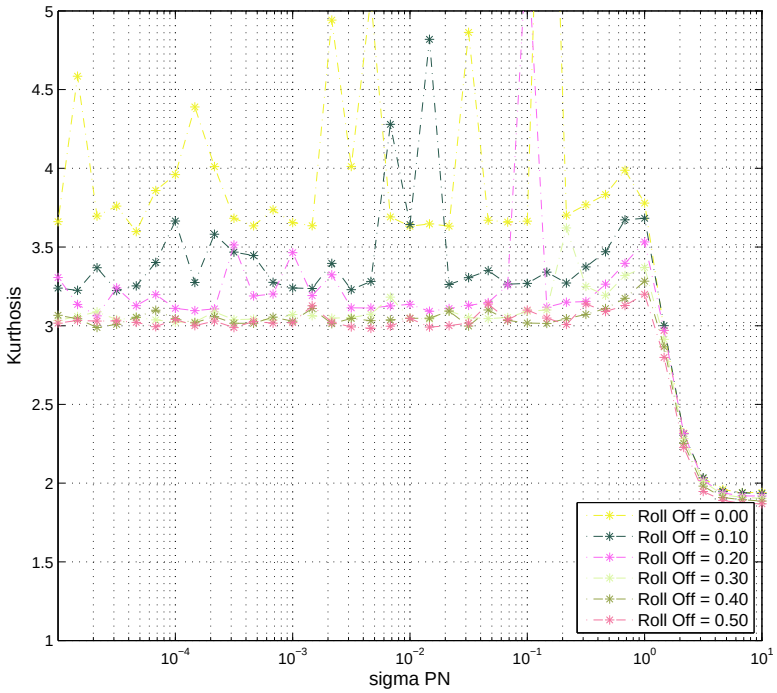

Figure 4.4: Kurtosis of the DTFN versus σ_{PN} with different roll-offs. Transmitted constellation: QPSK

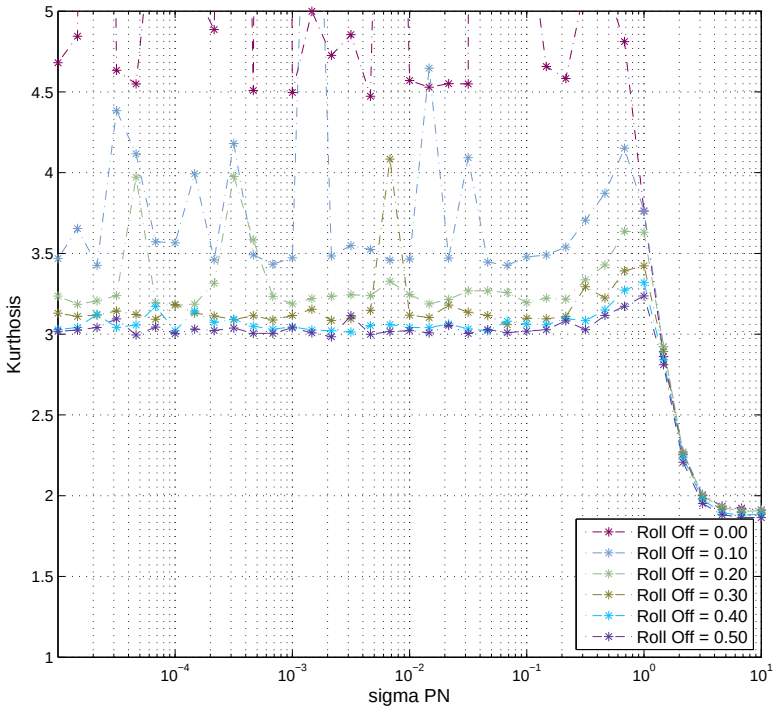

Figure 4.5: Kurtosis of the DTFN versus σ_{PN} with different roll-offs. Transmitted constellation: 16-QAM

We can observe in the Figures 4.3, 4.4 and 4.5 that the kurtosis of the DTFN is close to 3 in all simulations. In particular with roll-off $\alpha \geq 0.3$ the kurtosis is always close to 3 in all cases. Since the tails of a raised cosine filter are higher with lower roll-offs, one can note that if $\alpha < 0.3$ sometimes the kurtosis assumes value greater than 3 and this phenomenon is greater with phase and amplitude modulations (QPSK, QAM). One can also note that when σ_{PN} assumes values greater than a threshold the simulations become meaningless. This is because, when σ_{PN} is too high, we can have phase variations greater than π in one symbol-time and this introduces alias in the phase process since the phase is wrapped modulo 2π. From this simulations we can assume that this threshold is $\overline{\sigma}_{PN} \approx 0.3$.

4.3.4 Proof of Whiteness: results

In this subsection the 1-step and the 2-step NPC of the DTFN are plotted for different transmitted constellations versus the σ_{PN} for different values of the roll-off of the filters.

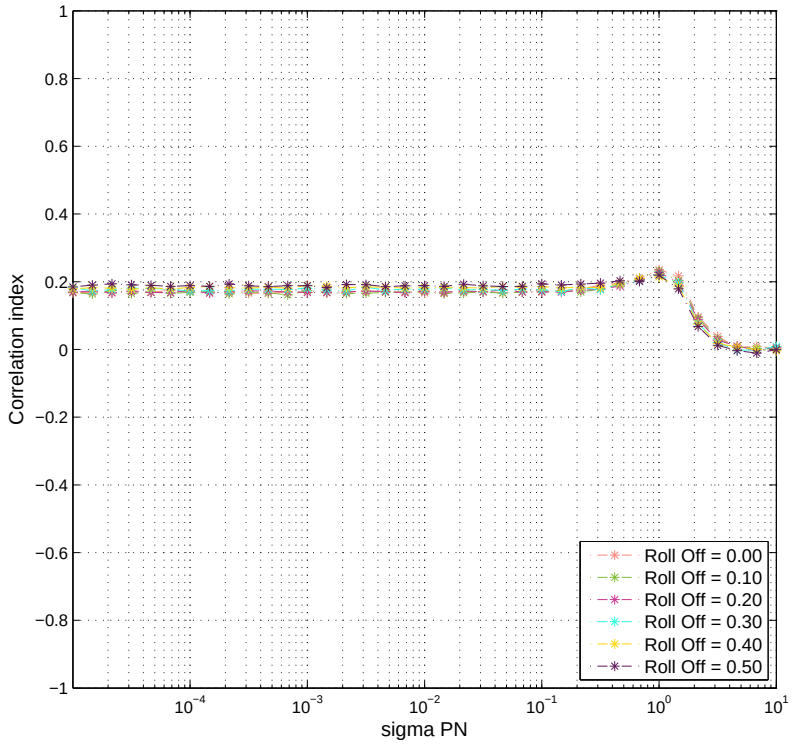

Figure 4.6: 1-step NPC of the DTFN versus σ_{PN} with different roll-offs. Transmitted constellation: Constant

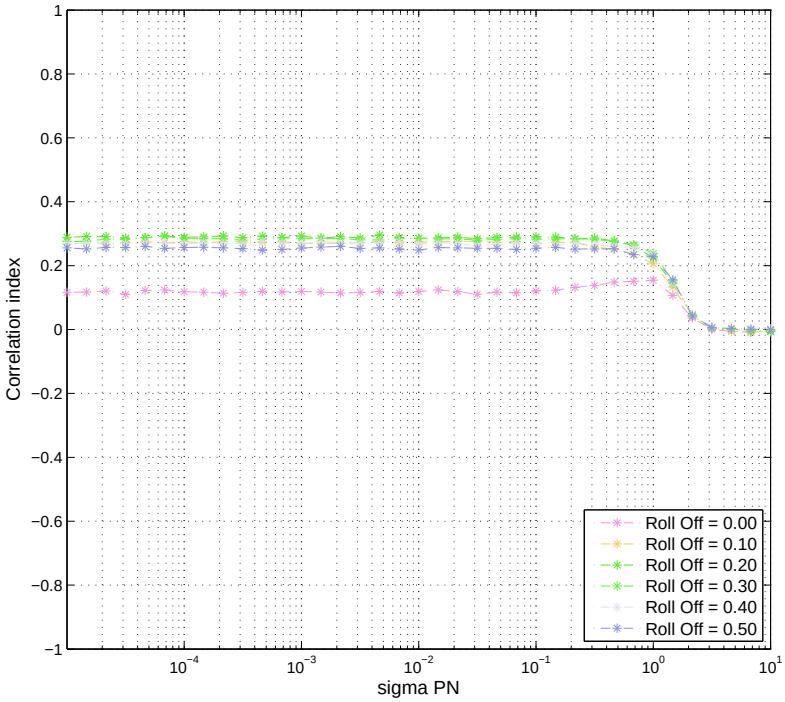

Figure 4.7: 1-step NPC of the DTFN versus σ_{PN} with different roll-offs. Transmitted constellation: QPSK

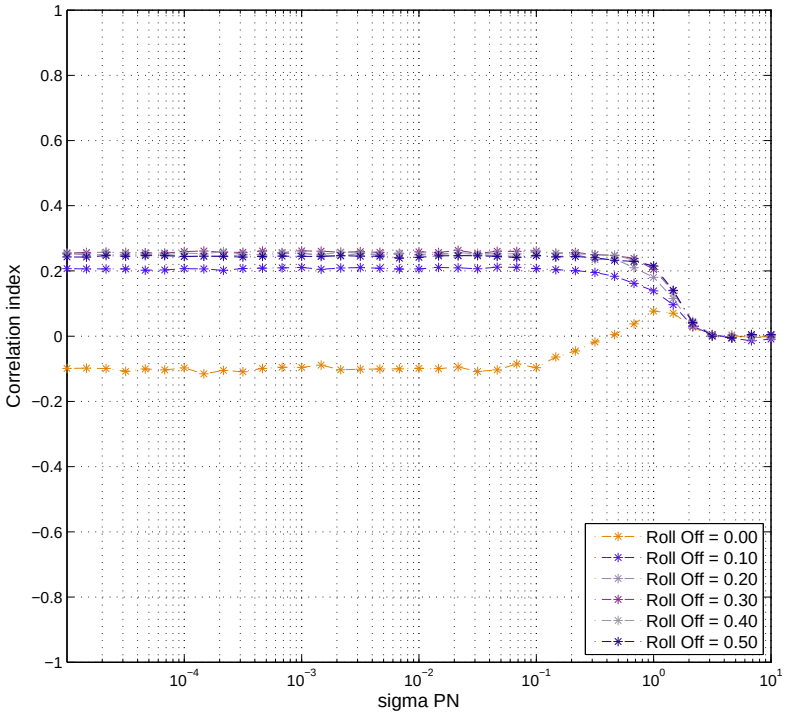

Figure 4.8: 1-step NPC of the DTFN versus σ_{PN} with different roll-offs. Transmitted constellation: 16-QAM

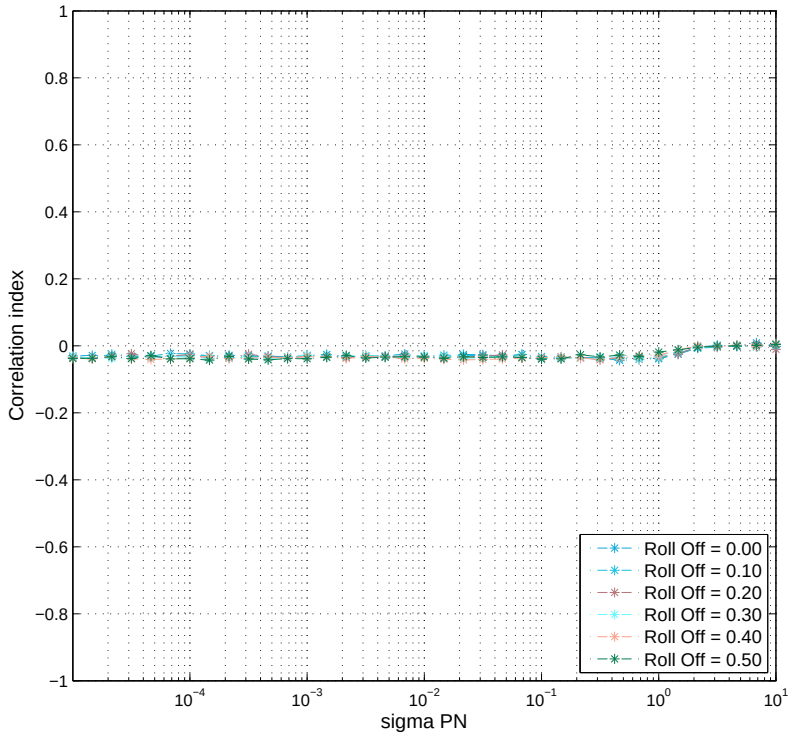

Figure 4.9: 2-step NPC of the DTFN versus σ_{PN} with different roll-offs. Transmitted constellation: Constant

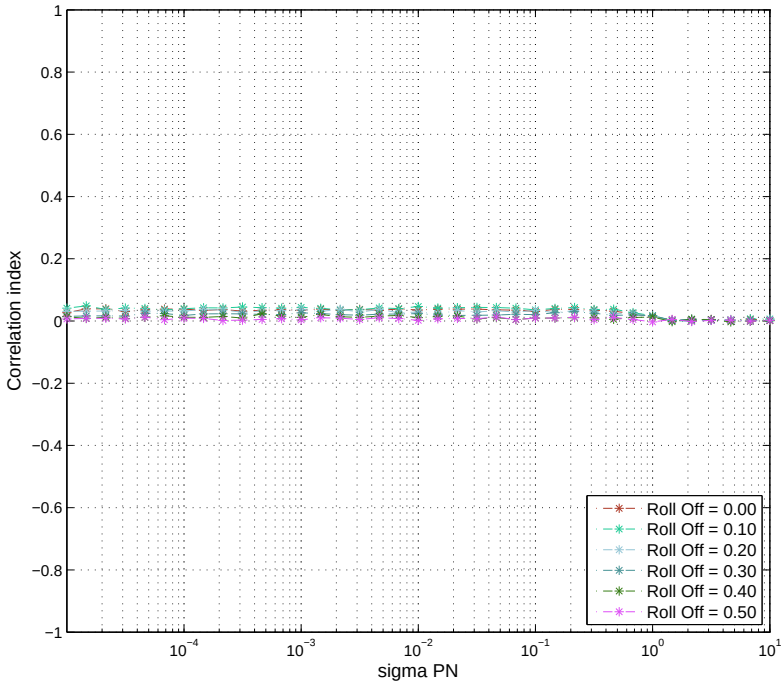

Figure 4.10: 2-step NPC of the DTFN versus σ_{PN} with different roll-offs. Transmitted constellation: QPSK

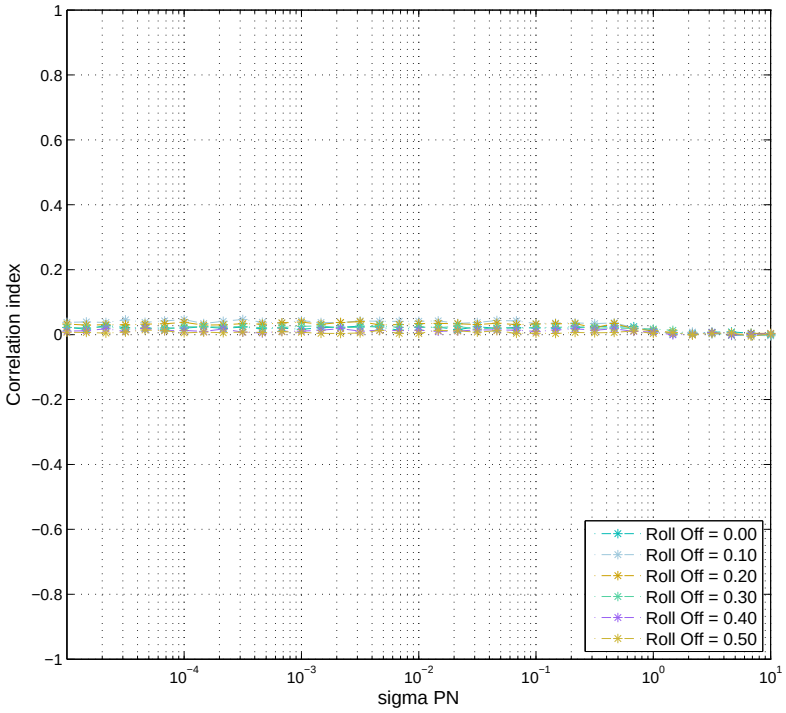

Figure 4.11: 2-step NPC of the DTFN versus σ_{PN} with different roll-offs. Transmitted constellation: 16-QAM

We can observe in Figures 4.6-4.8 that for all roll-offs α (with some differences with $\alpha = 0$), one have a $\text{NPC}_1 \approx 0.2$ with all values of σ_{PN}, before the saturation that occurs at $\overline{\sigma}_{PN} \approx 0.3$ as in the kurtosis simulations. In Figures 4.9-4.11 the 2-step NPC are considered. One can note that NPC_2 is almost to zero with all roll-offs and σ_{PN}. The figures with NPC_i, with $i > 2$, are not shown in this document, since $\text{NPC}_i \to 0, \ \forall i > 2$.

4.3.5 Proof of gaussianity and Whiteness: conclusions

In this section we have found out the behavior of the kurtosis and the Normalized Pearson Coefficient of the DTFN. Since the DTFN could be considered Gaussian in almost all cases, we can conclude that the main difference between a Wiener process and the phase θ_i of (4.7) is the 1-step NPC of the DTFN $\text{NPC}_1 \approx 0.2$. This is the main difference between the phase of the Complete Model and the SPNM: the phase $\{\theta_i\}$ is a third-order Wiener process, since its innovation process is not white, but has a 1-order causal by simmetry an 1-order noncausal memory. In Section 4.4 we will try to correct this phenomenon.

4.4 Simulation of the Power of the Mismatch

As we mentioned before, there is a Mismatch between the real model, called Mismatched Model, and the model used in the literature, the Symbol-spaced Phase Noise Model. In this section we try to estimate the power of this mismatch

4.4.1 Simulating the power of the mismatch

Let's define the sequence $\{\xi_i\}$ as

$$\xi_i = y_i e^{-j\phi_{iM}} - a_i \ , \tag{4.14}$$

where y_i is the simulated sampled sequence, ϕ_{iM} is the sampled Wiener Phase Noise in the channel, a_i the transmitted symbol. Note that the sampled phase ϕ_{iM} is an artifact done in simulations to perform a sort of carrier recovery on the received signal; however this carrier recovery could not be present in a real system since it doesn't know the received phase process.
Then one can compute the power of $\{\xi_i\}$ in order to obtain the Mean Square Error due to the model mismatch

$$MSE = E[|\xi_i|^2] \ . \tag{4.15}$$

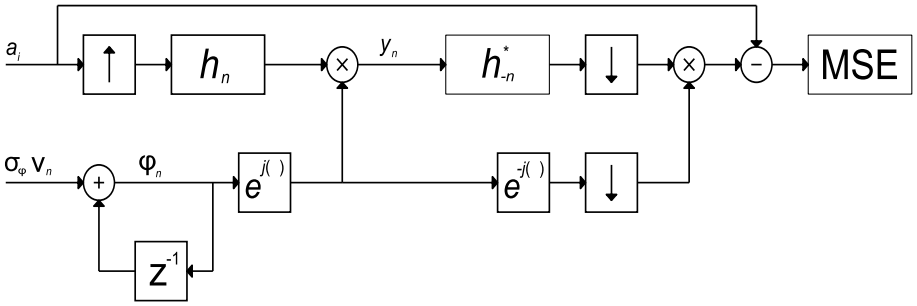

Figure 4.12: Block Diagram of the MSE due to the model mismatch simulation

The block diagram of the simulation is shown in Figure 4.12
The simulation parameters are set as follows:

$$a_k \in \text{constant, QPSK Modulation, 16-QAM,} \quad E|a_k|^2 = 1, \tag{4.16}$$

$$h_n = \text{Square-root Raised Cosine Filter}, \alpha = \text{roll-off} \tag{4.17}$$

$$\alpha = (0, 0.1, 0.2, 0.3, 0.4, 0.5),$$

$$\phi_n = \phi_{n-1} + \sigma_\phi \nu_n \ , \quad \nu_n \sim N(0,1), \tag{4.18}$$

$$\sigma^2_\phi = \frac{\sigma^2_{PN}}{M} \ . \tag{4.19}$$

As we have previously seen, we want to simulate with different inputs and roll-offs: the transmitted constellations are a Constant, a QPSK and a 16-QAM.

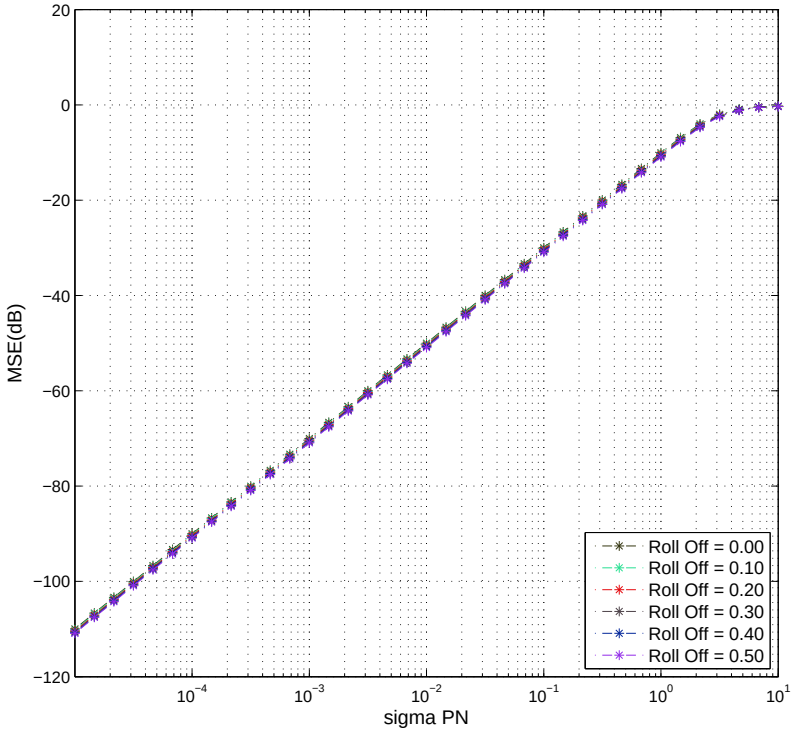

Figure 4.13: Mean Square Error of the model mismatch versus σ_{PN} with different roll-offs. Transmitted constellation: ConstantMSE

4.4.2 Simulations and results

In the Figures 4.13-4.15 the MSE of the model mismatch is plotted in a logarithmic scale versus a logarithmic scale of the σ_{PN} with different roll-off values in the 3 cases of transmitted data: constant, QPSK, 16-QAM.

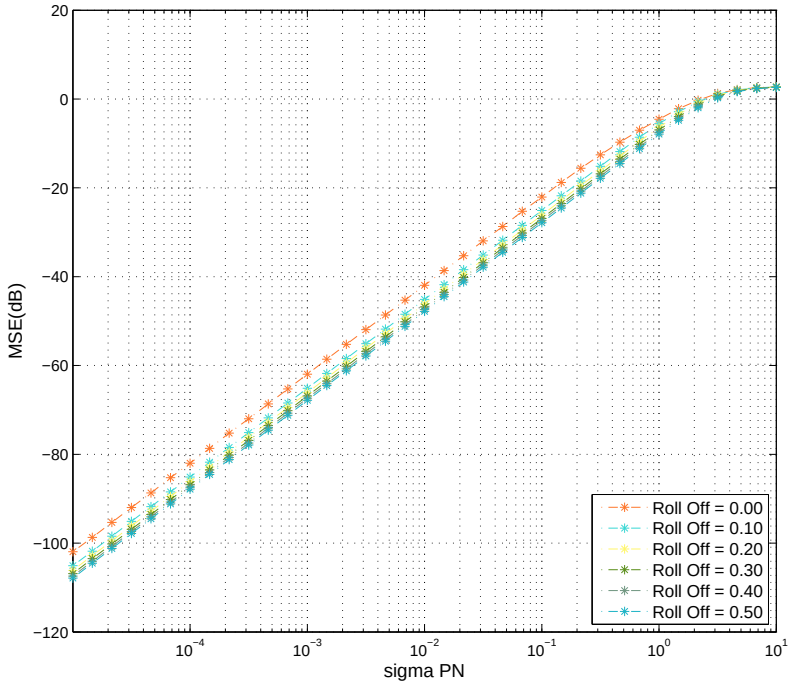

Figure 4.14: Mean Square Error of the model mismatch versus σ_{PN} with different roll-offs. Transmitted constellation: QPSK

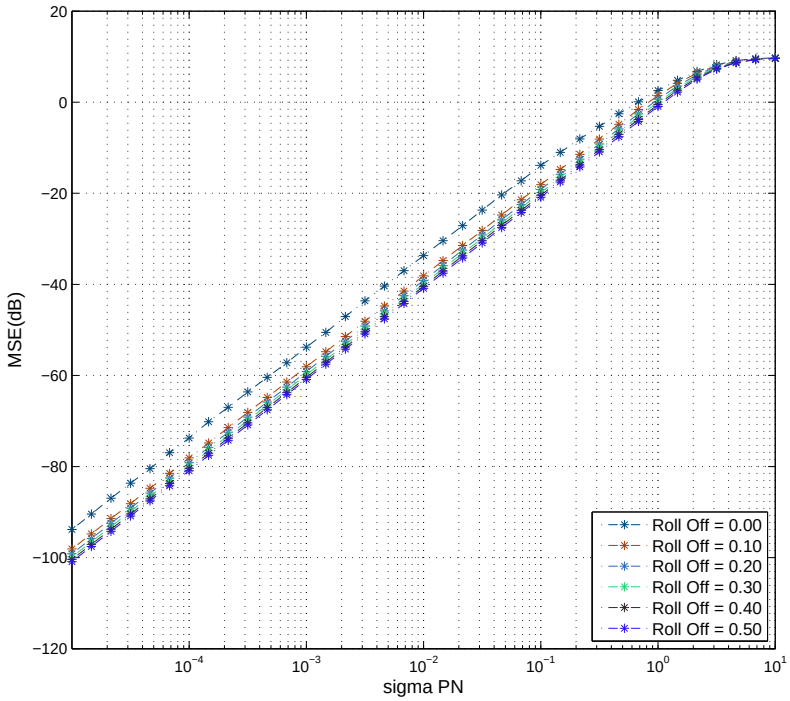

Figure 4.15: Mean Square Error of the model mismatch versus σ_{PN} with different roll-offs. Transmitted constellation: 16-QAM

One should immediately note that the power of the mismatch versus σ_{PN} has a 20 dB per decade slope. This emphasize a linear dependency between the MSE and σ_{PN}^2. This in the future work can lead to an analytical formula for the power of the mismatch. It is important to point out that the power of this mismatch is very weak: if we consider a QPSK transmission in a strong phase-noisy channel, for example $\sigma_{PN} = 0.1$, the power of the mismatch is MSE $\approx -20/25$ dB below the signal power. Usually a QPSK requires SNR = 5 dB in the worst cases [Pro01]. Even in this bad case, one has the power of the mismatch 15/20 dB below the AWGN noise, so the model mismatch could be neglected in most practical contexts and doing so one can use the SPNM to model the received and sampled signal in digital communications impaired by Wiener phase noise.

4.5 Measure the mismatch with a different Carrier Recovery

As it is said in the previous section, the phase noise process θ_i is not a Wiener process, but something similar that has the Normalized Pearson Coefficient at the 1-step equal to 1. In this section we try to use this information to demodulate the signal in (4.14) with a different phase process

4.5.1 Aim and setup

Considering the signal in (4.14) and, instead of considering ϕ_{iM} to demodulate the signal, one can use

$$\hat{\theta}_i = \measuredangle(z_1 e^{j\phi_{i-1}} + e^{j\phi_i} + z_1 e^{j\phi_{i+1}}) \,, \tag{4.20}$$

where z_1 is the NPC at the 1-time ρ_1 of the sequence $\{\theta_i\}$. In this way the MSE due to the model mismatch is computed in this way:

$$\xi'_i = y_i e^{-j\hat{\theta}_i} - a_i \,, \tag{4.21}$$

$$MSE = E[|\xi'_i|^2] \,. \tag{4.22}$$

In (4.20) we have introduced a way to filter phase sequences. This is the concept of planar filtering [AB11]. When the phase values does not change too quickly one can filter the complex exponential of the phase and then taking the angle instead of filtering the phases. Filtering phase processes, that have aliasing modulus 2π is not as simple as it seems. This concept will be deepened in Section 5.3 in the next chapter.

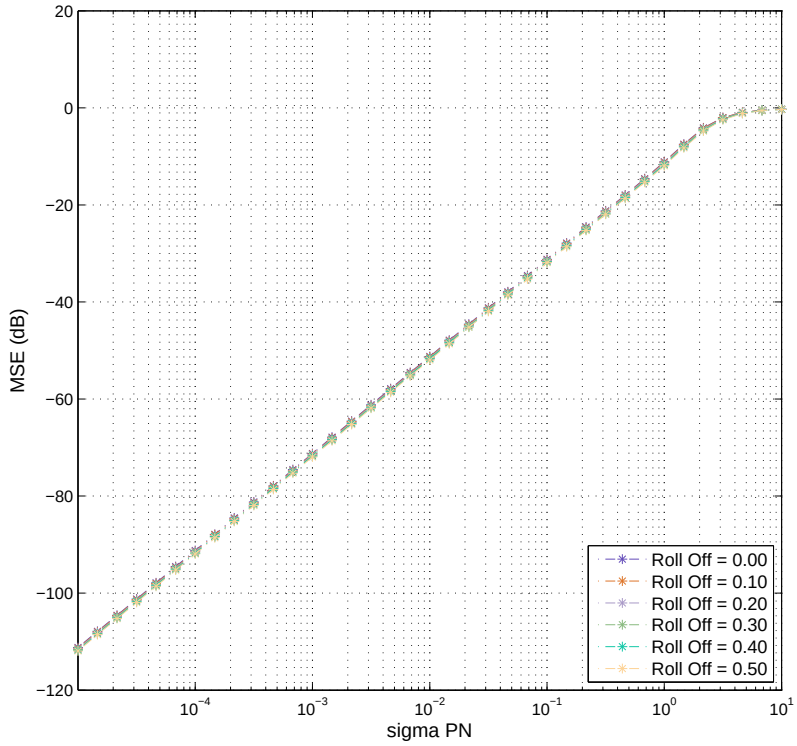

Figure 4.16: Mean Square Error of the model mismatch versus σ_{PN} with different roll-offs using the "Corrected Model". Transmitted constellation: Constant

4.5.2 Simulations and Results

The simulations are ran in the three previous cases of different transmitted data. In the thesis the figures and the discussion will be focused only onto the transmission of a constant; this is done because the results were the same for the QPSK and the QAM modulation. In Figure 4.16 it is plotted the MSE obtained by demodulating the signal with $\hat{\theta}_i$ of (4.20). The model in these simulations will is "Corrected Model" in this work. The model mismatch reduces its power by 1 dB. This is not a great improvement, but suggest that the idea of demodulating the signal a planar filtering on the phases is correct. The poor reduction is not a big issue since the power of the mismatch is still very weak compared with typical AWGN noise powers at which communication systems work.

4.6 Summary

In this chapter we have presented simulations in order to demonstrate that the Symbol-spaced Phase Noise Model is a good model for the filtered and sampled signal impaired by Wiener Phase noise under wide limits on the variance σ_{PN}^2 of the innovation in the Noise Process ϕ_i. In particular one must have that $\sigma_{PN} < \overline{\sigma}_{PN} \approx 0.3$. The power of the mismatch between the SPNM and the complete model is very weak, at least 15/20 dB lower than the required Noise-to-Signal Ratio (NSR) by real systems for big phase noise, for example $\sigma_{PN} = 0.1$. We pointed out also that the big difference between the Complete Model and the SPNM is the Normalized Pearson Coefficient at 1-step $\rho_1 \approx 0.2$. This memory can be taken into account in the Carrier Recovery System in the Section 2.2 and in the derivation of the Channel capacity of the system. This is the subject of the next chapter.

Chapter 5

Channel Capacity

In this chapter we will investigate some information theory topics related to the Symbol-spaced Phase Noise Model and this new "Complete Model" of the (3.14), in particular we focus only on the new phase noise impairment model, as in (4.7).

In the first section of the chapter we will compute the Upper Bound (UB) and the Lower Bound (LB) of the Constrained Channel Capacity of the SPNM, then a model in order to improve the computation in the case of the Mismatched Model is proposed. The last section will focus on the topic of filtering phase processes, since this is a complex problem when one has to take into account the wrapped nature of a quantized phase.

5.1 Channel Capacity Bounds of a Symbol-spaced Phase Noise Model Channel

In this section, following the approach of [BMS12] we find a way to compute the Upper Bound and the Lower Bound of the constrained channel capacity of the SPNM Channel. Let's recall the SPNM in (2.1),

$$\begin{aligned} y_k &= x_k e^{j\theta_k} + w_k, \\ \theta_{k+1} &= \theta_k + \sigma_{PN}\nu_k. \end{aligned}$$

where the vectors x_1^k and y_1^k the channel input and the channel output respectively, θ_1^k the sampled phase noise process and w_1^k the AWGN. By using the derivation in [BMS12], we can determine the upper and the lower bound on the capacity of a channel. Let us introduce a discrete-state channel with state s, instead of the continuous-state channel with

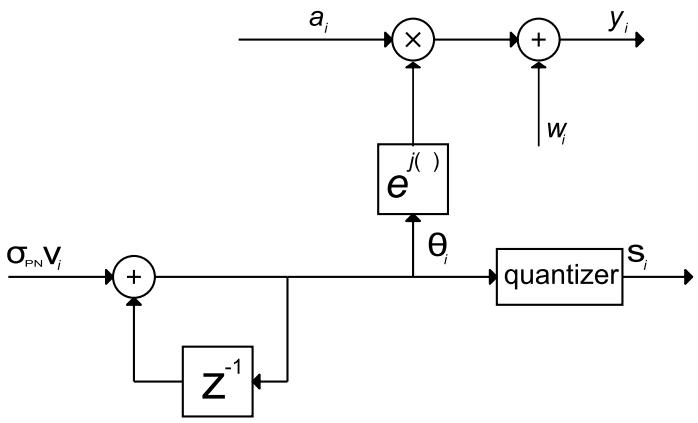

Figure 5.1: Block Diagram of the Capacity Bounds Simulation

θ, where s is the sampled version of θ in N bins; in this way the new "Auxiliary Channel" is an approximated version of the real channel with a discretized state (see Appendix B). This channel is easy to simulate and also it is usable in trellis-based recursions in order to compute the entropies that are needed for the computation of the capacity. The capacity bounds considered in [BMS12] are the following:

$$I(X,Y) \geq I_{LB}(X,Y) = \tilde{h}(Y) + H(X) - \tilde{h}(X,Y) \tag{5.1}$$

and

$$I(X,Y) \leq I_{UB}(X,Y) = \tilde{h}(Y) - h(Y|X,S) - H(S) + \tilde{H}(S|X,Y)\ , \tag{5.2}$$

where $\tilde{h}$ and $\tilde{H}$ are the entropies measured by the auxiliary channel simulations. One can write that if k is sufficiently large, the following approximation hold:

$$\tilde{h}(Y) = \lim_{k\to\infty} \frac{1}{k} \log_2 \prod_{i=1}^{k} p(y_i|y_1^{i-1}) = \lim_{k\to\infty} \frac{1}{k} \sum_{i=1}^{k} \log_2 p(y_i|y_1^{i-1}). \tag{5.3}$$

It is important to note the difference of p and P: the first is a probability density function, the second a Probability Mass Function. The same is for the differential entropy h and the entropy H of a random variable.
The simulation of the processes x_1^k, y_1^k and s_0^k follows the block diagram in Figure 5.1. Note that in the channel goes the real phase θ, but in the trellis computation the discrete state s is considered.

Considering a lightwave communication system modeled by a discrete-state Wiener process s we have that

$$P(s_0^k) = P(s_0) \prod_{i=1}^{k} P(s_i|s_{i-1}) \; ; \tag{5.4}$$

then we assume that the channel is memoryless given the input and the state, the input process and the state process are independent and the input process are independent random variables, so we can write

$$p(y_1^k|x_1^k, s_0^k) = \prod_{i=1}^{k} p(y_i|x_i, s_i) \tag{5.5}$$

and the channel probability law can be factorized in

$$p(y_1^k, s_0^k|x_1^k) = P(s_0^k)p(y_1^k|x_0^k, s_0^k) \; . \tag{5.6}$$

Substituting (5.4) and (5.5) in (5.6), one obtains

$$p(y_1^k, s_0^k|x_1^k) = P(s_0) \prod_{i=1}^{k} P(s_i|s_{i-1})p(y_i|x_i, s_i) \; . \tag{5.7}$$

This last equation shows that it is possible to compute the entropies $\tilde{h}(Y)$ and $\tilde{h}(X,Y)$ with the probabilities $p(y_i|y_1^{i-1})$ and $p(x_i, y_i|x_1^{i-1}, y_1^{i-1})$ in a recursive way in trellis with S states, where S is the number of the states of the discrete-state channel, as we saw in (5.3). The computation of $\tilde{H}(S|X,Y)$ with $P(s_i|s_{i-1}, x_i, y_i)$ derives easily from the trellis built to compute $p(x_1^k, y_1^k)$.
Let's focus on the computation of $p(y_i|y_1^{i-1})$, this is achieved as follows. Consider

$$P(s_i|y_1^i) = \frac{p(y_i|s_i)P(s_i|y_1^{i-1})}{p(y_i|y_1^{i-1})} = \frac{p(y_i|s_i)\sum_{s_{i-1}} P(s_i|s_{i-1})P(s_{i-1}|y_1^{i-1})}{p(y_i|y_1^{i-1})} =$$
$$= \frac{\sum_{x_i} p(y_i|x_i, s_i)P(x_i)\sum_{s_{i-1}} P(s_i|s_{i-1})P(s_{i-1}|y_1^{i-1})}{p(y_i|y_1^{i-1})} \; , \tag{5.8}$$

where $p(y_i|x_i, s_i)$ is pdf of the AWGN noise, $P(s_i|s_{i-1})$ is the discrete-state s innovation law and $P(s_{i-1}|y_1^{i-1})$ is the previous metric. This is the recursion we want to find; $p(y_i|y_1^{i-1})$ is a normalization factor in order to have $\sum_i P(s_i|y_1^i) = 1$ and is the probability we need in order to compute $\tilde{h}(Y)$ of the (5.3).
The same mathematics is behind the proof of

$$P(s_i|x_1^i, y_1^i) = \frac{p(y_i|x_i, s_i)P(x_i)\sum_{s_{i-1}} P(s_i|s_{i-1})P(s_{i-1}|x_1^{i-1}, y_1^{i-1})}{p(x_i, y_i|x_1^{i-1}, y_1^{i-1})} \; , \tag{5.9}$$

where we note the desired normalization factor $p(x_i, y_i|x_1^{i-1}, y_1^{i-1})$ in order to compute $\tilde{h}(X, Y)$. It is important to note that for the trellis (5.8) we must not substitute the real x_i since the variable is saturated, but in the (5.9) one must use the real transmitted symbol.
In the Upper Bound formula (5.2) there is the need to compute

$$\tilde{H}(S|X,Y) = -\lim_{k\to\infty} \frac{1}{k+1} \log_2 P(s_1^{k+1}|x_1^k, y_1^k). \tag{5.10}$$

If one writes

$$P(s_1^{k+1}|x_1^k, y_1^k) = P(s_{k+1}) \prod_{i=1}^{k} P(s_i|s_{i+1}, x_1^k, y_1^k) \ , \tag{5.11}$$

the $\tilde{H}(S|X,Y)$ can be computed with the same trellis used in order to compute $p(x_i, y_i|x_1^{i-1}, y_1^{i-1})$, with some manipulations:

$$P(s_1^{k+1}|x_1^k, y_1^k) = P(s_{k+1}) \prod_{i=1}^{k} P(s_i|s_{i+1}, x_1^i, y_1^i) \ , \tag{5.12}$$

where we used the (5.7); by Bayes rule,

$$P(s_i|s_{i+1}, x_1^i, y_1^i) = \frac{P(s_{i+1}|s_i, x_1^i, y_1^i) P(s_i|x_1^i, y_1^i)}{P(s_{i+1}|x_1^i, y_1^i)} = \frac{P(s_{i+1}|s_i) P(s_i|x_1^i, y_1^i)}{P(s_{i+1}|x_1^i, y_1^i)} \ , \tag{5.13}$$

where $P(s_{i+1}|x_1^i, y_1^i)$ is the metric of the trellis (5.9), $P(s_{i+1}|s_i)$ the state change probability and $P(s_{i+1}|x_1^i, y_1^i)$ a normalization factor. Note that the real quantized state s_i must be put in the formula in order to have that $P(s_i|s_{i+1}, x_1^i, y_1^i)$ is a number, and not a function of s_i.
In the simulations the parameters to choose are k and the number of the bins N in which the phase is quantized. Obviously one should expect that the UB and LB are closer the higher is the value N. In the Figure 5.2 the greater bounds are with $\sigma_{PN} = 0$, then the bounds with $\sigma_{PN} = 0.125$ are slightly below; however the bounds with $\sigma_{PN} = 0.5$ loses almost 25% of the capacity, even with high SNR values. It can be observed how the bounds converge very tightly, even in the case of $\sigma_{PN} = 0.5$.

5.2 Trellis change with a 3-order State

As we have seen in Chapter 4, the phase noise θ_k in the model

$$y_i = a_i e^{j\theta_i} + w_i$$

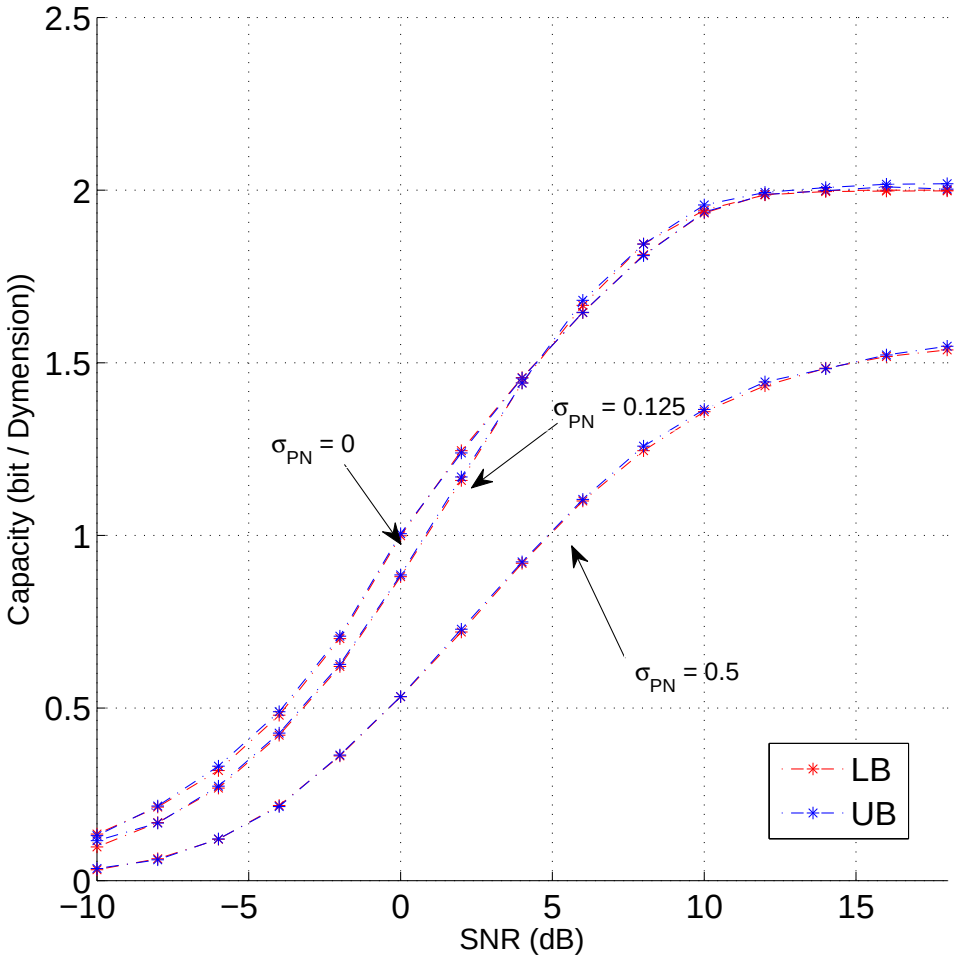

Figure 5.2: Capacity bounds of a SPNM versus the SNR with $\sigma_{PN} = \{0, 0.125, 0.5\}$

is not a First-Order Wiener Process, but a sequence that is described as a Wiener process ϕ_i, filtered by

$$P(z) = z_1 z^1 + 1 + z_1 z^{-1} \ . \tag{5.14}$$

Filtering a phase process is not as simple as may appear, so this topic will be explained in the next section. However one can immediatly notice that the Phase Process θ_i is not a First-Order Wiener Process anymore: this has one direct impact on our formulation in the Section 5.1: now (5.4) is not valid because we added memory on the phase beyond the first order. We need a device in order to find a state that represent the 3-memory channel and has a first-order memory. Then, if we define the vector of the state

$$\mathbf{s}_i = \begin{bmatrix} s_{i+1} \\ s_i \\ s_{i-1} \end{bmatrix} \ , \tag{5.15}$$

where s_i is the quantized phase at the time i, one can easily prove that $P(\mathbf{s}_i|\mathbf{s}_1^{i-1}) = P(\mathbf{s}_i|\mathbf{s}_{i-1})$. A state vector is defined: the computations in the trellis does not change, but if the state is multi-dimensional the problem of computation time arises: in the mono-dimensional state trellis for each symbol one has to compute the metric and sum over the states N times, here one has to do it N^3 times; this is computationally heavy, even if N is not very large (for example 64).
In order to reduce the complexity and the operations in the trellis, one can expand one trellis' sum over the state $\mathbf{s}_{i-1}$ in three different sums over the three variables of that state. If one does this from the (5.8), the trellis becomes

$$P(\mathbf{s}_i|y_1^i) = \frac{\sum_{x_i} p(y_i|x_i, \mathbf{s}_i) P(x_i) \sum_{\mathbf{s}_{i-1}} P(\mathbf{s}_i|\mathbf{s}_{i-1}) P(\mathbf{s}_{i-1}|y_1^{i-1})}{p(y_i|y_1^{i-1})} \ . \tag{5.16}$$

By expanding $P(\mathbf{s}_i|\mathbf{s}_{i-1})$, he can write that

$$P(\mathbf{s}_i|\mathbf{s}_{i-1}) = P(s_{i+1}, s_i, s_{i-1}|s_{i-2}, s_{i-1}, s_i) = P(s_{i+1}|s_i) \ , \tag{5.17}$$

that is the state change probability of the First-Order Process s_i. In this way, the $\sum_{\mathbf{s}_{i-1}}$ in the (5.16) can be decomposed into three different summations, and using the (5.17), one obtains

$$P(\mathbf{s}_i|y_1^i) = \frac{\sum_{x_i} p(y_i|x_i, \mathbf{s}_i) P(x_i) \sum_{s_i} P(s_{i+1}|s_i) \sum_{s_{i-2}} \sum_{s_{i-1}} P(\mathbf{s}_{i-1}|y_1^{i-1})}{p(y_i|y_1^{i-1})}$$

$$P(\mathbf{s}_i|y_1^i) = \frac{\sum_{x_i} p(y_i|x_i, \mathbf{s}_i) P(x_i) \sum_{s_i} P(s_{i+1}|s_i) P(s_i|y_1^{i-1})}{p(y_i|y_1^{i-1})} \ , \tag{5.18}$$

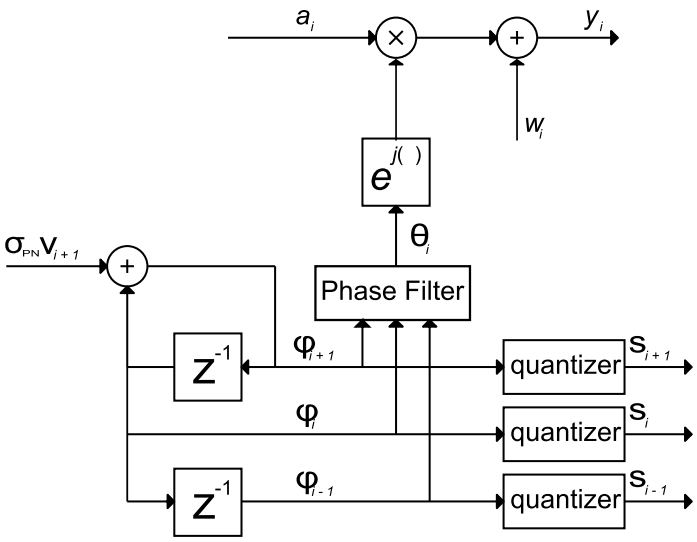

Figure 5.3: Block Diagram of the Capacity Bounds Simulation with a 3-Order Phase State

where $P(s_i|y_1^{i-1}) = \sum_{s_{i-2}} \sum_{s_{i-1}} P(\mathbf{s}_{i-1}|y_1^{i-1})$. Considering the equation (5.13) that permits to compute $\tilde{H}(S|X,Y)$ and the (5.17), the (5.13) becomes

$$P(\mathbf{s}_i|\mathbf{s}_{i+1}, x_1^i, y_1^i) = \frac{P(\mathbf{s}_{i+1}|\mathbf{s}_i)P(\mathbf{s}_i|x_1^i, y_1^i)}{P(\mathbf{s}_{i+1}|x_1^i, y_1^i)} = \frac{P(s_{i+2}|s_{i+1})P(\mathbf{s}_i|x_1^i, y_1^i)}{P(\mathbf{s}_{i+1}|x_1^i, y_1^i)}\,, \tag{5.19}$$

where $P(s_{i+2}|s_{i+1})$ is the state change probability of s_i.
With those semplifications of the trellis proposed, we can notice that the memory required in the computation is still N^3 at each iteration; however we have reduced the order of the computational load of $\sum_{\mathbf{s}_{i-1}} P(\mathbf{s}_i|\mathbf{s}_{i-1})P(\mathbf{s}_{i-1}|y_1^{i-1})$ from N^3 to N. Simulations with the third-order State are already to be run. This is because this considerations were developed too late in the thesis work and running a simulation of this computational load take almost forty hours for each σ_{PN}. The simulations follow the block diagram in the Figure 5.3.

5.3 About the Filtering of a Phase Process

In the last section the topic of the filtering a Phase process was left unsolved. Now we explain why is not an easy topic as it could appear. Considering the filter $P(z)$ defined in (5.14), in order to produce the

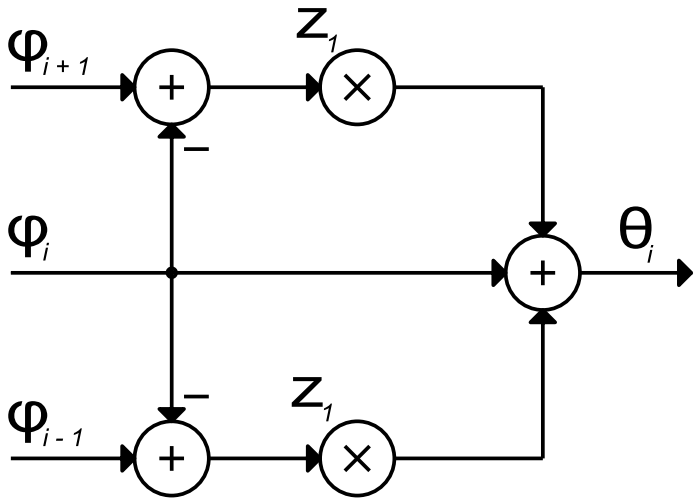

Figure 5.4: Block Diagram of the Filter for the Phase with three taps

output θ_i, one needs at the input ϕ_{i-1}^{i+1}, in fact

$$\theta_i = z_1\phi_{i-1} + \phi_i + \phi_{i+1} \ . \tag{5.20}$$

As now the normalization factor in order to obtain a process with the same variance of ϕ is omitted in order to mantain the formulas cleaner as possible. Let's focus on the last formula: one could say that there is no problem in that filtering. Now, let's consider a numerical example. If we take

$$\phi_{i-1} = \pi - \pi/20 + 2\pi \cdot 3 \quad \phi_i = \pi + 2\pi \cdot 3 \quad \phi_{i+1} = \pi + \pi/20 + 2\pi \cdot 3 \ , \tag{5.21}$$

we expect that the filtered value will be $\theta_i = \pi + 2\pi \cdot 3$. However with $z_1 = 0.2$ the result is $\theta_i = 9/5\pi + 2\pi \cdot 4$. There is a big displacement in the phase process due to the filtering of the multiples of 2π and to the ill-posed nature of the problem. Recalling the results of the Chapter 4 we said that the Discrete-Time Frequency Noise has an autocorrelation of z_1; note that is a Frequency, not a Phase. So we must filter in some way the phases by filtering the frequencies and not the phases. Remember that the filter we search must not multiply any $2k\pi$. The solution was found during the thesis work is explained in Figure 5.4. From the figure one can observe that the multiplication is done on the difference of two consecutive phases, that could be seen as a frequency. Since now we consider that the phase ϕ is slow in a time symbol: $|\phi_i - \phi_{i-1}| < \pi$. This relation, that eliminates aliases, could be seen as the satisfaction of the Sampling Theorem [AMR11, PM92]. In a

sampled phase scenario, when the phases assume only the values $\phi = \{0, \Delta, 2\Delta, \ ... \ , (N-1)\Delta\}$, where $\Delta = 2\pi/N$, may happen that $\phi_i = 0$ and $\phi_{i+1} = (N-1)\Delta$. In that case obviously one must unwrap the phase from one sample to the next in order to have maximum difference lower than π. One can build an operator that unwrap the phases while it subtracts them. Let's call this operator $-^*$:

$$a -^* b = \begin{cases} a - b - 2\pi & \text{if } a \leq \pi \text{ and } a + \pi < b \leq 2\pi \\ a - b + 2\pi & \text{if } a > \pi \text{ and } 0 \leq b < a - \pi \\ a - b & \text{elsewhere} \end{cases} . \quad (5.22)$$

If we recall the ϕ values in (5.21), filtering with the proposed scheme one obtains

$$\begin{aligned} \theta_i &= \phi_i + 0.2(\phi_{i-1} -^* \phi_i) + 0.2(\phi_{i+1} -^* \phi_i) = \\ &= \pi + 3 \cdot 2\pi + 0.2(-\frac{\pi}{20}) + 0.2(\frac{\pi}{20}) = \pi + 3 \cdot 2\pi \ , \end{aligned}$$

as we would expect.
In this section we found a way to filter phase processes that have memory in their frequency; filter a phase process in other ways has no practical and mathematical meaning, as it has been done in (5.20).

5.4 Summary

In this chapter we have seen how the bounds on the capacity of a Symbol-spaced Phase Noise Model can be computed by simulations with an auxiliary channel method. The method works recursively with trellis of probability with a discrete-value state. Then, the generalization for a third-order state is proposed. Simplifications used in order to reduce the simulation time are presented: they are based on the first-order nature of a Wiener Process. Then the problem of filtering a phase process is deepened and solved. First simulations results seem to indicate that for values of $z_1 \approx 0.2$ the bounds do not separate much from the bounds of the SPNM. This could be predictable, as we saw in Chapter 4, the power of the mismatch between the two models in very weak versus the AWGN power in almost the totality of practical systems.

Chapter 6

Conclusions

A Complete Model for the filtered and sampled signal impaired by Wiener phase noise at the oscillator has been found. The model has been compared with the the Symbol-spaced Phase Noise Model (SPNM) usually used in the literature of digital communications to take into account phase noise [BMS12, DMR00, AB11, MAV+11]). A mismatch between the two models has been found out. In particular the SPNM seems to be an approximation of the Complete Model. By simulation the power of this mismatch has been computed and it has been demonstrated that it is very weak up to a threshold that is defined on the innovation of the Wiener phase noise process in one symbol-time. In particular its standard deviation threshold seems to be $\overline{\sigma}_{PN} \approx 0.3$.
Moreover the symbol-sampled phase of the Complete Model presents memory in its innovation process: it is shown in the simulation that the innovation process has a correlation index at 1-step

$$\mathrm{NPC}_1 = z_1 \approx 0.2 \; .$$

This is the main difference between the Complete Model and the SPNM, that assumes whiteness in the phase innovation sequence. Despite this memory, that is not present in the SPNM, we can say that the Complete Model is well approximated by the Symbol-spaced Phase Noise Model if the standard deviation of the innovation process of the phase is

$$\sigma_{PN} < \overline{\sigma}_{PN} \approx 0.3 \; .$$

This memory introduced by the receive filter could be exploited by the carrier recovery system in order to improve its performance in the phase estimation. Following [BMS12] bounds on the channel capacity of the SPNM have been computed. In particular, in [BMS12] the bounds on the capacity are derived assuming a first-order Wiener process for the phase with a trellis-based recursion. In this document is reported how

to bring back the 3-order channel of the Complete Model to a 1-order Channel with a 3-dimensional state. In order to speed up simulations then is proposed a way to treat the trellis of the channel in order to reduce the computational load. In the last section the problem of filtering phase processes is addressed: since the phase seems to have an innovation process with memory, one has to filter phases in some ways. This is not very easy because filtering wrapped sequences is not as simple as it seems: a solution has been proposed by filtering the differences of consecutive phase values, under the assumption that they are all lower than π.

Appendix A

Sampling a continuous-time Wiener process

If we recall the Wiener continuous-time phase noise from (3.5)

$$\phi(t) = \sigma \int_{-\infty}^{t} w(t)dt \ , \tag{A.1}$$

where σ is a constant and $w(t) \sim N(0,1)$ is a white Gaussian variable. Now we define

$$\phi_i = \phi(iT) \ , \tag{A.2}$$

where T is the sample time and $\{\phi_n\}$ is a discrete-time process. In this appendix the aim is to demonstrate the following:

$$\begin{aligned} Hp: \ & \phi(t) = \sigma \int_{-\infty}^{t} w(t)dt \ ; \\ & w(t) \sim N(0,1) \text{ white} \ . \\ Th: \ & \phi_i = \phi_{i-1} + \overline{\sigma}\nu_i \\ & \overline{\sigma}^2 = \sigma^2 \cdot T \ ; \\ & \nu_i = N(0,1) \end{aligned} \tag{A.3}$$

Proof: if we consider

$$\begin{aligned} \phi_i = \phi(iT) = \sigma \int_{-\infty}^{iT} w(t)dt = \sigma \int_{-\infty}^{(i-1)T} w(t)dt + \sigma \int_{(i-1)T}^{iT} w(t)dt \\ = \phi((i-1)T) + \sigma \int_{(i-1)T}^{iT} w(t)dt = \phi_{i-1} + \sigma \int_{(i-1)T}^{iT} w(t)dt \ , \end{aligned} \tag{A.4}$$

the proof is reduced to the demonstration that

$$\sigma \int_{(i-1)T}^{iT} w(t)dt \sim N(0, \overline{\sigma}^2) \ . \tag{A.5}$$

Since the integral (A.5) could be seen as an infinite sum of infinitesimal elements $\sigma w(t)dt$, the central limit theorem applies (REFERENCE)

$$\sigma \int_{(i-1)T}^{iT} w(t)dt \sim N(0, \beta^2) \tag{A.6}$$

with the variance β^2 is the sum of the variances of the variables summed. Since there is not a sum, but an integral, one obtains

$$\beta^2 = \int_{(i-1)T}^{iT} \sigma^2 \cdot 1 \; dt = \sigma^2 T = \overline{\sigma}^2 \quad \text{Q.E.D.} \tag{A.7}$$

Appendix B

Quantizing a Phase Process

In this appendix we deepen the topic of quantizing in N bins the continuous-value phase process $\{\theta_i\}$ in a discrete-value process $\{s_i\}$. This is required since if one wants to operate the trellis-based algorithm of Section 5.1 he must have a finite-state channel and not a continuous one. To achieve this one must do two steps: the first is to wrap the phase modulo 2π. The second is to discretize the obtained phase. Let us focus on the discretization: since we are dividing the interval $[0, 2\pi)$ in N bins, we assign θ_i to the that the k-th bin if

$$\theta_i \in [(k - 1/2)\Delta \ , \ (k + 1/2)\Delta] \ , \tag{B.1}$$

where

$$\Delta = \frac{2\pi}{N} \ . \tag{B.2}$$

Note that for $k = 0$ one must consider

$$\theta_i \in [0 \ , \ (k + 1/2)\Delta] \ \cup \ [(k - 1/2)\Delta \ , \ 2\pi) \ . \tag{B.3}$$

Then we can say that the quantized state s_i assumes value

$$s_i = k \cdot \Delta \ \text{ if } \ \theta_i \in k\text{-th bin} \ . \tag{B.4}$$

Observe that the quantization error

$$\epsilon_Q = s_i - \theta_i \tag{B.5}$$

is distributed as a uniform random variable between $-\Delta/2$ and $\Delta/2$. Obviously when one quantizes the phase modulo 2π has to take into account the wrapped nature of the resulting phase. If one wants to reconstruct the quantized starting sequence, the phase sequence must be unwrapped [Pro01] obtaining a new sequence $\{s'_i\}$. A simple way

to do it is to proceed step by step: if one has the unwrapped phase ${s'}_{i-1}$, then

$$ {s'}_i = \begin{cases} s_i - 2\pi + {s'}_{i-1} - s_{i-1} & \text{if } s_i - s_{i-1} > \pi \\ s_i + {s'}_{i-1} - s_{i-1} & \text{if } |s_i - s_{i-1}| \leq \pi \\ s_i + 2\pi + {s'}_{i-1} - s_{i-1} & \text{if } s_i - s_{i-1} < -\pi \end{cases} . \tag{B.6} $$

Note that the term ${s'}_{i-1} - s_{i-1}$ contains the information about the accumulated multiples of 2π in the reconstruction of the unwrapped sequence.

Bibliography

[AB11] A.Spalvieri and L. Barletta. Pilot-aided carrier recovery in the presence of phase noise. *IEEE Transactions on Communications*, 59(7):1966–1974, Nov 2011.

[AMR11] F. Argenti, L. Mucchi, and E. Del Re. *Elaborazione numerica dei segnali*. Mc-GrawHill Companies, Milan, 2011.

[AN09] Y. Atzmon and M. Nazarathy. Laser phase noise in coherent and differential optical transmission revisited in the polar domain. *Journal of Lightwave Tecnology*, 27:19–29, 2009.

[BMS12] L. Barletta, M. Magarini, and A. Spalvieri. The information rate transferred through the discrete-time wiener's phase noise channel. *Journal of Lightwave Tecnology*, 30:1480–1486, 2012.

[BS50] H. W. Bode and C. E. Shannon. A simplified derivation of linear least square smoothing and prediction theory. *Proceedings of the IRE*, 38(4):417–425, 1950. ID: 1.

[DMR00] A. Demir, A. Mehrotra, and J. Roychowdhury. Phase noise in oscillators: a unify theory and numerical methods for characterization. *IEEE Transactions on Circuits Systems*, 47:655–674, 2000.

[FV88] G. J. Foschini and G. Vannucci. Characterizing filtered light waves corrupted by phase noise. *IEEE Transactions on Information Theory*, 34(6):1437–1448, 1988. ID: 1.

[LCY+11] X. Li, Y. Cao, S. Yu, W. Gu, and Y. Ji. A simplified feedforward carrier recovery algorithm for coherent optical qam system. *Journal of Lightwave Tecnology*, (5):801–807, 2011.

[MAV+11] M. Magarini, A., F. Vacondio, M. Bertolini, M. Pepe, and G. Gavioli. Empirical modeling and simulation of phase

noise in long-haul coherent optical transmission systems. *Optics Express*, 19(23):22455–22461, Nov 2011.

[MD97] U. Mengali and A. N. D'Andrea. *Synchronization techniques for digital receivers*. Springer, 1997.

[PHN09] T. Pfau, S. Hoffmann, and R. Noe. Hardware-efficient coherent digital receiver concept with feedforward carrier recovery for m-qam constellations. *Journal of Lightwave Tecnology*, (8):989–999, 2009.

[PM92] J. G. Proakis and D. G. Manolakis. *Digital Signal Processing: Principles, Algorithms, and Applications*. MacMillan Publishing, 1992.

[Pro01] J. G. Proakis. *Digital communications*. Mc-GrawHill International Editions, New York, 2001.

[SM08] A. Spalvieri and M. Magarini. Wiener's analysis of the discrete-time phase-locked loop with loop delay. *IEEE Transactions on Circuits and Systems II: Express Briefs*, 55(6):596–600, 2008.

[Tay09] M. G. Taylor. Phase estimation methods for optical coherent detection using digital signal processing. *Journal of Lightwave Tecnology*, (7):901–914, 2009.