REPORT NO. EXO24/2013

Phase noise filtering for Coherent optical communications

Thesis for the Degree of Master of Science

Pedro Fernandez Acuna

CHALMERS

Communication Systems Group
Department of Signal and Systems
CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden 2013

Abstract

The receiver's carrier phase synchronization with the received signal is essential for coherent detection, especially for those employing a high bandwidth-efficient modulation scheme such as quadrature amplitude modulation (QAM). Traditionally, carrier synchronization has been performed by a phase-locked loop (PLL) in the receiver. There are, however, two major disadvantages of this approach. The first is the sensitivity to propagation delay in the PLL feedback path. The second disadvantage, equally important, is the fixed loop bandwidth of the PLL that imposes a tradeoff between acquisition time and steady-state phase tracking performance. Particularly, a large loop bandwidth results in faster acquisition time or loop setting time, larger steady-state phase variation and cleaner VCO phase noise. Feedforward (FF) carrier synchronization technique, based on a two-stage iterative algorithm, overcomes this problem. The first stage is a symbol by symbol phase detector of the phase noise. The second stage is a hard decision phase estimator. The advantage of the FF synchronizer is that it uses both past and future symbols to estimate the carrier phase. As a result, it can achieve better performance than PLL which, as a feed back system, can only employ past symbols.

In the present Master's thesis we estimate the phase noise on a symbol by symbol basis using FF technique. In order to improve the obtained phase noise estimation, optimum digital filters - including standard Kalman, Kalman smoother and Wiener filters - are considered. We derive here the equation of the discrete-time Kalman filter step by step. We also derive the Wiener transfer function W(z) and prove that the the optimal Wiener coefficients are two exponential decaying sequences that are symmetric about the origin. We implement Kalman filter, Kalman filter smoother and Wiener filter in MATLAB. Finally we demonstrate our approach by analyzing the simulation results for a QAM modulation scheme which show that the use of linear filters may significantly reduce the phase noise estimate in a FF carrier recovery schemes.

Contents

List of Figures

List of Tables

Nomenclature

ARMA	Auto-regresive moving average
ASIC	Application-specific integrated circuits
ASK	Amplitude shift keying
AWGN	Additive white Gaussian noise
BER	Bit-error rate
CD	Chromatic dispersion
DCM	Dispersion compensation module
DFB	Distributed feedback laser
DSP	Digital signal processing
DTFT	Discrete time Fourier transform
EDFA	Erbium-doped filter amplifiers
EEPN	Electronic equalization phase noise
FF	Feed forward
FIR	Finite impulse response
HPE	Hard phase estimator
iid	independent identically distributed
IMDD	Intensity modulation and direct detection
LMS	Least mean square
LO	Local oscillator
LPEM	Linear phase estimator model

MMSE	Minimum mean square error
NADA	Alternative NDA
NDA	Non-data-aided
PDM	Polarization division multiplexing
PLL	Phase locked-loop
PMD	Polarization mode dispersion
PNM	Phase noise model
PSK	Phase shift keying
QAM	Quadrature amplitude modulation
QPSK	Quadrature phase-shift keying
ROC	Radius of convergence
RTS	Rauch-Tung-Striebel
RX	Receiver
SER	Symbol error rate
SNR	Symbol to noise ratio
TX	Transmitter
VaV	Viterbi and Viterbi
VCO	Voltage controlled oscillator

List of Symbols

a_k	Transmitted symbol
r_k	Received signal
n_k	AWGN
v_k	i.i.d.Gaussian system noise with variance σ_θ^2
n'_k	i.i.d.Gaussian measurement noise with variance
θ_k	True phase noise.
ψ_k	Soft phase estimate of θ_k
$\hat{\theta}_k$	MMSE estimate of θ_k
$\hat{\theta}_k^r$	Raw phase noise estimate of θ_k
$\hat{\theta}_k^u$	Unwrapped phase estimate of θ_k
$\hat{\theta}_{k-\Delta}$	Wiener-filtered phase noise shifted
$W(z)$	Wiener filter transfer function
w_k , $w(k)$	Wiener filter coefficients
L	Wiener filter length
L_f	Fiber length
M_s	Length of the block of V&V algorithm
Δ	Wiener filter delay
σ_θ^2	Phase noise variance, sigma2_theta in MATLAB simulation
σ^2	Additive noise variance, sigma2 in MATLAB simulation
r	Ratio between σ_θ^2 and σ^2
Q	Vector of phase noise variances in MATLAB simulation
R	Vector of additive noise variances in MATLAB simulation
$H(\omega)$	Channel frequency response
$T(\omega)$	Jones matrix accounting for PMD
β_2	Group velocity dispersion
T	Symbol period
$1/T$	Baud rate
Δ_v	Sum of the linewidths of the signals and the LO lasers

Acknowledgments

I would like to thank my supervisor Assistant Prof. Henk Wymeersch for introducing me to the World of Communication. I appreciate his vast knowledge, skill, patience and support throughout the completion of my master's thesis.

I am also deeply grateful to my family, my lovely G-ANLERA team, thanks for your unfailing support and encouragement. Finally, my thanks go to Beto, who has always been an inspiration on my journey through life.

Pedro Fernandez Acuna
Gothenburg, July 2013

Chapter 1

Introduction

1.1 Background

The commercialization in 2008 of the first 40 Gb/s coherent optical communication systems employing polarization division multiplexing (PDM), quadrature phase-shift keying (QPSK) and intradyne detection assisted by digital signal processing (DSP) marked a major milestone in long-haul transmission [15]. At the core of this major breakthrough innovation there was the coherent detection technique that took shape in the 1990s and high-speed DSP technology.

The origins of the modern communication technology, including coherent optical communication, reach back to research of the 1970s when the IMDD scheme became the mainstream in the optical fiber communication system. Such IMDD scheme had a great advantage that the receiver sensitivity was independent of the carrier phase and the state of the polarization (SOP) of the incoming signal[9] . In the late 1980s and early 1990s coherent receiver system, which forms the foundation of a digital coherent system, attracted a lot of attention as it was a promising way to obtain larger bandwidth over long distance and higher receiving sensitivity [1]. However, they were considered impractical at the time due to their high cost and complexity as well as their vulnerability to phase-noise and polarization rotations [15]. But the reason why the interest for employing coherent receivers for optical fiber communication faded in the early 1990s were the development of the erbium-doped filter amplifiers (EDFA) and the introduction of the high-capacity wavelength division multiplexed (WDM) systems.

In 2005, the demonstration of the digital carrier-phase estimation in coherent receivers stimulated a widespread interest in coherent communication again [8]. The revived interest in coherent detection is largely due to the substitution of the previously proposed analog electronic and optoelectronic modules, which were bulky, slow, expensive and largely inefficient, in coherent receivers with a relatively inexpensive, high-speed, application-specific integrated

[1] defined as the required signal power to achieve a given bit error rate

circuits (ASICs)[15] . The main driver for coherent optical communication is the possibility to compensate for transmission by using DSP [22]. The ASICs enable adaptive electronic equalization of linear impairments in fiber-optic transmission systems, i.e., chromatic dispersion (CD), polarization mode dispersion (PMD) and polarization-dependent loss and, to some extent, of fiber nonlinearities in the digital domain. This is due to the fact that in the digital coherent receivers the phase information is preserved after detection. The ASICs also allow for adaptive electronic compensation of imperfections of the analog optical transmitter and receiver front-ends such as time skew of quadrature components and polarization tributaries, quadrature and polarization imbalance, etc. They perform all standard digital receiver functions such as digital clock recovery, intermediate frequency offset [15]. Finally, in order to avoid the difficulties associated with the PLL, carrier synchronization can be done in the DSP by digital phase estimation techniques (allowing for a free running LO) avoiding being phase-locked by the PLL

However, using coherent receiver systems for optical communication systems came along with laser phase noise that is one of the biggest obstacles to its realization. Laser phase noise, defined as the phase-difference between the phase of the received carrier and the phase of the LO to the receiver, may arise in the transmitter's LO, in the receiver's LO and/or from environmental factors e.g transmission medium density variation due to temperature, pressure gradients, etc. Consequently, the output of a practical LO is not a perfect sinusoid. Instead of being an ideal pulse at the carrier frequency, the spectrum of a real LO output is spread and has a Lorentzian shape i.e the shape of the squared magnitude of a one-pole lowpass filter (LPF) transfer function [2, 1]. This effect is the major contributor to the undesired phenomena such as interchannel interference leading to increased bit-error-rates BER in coherent communication systems.

Clearly the combination of coherent detection and DSP provides new capabilities that were not possible without detection of the phase of the optical signal. "The born again coherent optical technology will renovate existing optical communication systems in the near future" [8].

1.2 Previous works and future challenges

One of the first works dealing with FF carrier recovery after the revival of interest in coherent detection was presented by Reinhold Noé [16]. He proposed a feedforward carrier recovery scheme based on regenerative intradyne frequency divider i.e. the regenerative frequency divider is extended to process baseband in-phase and quadrature (I and Q) signals. It's worth noting that no phase unwrapper is considered in [16]. A rigorous and theoretically comprehensive work of FF carrier recovery has been that of Ezra Ip and Joseph Khan[6]. In this work FF carrier recovery architecture is analyzed using Monte Carlo simulation for a single polarization case. A thorough review of detection and modulation methods with emphasis on coherent detection and digital compensation of channel impairments was presented by Ezra Ip et al. in [4]. The analysis of the phase noise requirement is often performed without consideration of the large chromatic dispersion. In reference [18], W. Shieh and Kean-Po Ho thoroughly evaluate the electronic equalization phase noise (EEPN) from the interaction between LO phase noise fluctuation and the fiber dispersion in coherent transmission systems. Chongjin Xie showed in [23], for the first, time that the LO phase noise to amplitude noise conversion caused by electronic CD compensation significantly degrades the performance of a DCM free high-speed coherent system.

In particular, the digital coherent receiver systems for 100 Gb/s-class signals came to seem possible thanks to the dramatically improvement of the digital signal processing capacity (number of gates and operating frequency) as well as higher bit rates addressing laser phase noise which was one of the biggest obstacles to the realization of a coherent receiver system [21] . With these factors in the background some test results have been reported. However the majority of the reports are confined to so-called “offline tests” while the numbers of test reports on real-time operation with an actual digital coherent receiver prototyped is relatively small. “For that reason, there are no studies or reports available that delve into the challenges of having real systems operate stably in reality and there are still problems remaining before the technology can be put to practical use”[21].

1.3 Motivation

The primary purpose of the receiver is to generate an accurate replica of the sequence of the transmitted data symbols. The receiver has to extract the synchronization information from the received signal and then use this information to achieve demodulation and detection of the transmitted data. In coherent detection accurate phase recovery is needed as the received signal is demodulated by a LO that serves as an absolute phase reference.

Recent experiments have shown that Feedforward carrier recovery schemes are more tolerant to laser phase noise than PLL [6] . Furthermore, high spectral efficiency - that describes how efficiently the given bandwidth is utilized or the ability of a modulation scheme to accommodate data within a limited bandwidth - and narrow power spectrum can be achieved with QAM by setting a suitable constellation size limited only by the noise level and linearity of communication systems.

The present Master’s thesis aims at investigating the way how to improve the phase noise estimates in the Feedforward carrier recovery scheme for coherent communication for the detection of quadrature amplitude modulation (QAM) transmission.

1.4 Delimitation

The most important limitation of the current research lies on the fact that the it does not include the analysis of the cycle slips that if undetected can significantly degrade the filter's performance that can lead to abrupt changes of mean and variance values. Other limitations are highlighted in the conclusion section as recommendations for further study which might improve upon the present study.

1.5 Contribution

We have implemented the two-stage feedforward carrier phase estimation algorithm in a coherent system for single polarization. We have developed a common framework for the carrier phase dynamic model that can be used by both Kalman and Wiener filters. The Kalman filter-based phase estimator stage as well as the Wiener filter-based phase estimator for the two-stage feedforward phase noise estimation algorithm has been designed, developed and presented in detail. We have derived the Wiener transfer function corresponding to the Wiener coefficients and proved that the Wiener coefficients consists of two exponential decaying sequences that are symmetric about the origin whose decay rate depends only on the ratio between the phase noise variance and additive noise variance.

The simulation results show that the phase noise estimation, crucial to perform before the recovery of the transmitted symbol, could significantly be improved using minimum mean square estimators.

1.6 Methodology

Although both Kalman and Wiener filters are based on the same method, namely least-squares estimation procedures, they differ significantly from each other in the way they obtain the optimal estimator. Thus, while Wiener filter involves an input-output signal model, the Kalman filter requires a state-space model and operates recursively on the stream of noisy input data to produce a statistically optimal estimate of the underlying system state. Consequently, the main task is to develop a common framework within which both filters can be used to solve the same least-squares estimation problem. Basically, it involves two steps. The first step deals with generating the input-output signal model for the Wiener filter based on Wiener process which is used for modeling of phase noise in semiconductor lasers. This latter result becomes the linear-phase noise model for Kalman filtering which is also know as the model equation in Kalman theory.

The second step becomes an estimation problem and is aimed at determining the observation linear-phase noise estimator model which is also called observation equation in the general Kalman theory. The linear-phase noise estimator model is obtained from the phase estimation stage and phase unwrapping stage in sequence. Taken together these two stages make up the first stage of the feedforward phase noise estimation algorithm. The second stage of the feedforward phase noise estimation algorithm is a linear filter optimal in the sense of the MMSE

The above mentioned signal models, namely the linear-phase noise model and the linear-phase noise estimator model, form the linear state phase model for Kalman filter.

We consider that the receiver takes a vector $r_k = (r_k, r_{k-1}, ..., r_{k-L+1})$ of L received sequential symbols. The problem consists of computing the best estimates of the carrier phases $\Theta_k = (\theta_k, \theta_{k-1}, ..., \theta_{k-L+1})$.

We employ the temporal correlation in the carrier phase by modeling the phase noise as a Wiener process as the phase at any symbol period is likely to have a value similar to the phases at adjacent symbols thereby obtaining the linear phase noise model. NDA algorithm for QAM modulation allows the phase noise θ_k to be estimated without any symbol decisions after removing the data modulation by raising the received signal to the 4th power. The raw phase noise thus obtained has the ambiguity by $\frac{\pi}{2}$ because we cannot know the absolute phase. The unwrapping procedure is aimed at removing the $\frac{\pi}{2}$ giving rise to the soft phase noise estimate ψ_k. It is worth noting that no temporal correlation is used in computing ψ_k. After unwrapping, the linear-phase-estimator model appears naturally.

The second stage involves phase noise estimation that produces the phase noise estimate $\hat{\theta}_k$. Phase noise estimation is needed to reduce effect of additive noise in the previous stage. The soft phase noise estimate ψ_k is passed through a FIR Wiener filter, with sufficient number of taps, whose output is the MMSE estimate $\hat{\theta}_k$ of the actual carrier phase θ_k. Another way to obtain the MMSE estimate $\hat{\theta}_k$ is using Kalman filter which is also optimal in MMSE sense. The phase noise estimate $\hat{\theta}_k$ can be determined by the Kalman recursion equation: $\hat{\theta}_{k+1}$=$\hat{\theta}_k$+$K(\psi_k - c\hat{\theta}_k)$, where K is the Kalman gain. We can further improve the estimation of $\hat{\theta}_k$ using a Kalman smoothing filter.

1.7 Outline

The thesis is structured as follows. In chapter 2, as a foundation for the feed forward carrier recovery algorithm, the input signal to the carrier recovery unit as well as assumptions about the receiver model are presented. The phase noise model, which gives rise to process equation in the state space system and to the input signal to the Wiener filter, is also considered. Chapter 3 is mainly concerned with the two stage iterative carrier-phase estimator algorithm including the soft phase estimator and phase noise filtering. The equations of the scalar Kalman filter and Wiener filter are derived. Kalman smoother is presented without any proof and some theoretical and practical results aspects of filtering are considered. In chapter 4 Kalman and Wiener filters, based on the FF carrier recovery algorithm, are evaluated in MATLAB. Chapter 5 contains conclusions of the thesis and suggestions for future work as well as changes that can improve the results.

Chapter 2

System model

In coherent detection the best receiver sensitivity[1] and higher bit rate signals are achieved when homodyne receiver is used [22]. However, in this case a narrow linewidth transmitter laser as well as a narrow linewidth LO laser are required. In addition, they should be phase-locked. These two requirements make the realization of a homodyne receiver difficult to implement in practice. Several receiver schemes employing high-speed signal processing has been proposed to overcome this problem. These schemes maintain the advantages of the homodyne detection without phase locking using instead digital Feedforward carrier recovery.

In this chapter we introduce the problem for a single polarization case. The phase noise process is modeled as a Wiener process. It is assumed that the symbol synchronization has been achieved and the received signal has low polarization crosstalk and are at one sample per symbol.

2.1 Discrete-time receiver model

We consider a base-band frequency synchronized communication system over a AWGN channel. The modulation interval T is considered as perfectly known at the receiver side. We assume also a constant channel phase model, then when phase noise and AWGN noise are the only impairments in the digital coherent receiver, the input signal r_k to the FF carrier synchronizer is of the form:

$$r_k = a_k e^{j\theta_k} + n_k, \qquad k = 0, 1, ..., N-1 \tag{2.1}$$

where k denotes the kth time interval, a_k is the kth data complex-valued transmitted symbols of $\frac{\pi}{2}$ rotational symmetry constellation of a unit average energy, $E\left\{\left\|a_k^2\right\|\right\} = 1$. θ_k [2] stands

[1] lowest bit-error rate at a given optical signal-to-noise ratio (OSNR) [22]

[2] We will use the character θ_k to mean both the carrier phase of a transmitted symbol and the phase noise from the Wiener process Θ.

for the uncompensated carrier phase to be estimated by observing the received signal r_k and n_k is a zero-mean circular complex-valued additive white Gaussian noise process, assumed independent of the symbols a_k and zero-mean circular (i.e. $E\{n_k^p\} = 0$ for all positive integers p, [13]) with variance $N_o/2$ in each component where N_0 is the noise spectral density. N denotes the observation window size. Furthermore the signal-to-noise ratio per symbol is defined as follows:

$$SNR = \frac{E\left\{\left\|a_k^2\right\|\right\}}{E\left\{\left\|n_k^2\right\|\right\}} = \frac{1}{N_0} \tag{2.2}$$

where $E\{.\}$denotes the expectation operator. In this project the SNR value is equal to 17.5 dB and the level of reliability, measured by symbol error rate (SER), should be less than 10^{-3}.

2.2 Phase noise model

In communication receivers, the output of a practical LO is not a perfect sinusoid. As a result, instead of being an ideal impulse at the carrier frequency, the output spectrum of the a real LO is spread and has a skirt shape. Therefore, the phase noise, which is the phase difference between the phase of the carrier signal and the phase of the LO, should be accurately estimated and compensated. For simplicity, and without loss of generality, the initial phase of the carrier can be assumed to be zero.

In an ideal communication systems, the carrier frequency oscillators of the transmitter and the receiver would be perfectly match in frequency and phase thereby permitting perfect coherent demodulation of the modulated based signal. However, in the real world, the frequency of the LO at the demodulator is not perfectly matched to the frequency of the carrier signal. The mismatch between the carrier frequency and the local oscillator's frequency is denoted here by $\Delta\nu$ and is assumed to be much smaller than the symbol rate $1/T$ [3]. The existence of $\Delta\nu$ causes the received signal to rotate at an angular speed of $2\pi\Delta\nu$ for every T seconds [14]. Thus, the strength of the phase noise will depend on the product $\Delta\nu T$ and a larger value of $\Delta\nu T$ means a faster changing carrier phase.

When the oscillator is locked to the carrier phase with the aid of a phase-locked loop, phase noise θ_k is modeled as a WSS process. However, when the LO is locked to the carrier frequency only i.e when carrier synchronization can be done through digital phase estimation techniques allowing for a free running LO, the time varying phase θ_k is satisfactorily modeled as a Wiener process. Although Wiener phase noise θ_k is not stationary the LO output $e^{j\theta_k}$ [4] can be assumed stationary with a Lorentzian spectrum[24] .

[3]This assumption can be assured with practical oscillators [14]

[4]LO output in base-band complex form as a function of the time-varying phase noise can be expressed as $e^{j\theta_k}$

In most semiconductor laser, phase noise can be modeled as Wiener process [6] :

$$\theta_k = \sum_{m=-\infty}^{k} \nu_m \tag{2.3}$$

where ν_m are independently identically distributed Gaussian random variables with zero mean and variance $\sigma_\theta^2 = 2\pi\triangle_\nu T$. $\triangle_\nu$ is the sum of the 3-dB linewidths of the signal and LO lasers and T is the symbol period. As we mentioned above, we will initially assume that the frequencies of the signal and LO lasers are synchronized so that the time variation in θ_k is due to phase noise only. Temporal correlation in the carrier phase at any symbol period is introduced by the Wiener process as the phase at any symbol period is likely to have a value similar to the phases at adjacent symbols.

The discrete-time Wiener phase noise can be expressed as [24]

$$\theta(k) = \theta(k-1) + \nu(k) \tag{2.4}$$

where $\nu(k)$ is a zero-mean white Gaussian with variance $\sigma_\theta^2 = 2\pi\triangle_\nu T$.

The phase noise model given by Equation 2.4, can be understood as the process equation in the state space model for Kalman filtering given by Equation 3.19. Equation 2.4 is also considered as input to the Wiener filter.

Chapter 3

Feedforward carrier phase estimation algorithm

In this chapter we begin by presenting the foundation for the Feedforward carrier phase algorithm. We consider a coherent digital communication system employing linear modulation format which uses a symmetric signal constellation, such as QAM, that lends itself to phase ambiguity due to the quadrant symmetry of the constellation. We assume that carrier recovery is performed separately so quantities such as power noise and symbol rate are referenced for one polarization. As seen in the previous chapter, we will initially assume that the TX and LO lasers are frequency-synchronized but not phase-synchronized so that the evolution of θ_k is due to phase noise only.

3.1 Introduction.

Since the linewidth of the semiconductor DFB lasers used as the transmitter and LO typically ranges from 100 kHz to 10 MHz, the true carrier phase varies much slowly than the phase noise modulation [9]. Therefore, by averaging the carrier phase over a many symbols interval it is possible to obtain an accurate phase noise estimate. [9]. As shown in Fig. 3.1, the FF carrier recovery unit is performed in two stages. The first step is the soft phase estimator which computed the symbol-by-symbol phase estimate ψ_k of the actual carrier phase θ_k. The soft phase estimate ψ_k is the input to both Kalman and Wiener filters.

The second stage, called phase noise filtering is implemented by both Kalman and Wiener filters. The phase noise filtering is aimed at improving the phase noise estimates obtained after unwrapping. In the first case, the phase noise filtering consists of a linear filter $W(z)$ whose output $\hat{\theta}_{k-\Delta}$ (Δ is the filter delay) is the MMSE of ψ_k. The primary limitation to the Wiener filter is that both the desired signal θ_k and the observation signal ψ_k must be jointly white-sense stationary (WSS). Since most processes encountered in practice are nonstationary, this constrains the usefulness of Wiener filters. In this regards, Kalman filter is a next step in the evolution which drops the stationary constraint.

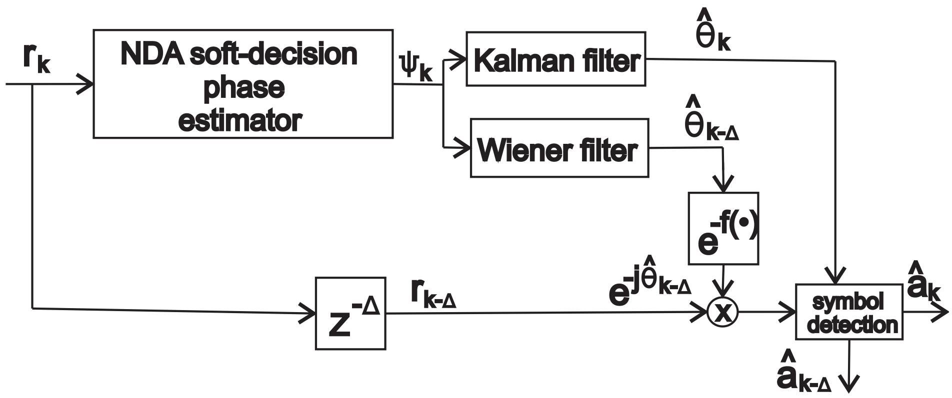

Figure 3.1: Two-stage Feedforward carrier phase estimator

A variety of methods are used to estimate carrier phase, each has its advantages and drawbacks. In this project we will use the Fourth power phase estimator which is a special case of the Viterbi and Viterbi (V&V) algorithm. A graphical view of the fourth power estimator is given in Fig. 3.2. The V&V algorithm computes the rough phase noise estimation $\hat{\theta}_k^r$. It also provides the linear phase estimator model which can be understood as the observation equation, Eq. 3.20, in the system state space for Kalman filtering. The variance σ^2 for the AWGN in the linear phase estimator model is numerically determined by the SNR which, in turn, is determined by the received symbol error rate SER. The SER is the constrain that should be fulfilled and in this particular case is set to 10^{-3}. The final goal in this chapter is to estimate θ_k using FF algorithms which will allow derotation of r_k by multiplying it with $e^{-j\hat{\theta}_k}$.

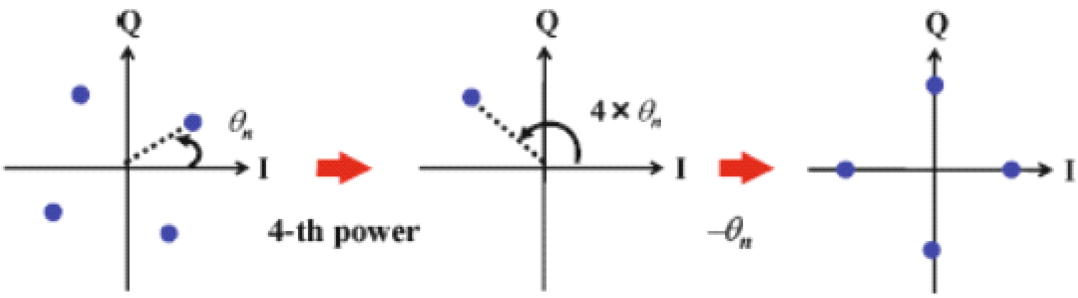

Figure 3.2: Principle of the 4th power phase estimation. Taking the 4th power of the received complex amplitude we can eliminate the phase modulation and measure the phase noise.

Detailed description of each of stages of the FF carrier recovery algorithm is given in the following sections. In addition, two important results used in the derivation of Wiener filter equations are considered in this chapter. The derivation of these two results are presented in the Appendix A and and Appendix B.

3.2 NDA soft-decision phase estimator.

The NDA soft-decision phase estimator is implemented using the V&V carrier recovery stage followed by the unwrapping stage as shown in Fig. 3.3. The rotational symmetry of the transmitted signal constellation allow us to rise the received signal r_k to the fourth power to remove the data modulation from the QAM signal.

A running average is used over a predefined number of symbols of r_k. Thus, for an $2N+1$ symbol carrier phase estimation, the N forerunner and the N after-runner symbols are considered. The argument then gives the raw phase estimate $\hat{\theta}_k^r$. Since $\hat{\theta}_k^r$ fall in the range $[0, \frac{\pi}{2})$, they need to be unwrapped to remove discontinuity in the raw estimates from the previous stage. The soft phase estimate ψ_k^u, obtained after unwrapping, is then fed into the phase noise filtering stage. Notice that for simplicity, throughout this thesis, the output for unwrapping stage ψ_k^u will be denoted by ψ_k.

Figure 3.3: NDA soft-decision phase estimator

3.2.1 Viterbi and Viterbi Estimator

The Viterbi and Viterbi estimator is a carrier phase estimator designed for $\frac{\pi}{2}$ rotational symmetric constellations such as QAM. The set of complex-valued transmitted symbols a_k in Equation 2.1 , can be written as

$$a_k = e^{\frac{jm\pi}{M}} \tag{3.1}$$

where $m = 1,..,M$ and M is the order of the rotational symmetry in Equation 2.1. From 3.1, it follows that a_k can only take on values from the finite set given by

$$a_k = \{1, -1, -j, j\} \tag{3.2}$$

the forth power applied to a_k always gives 1. Raising r_k to the 4-th power we get

$$r_k^4 = \left(a_k e^{j\theta_k} + n_k\right)^4 \tag{3.3}$$

As a result, the modulated signal is removed by the 4-th power calculation and the white noise is suppressed by the averaging so the received signal is determined as

$$r_k^4 = e^{j4\theta_k} + m_k \tag{3.4}$$

where m_k is the sum of the unwanted cross terms between the signal and AWGN [6]. Now taking the argument of r_k^4 and scaling by 4 we get the raw phase noise estimate $\hat{\theta}_k^r$.

The V&V estimator is in fact a kind of block window estimator as the received signal r_k is processed in blocks of length M_s. In practice, the phase estimate in the kth block is determined by [5] .

$$\hat{\theta}_k^r = \frac{1}{4}\measuredangle\left(\sum_{l=s_k}^{s_k+M_s-1} r_l^4\right) \tag{3.5}$$

where the argument function $\measuredangle(\cdot)$ returns phase estimation values between $\left[0, \frac{\pi}{2}\right]$ and s_k is the index of the first symbol in the kth block. As mentioned previously, the removal of the additive Gaussian noise is accomplished by taking the average over an observation period. Therefore a large value of M_s minimizes the effect of the noise while, on the other hand, a small value of M_s is required to accurately track variation in the carrier phase.

A good insight into the tradeoff between the tap numbers M_s and the variance can be obtained if assuming that the carrier phase is slowly varying so that it is considered constant over a series of consecutive symbols. The variance of the phase estimation error due to additive noise is then reduced by a factor equal the symbol sequence length M_s . The process itself introduces an error in phase estimation as the carrier phase is actually not constant due to the finite beat linewidth of transmitter and LO lasers . This error thus increases with M_s [10]. A trade off between these two effects suggests that the tap number for the additive noise estimator must be optimized. The MATLAB code of V&V is found in Appendix E.

3.2.2 Phase Unwrapping

A close look at Equation 3.5 shows that the $\hat{\theta}_k^r$ is computed by multiplying by the factor of $\frac{1}{4}$ the results gathered withing an observation period. Consequently, the raw phase noise estimate sequence $\hat{\theta}_k^r$ can only take wrapped values in the range from 0 to $\frac{\pi}{2}$ whereas the carrier model, described by Equation 2.3 , assumes θ_k to be an unwrap angle extending from $-\infty$ to $+\infty$.

The underlying idea behind phase unwrapping procedure consists in choosing the nearest phase noise that is less than $\pm\frac{\pi}{2}$ rad. It is obtained by adding multiples of $\pm\frac{\pi}{2}$ rad to the carrier phase prior to unwrapping, that is to the raw phase estimate $\hat{\theta}_{k-1}^r$, thus ensuring that the magnitude of the phase difference between two adjacent elements in the unwrap estimated phase noise sequence $\hat{\theta}_k^u - \hat{\theta}_{k-1}^u$ is the smallest . This is obtained using the floor operator $floor(x) = \lfloor x \rfloor$ which returns the largest integer not greater than x. In [6] two ways for numerically computing phase unwrapping are presented. In this project we will used what the author considered the most robust variation given by

$$\hat{\theta}_k^r = \hat{\theta}_{k-1}^r + p\frac{\pi}{2} \tag{3.6}$$

where p is obtained using the floor operator

$$p = \left\lfloor \frac{1}{2} + \left(\left(\psi_{k-1}^u + \psi_{k-2}^u + \psi_{k-3}^u \right) \frac{1}{3} - \psi_k^u \right) / \frac{\pi}{2} \right\rfloor \tag{3.7}$$

In Eq. 3.7, the three first unwrap phase noise values, ψ_{k-1}^u, ψ_{k-2}^u and ψ_{k-2}^u are computed by

$$p = \left\lfloor \frac{1}{2} + \left(\psi_{k-1}^u - \hat{\theta}_k^r \right) / \frac{\pi}{2} \right\rfloor \tag{3.8}$$

Once the $\frac{\pi}{2}$ jumps, that are present in the wrapped phase noise estimate sequence due to the constrain on $\hat{\theta}_k^r$, have been removed the raw-wrapped phase noise estimate sequence becomes a unwrap phase noise estimate sequence ψ_k^u of continuous form, free from $\frac{\pi}{2}$ jumps and hence making the phase noise estimates ψ_k^u usable in further processing.

Since m_k in Eq. 3.4 contains high-order terms, it is clear that it is not exactly a Gaussian noise process. Now taking the argument of r_k^4 3.4 and scaling by 4 and assuming that the additive noise m_k is small, ψ_k can be expanded using a small angle approximation

$$\psi_k = \theta_k + n_k' \tag{3.9}$$

Where n_k' is the Gaussian measurement noise. At high SNR it can be shown that n_k' is approximately i.i.d. with zero mean and $\sigma^2 = \eta(M,\gamma)\frac{1}{\gamma}$ where γ is the SNR per symbol, M is the order of the rotational symmetry and η is a function depending on γ and M [6]. Notice that for simplicity's sake, the superscript $'u'$, used to denote the unwrapped phase has been dropped in the Equation 3.9 . The MATLAB script for both V&V and phase unwrapping can be found in Appendix E.

3.3 Phase noise filtering.

First, we need to establish a common framework for the carrier phase dynamic model that can be used by both Kalman and Wiener filters. So, we begin by considering θ_k, that was modeled as a discrete-time Wiener noise, defined by the Equation 2.4, as the state parameter recursively updated by Kalman filter. Consequently, the Equation 2.4 becomes the state transition equation of the discrete state-space model. Moreover, the true phase noise θ_k, modeled by the Equation 2.4 as the running sum of the input frequency noisy v_k , will be considered the desired signal to be recovered from the noisy observation ψ_k by the discrete Wiener filter, see Fig. 3.4.

On the other hand, Equation 3.9 is considered as the observation equation or measurement equation of the discrete state-space model which gives the observed value ψ_k as a function of θ_k corrupted by the Gaussian measurement noise n'_k. It is worth noting that Kalman and Wiener filter are equivalent in steady state [17]. They also depend upon the same process and measurement covariances. When the delay in the Wiener filter output is reduced to zero, the Kalman filter, which is a generalization of the Wiener filter, can be used to estimate θ_k instead [11, 12] . Improved phase noise estimation can be achieved using Kalman filter smoothing. In what follows, we will consider θ_k and $\theta(k)$ to be interchangeable as they both represent samples taken at the symbol centers.

3.3.1 Wiener Filtering.

Traditionally, filters are designed for a desired frequency response. However, in case of the Wiener filter, its design takes a different approach which consists in reducing the amount of noise present in a signal by comparison with an estimation of the desired noiseless signal.

Consider the Fig. 3.4 that depicts the linear phase estimator model determined by the observation Eq. 3.9 . The input frequency noise v_k is a white Gaussian process whose running sum is θ_k. At the output of the soft phase estimator, θ_k is corrupted by the noise n'_k to produce ψ_k. The best linear estimate of θ_k that can be made is applying a Wiener filter to ψ_k [7] . Thus, we pass ψ_k through a Wiener Filter $W(z)$ whose output is the MMSE estimate $\hat{\theta}_{k-\Delta}$ of the carrier phase θ_k where Δ is the filter delay. Put simply, the task of making a phase estimate that we address is to estimate θ_k from the observable ψ_k .

The phase noise model and the linear phase estimator model given by Equation 2.4 and Equation 3.9 respectively, give rise to the following system of equations:

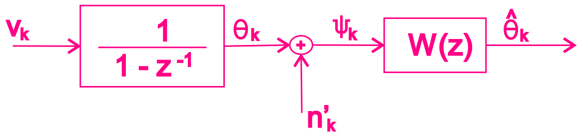

Figure 3.4: Carrier-phase estimator for Wiener's signal model

$$\theta(k) = \theta(k-1) + v(k), \qquad v(k) \sim N\left(0, \sigma_\theta^2\right) \tag{3.10}$$

$$\psi(k) = \theta(k) + n'(k), \qquad n'(k) \sim N\left(0, \sigma^2\right) \tag{3.11}$$

With ψ_k as the input to the filter, the MMSE Wiener-filtered phase noise estimate $\hat{\theta}(k)$ is evaluate by the convolution of the Wiener filter coefficients $w(k)$ and $\psi(k)$,

$$\hat{\theta}(k) = \sum_{l=\infty}^{+\infty} w(l)\, \psi(k-l) \tag{3.12}$$

So the problem considered here is as follows: Given the system obtained from the estimation model in Fig. 3.4, find the optimum filter coefficients $w(l)$ that minimize

$$E\left\{|\hat{\theta}(k) - \theta(k)|^2\right\} \tag{3.13}$$

3.3.1.1 Wiener Transfer Function

It can be shown that the Wiener filter $W(z)$ is given by [6]

$$W(z) = \frac{rz^{-1}}{-1 + (2+r)z^{-1} - z^{-2}} \tag{3.14}$$

where

$$r = \frac{\sigma_\theta^2}{\sigma^2} > 0 \tag{3.15}$$

r is the ratio between the magnitude of the phase noise variance σ_θ^2 and the additive noise variance σ^2. The derivation of the Wiener transfer function is presented in Appendix C.

3.3.1.2 Wiener Coefficients. Inverse Transform of $W(z)$

The Wiener filter given by Equation 3.14 has two poles at:

$$\alpha_{1,}\alpha_2 = \left(1 + \frac{r}{2}\right) \pm \sqrt{\left(\frac{1+r}{2}\right)^2 - 1} \tag{3.16}$$

These two poles are inverse of each other with α_1 inside the unit circle mapping to a causal sequence and α_2 outside the unit circle mapping to an anticausal sequence. It can be shown that the MMSE filter for $\Delta = 0$ has coefficients:

$$w_k = \begin{cases} \frac{\alpha r}{1-\alpha^2} & \alpha^k, \quad k \geq 0 \\ \frac{\alpha r}{1-\alpha^2} & \alpha^{-k}, \quad k < 0 \end{cases} \tag{3.17}$$

The usual way to derive the Wiener coefficients w_k from $W(z)$ is using the contour inversion formula but since we are not dealing with numerical values the computation might be a bit cumbersome. In Appendix D, using fraction expansion and some math tricks, we provide a simple way to obtain the Wiener coefficients w_k.

We note that the Wiener filter coefficients in Equation 3.17 consists of two exponentially decaying sequences that are symmetric about $k = 0$. The decay rate of the Wiener filter coefficients depends only on the ratio between the magnitude of the additive noise variance σ^2 and the phase noise variance σ_θ^2. In the limit of the low phase noise, i.e. when $\sigma^2 \ll \sigma_\theta^2$, we have $\alpha \to 1$. The decay rate is slow because of the long coherence time of the phase noise process. Conversely, in the limit of high phase noise, when $\sigma^2 \gg \sigma_\theta^2$, soft decision phases on either side of symbol k rapidly become poor estimator of θ_k , so $\alpha \to 0$, and the Wiener coefficients decay rapidly.

3.3.1.3 FIR Wiener Filter

Since the Wiener filter coefficients are non causal with two exponentially decaying tails toward the past and the future, it cannot be implemented. In practice, one can truncate Equation 3.12 to L significant coefficients and implement it as an FIR filter with delay Δ without significant performance degradation. Hence, the best estimates of the carrier phase at symbol $k-\Delta$ is the output $\hat{\theta}_{k-\Delta}$ determined as the convolution of its input $\psi = \psi_k,\ \psi_{k-1}$, ..., ψ_{k-L+1} with the Wiener coefficients w_k,

$$\hat{\theta}_{k-\Delta} = \sum_{l=0}^{L-1} w_l \psi_{k-l} \tag{3.18}$$

In Fig 3.5, the coefficients of FIR Wiener filter are shown for different values of $r = \frac{\sigma_\theta^2}{\sigma^2}$ and delay Δ. Intuitively, the lowest MSE is obtained when the delay Δ is equal to the half the filter length i.e. $\Delta = [\frac{L-1}{2}]$ which results in the same number of soft phases used from either side of the symbol period $k-\Delta$ for estimating $\hat{\theta}_{k-\Delta}$. Fig. 3.6 shows the MATLAB simulation of FIR Wiener filter for a delay $\Delta = \lfloor \frac{L-1}{2} \rfloor$. The true phase noise θ_k and its corresponding noisy measurement or soft phase noise estimate ψ_k are depicted in Fig. 3.6. From this figure, we can also see that the FIR Wiener filter resulted in significant noise reduction as shown by the Wiener-filtered phase noise.

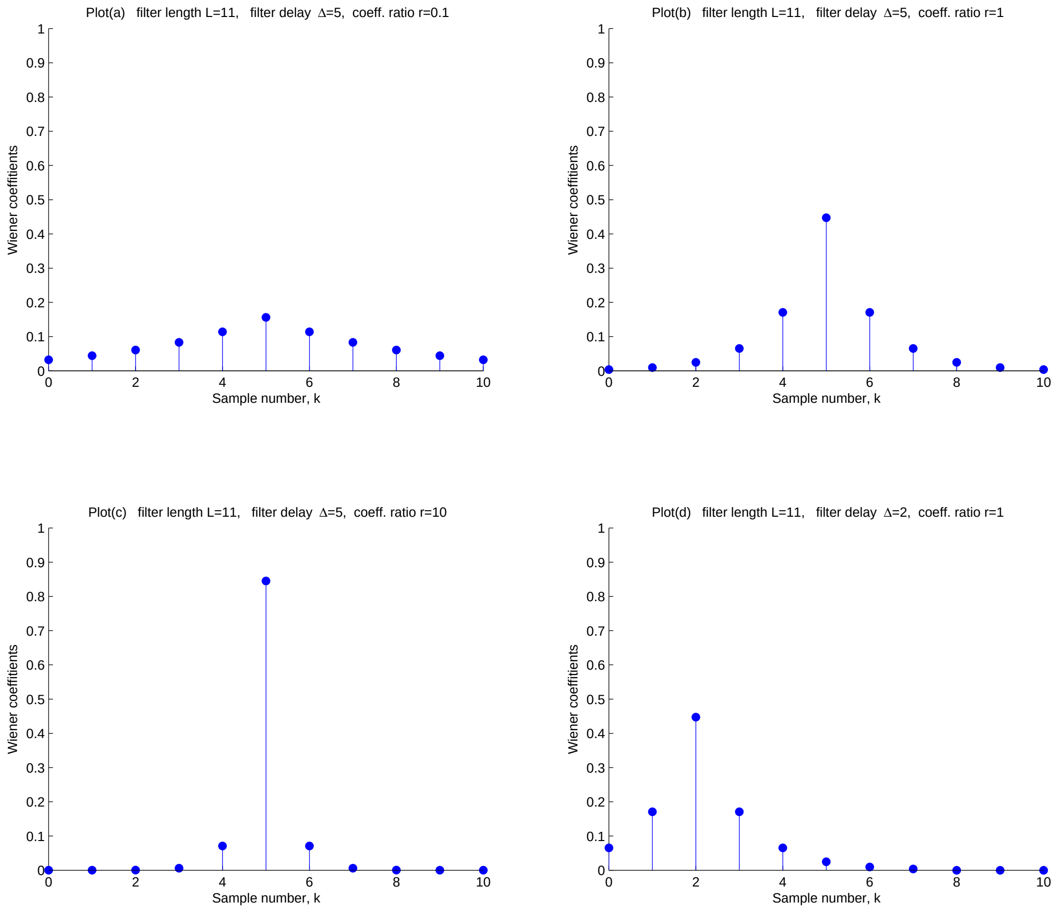

Figure 3.5: MATLAB simulation of FIR Wiener coefficients for parameters: Wiener filter length L , delay Δ and ratio $r = \frac{\sigma_\theta^2}{\sigma^2}$

3.3.2 Kalman filtering

In this section we will consider the well-known discrete time linear Kalman filter and the discrete time fixed interval Rauch-Tung-Striebel (RTS) smoother or Kalman smoother for short, which is an extension to the standard Kalman filter, based on the carrier phase dynamic model. The difference between Kalman filter and Kalman smoother lies in their analyzed time interval. Kalman smoother uses all the measured data before and after the time of estimation which does not fulfill the causality condition. On the other hand, Kalman filter estimates the state at the present time in a way that the measurement data used not object the causality for filtering. In what follows, the discrete standard Kalman filter is referred to as Kalman filter and the RTS smoother as Kalman smoother

In the following section the discrete scalar Kalman filter theory will be presented in a clear and and practical way. Kalman equations are derived step by step. The derived equations are

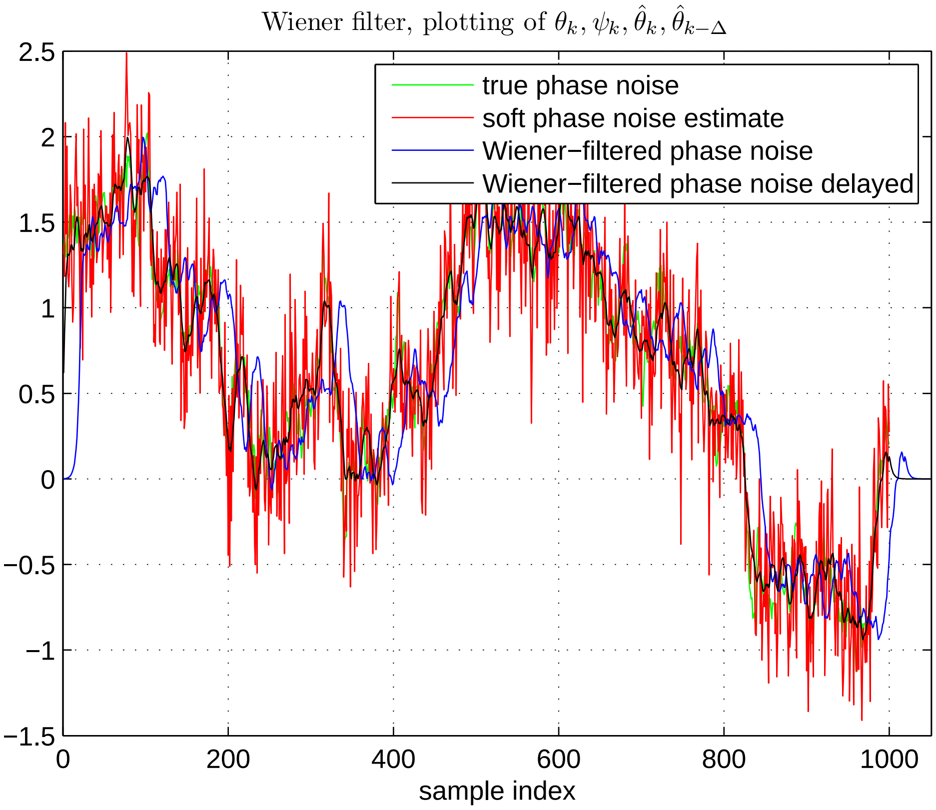

Figure 3.6: Wiener filter simulation

used for simulation in MATLAB. The Rauch-Tung-Striebel formula for Kalman fixed interval smoother is presented without proof. The performance of Kalman filter and Kalman smoother are evaluated for different values for the phase noise and additive noise variances. The MATLAB code are included in the Appendix E.

3.3.2.1 Discrete time Kalman filter.

Discrete Kalman filter is based on linear dynamic systems in the discrete time domain, hence it is capable of dealing with potentially time varying signal as opposed to Wiener filter. The Kalman filter is essentially a set of equations that implement a predictor-corrector type estimator that is optimal in the sense that it minimizes the estimated error covariance based on linear minimum mean square error (LMMSE) [1] criterion for problems in which the system can be described through a linear model and in which system and measurement noises are white and Gaussian. The whiteness of the noise implies that the noise value is uncorrelated in time. It also implies that the noise has equal power at all frequencies which results in a noise with infinite power that cannot really exist in real life. However, since any physical system of interest has a certain frequency bandpass, replacing it with a white noise, which has a bandpass above sys-

[1] LMMSE, as well as MMSE , is an estimator which seeks to minimize the MSE but, unlike MMSE, LMMSE constrains the estimator to be linear.

tem's frequency bandpass, make the mathematics involved in the filer computation simplified and more tractable. Gaussianness implies that the noise is assumed to be normal distributed, that is, at any single point in time, the probability density of the Gaussian noise amplitude takes on the shape of a normal bell-shape curve. When these three conditions are met, namely linearity of the system, whiteness and Gaussianness, the Kalman filter can be shown to be the best linear filter over the class of all linear filters.

One fact worth mentioning is the difference between Kalman and Wiener filter. While the causal Wiener and the Kalman filter are in general different, if we apply the Kalman filter to an stationary process arising from an LTI system over an infinite time horizon, the corresponding steady-state Kalman filter reduces to the causal Wiener filter. This result makes perfect sense since in this case they are both solving the same problem. In this case, the causal Wiener filter is a frequency domain solution while the Kalman filter is a time domain solution.

Kalman filter	Causal Wiener filter
Recursive algorithm	Close form solution
Nonstationary or stationary process in discrete time	Stationary process in continuous time
Time varying filter, i.e. $K(k)$ changes with k	LTI filter
Finite observation interval	Infinite observation interval

Table 3.1: Comparison of the Kalman filter and causal Wiener filter

The Table 3.1 displays a briefly comparison of the Kalman and Wiener filters. From it, we can deduce that Kalman filter presents some advantages over Wiener filter. In fact, the Wiener filter itself has a serious limitation: it applies only to stationary processes i.e. systems whose dynamics are constant. Kalman filter solves this problem as it applies to both stationary and nonstationary processes. Another limitation is that Wiener filter must be carried out by convolution making it extremely slow to execute. Moreover, Kalman describes his filter using state space model models, which, unlike Wiener filter, enables the filter to be used as either a filter, a smoother or a predictor. Finally, defining the filter in terms of state space model also simplifies the implementation of the filter in the discrete domain. All these reasons have contributed to the widespread use of Kalman filter.

The Kalman filter equations can be derived in several ways, our approach is to keep things simple and straightforward. The basic idea is considering only first order difference equations and perform optimal updates step by step. A brief presentation of the Kalman filter when the input is a vector quantity is also considered. The derived equations are being used in MATLAB simulation. The MATLAB code of Kalman and Wiener filters are presented in Appendix E.

3.3.2.2 The formulation of the Kalman filtering approach

Because of the time delay in the PLL, its time loop cannot be modeled as a standard Kalman filter [2] . Feedforward carrier recovery scheme, that can be implemented in digital hardware without PLL, can be advantageously used along with discrete standard Kalman filter for demodulating coherent optical signals. To that end, the phase noise estimation will be modeled as a Kalman filtering problem and the process and measurement variances, established in the preceding sections, will be incorporated in the Kalman algorithm.

3.3.2.3 Notation

In order to make the derivation of Kalman equations understandable and easy to follow we introduced the notation used in the derivation and explain its usage to the reader.

Notation	meaning
$\hat{\theta}\,(1\|0)$	predicted state estimate
$\hat{\theta}\,(1\|1)$	updated state estimate
$\tilde{\theta}\,(1\|0) = \theta\,(1) - \hat{\theta}\,(1\|0)$	predicted estimation error
$\tilde{\theta}\,(1\|1) = \theta\,(1)$ - $\hat{\theta}\,(1\|1)$	updated estimation error
$P\,(1\|0) = E\left(\tilde{\theta}\,(1\|0)\,\tilde{\theta}^T\,(1\|0)\right)$	predicted error covariance
$P\,(1\|1) = E\left(\tilde{\theta}\,(1\|1)\,\tilde{\theta}^T\,(1\|1)\right)$	updated error covariance.

Table 3.2: Notation for derivation of scalar Kalman filter

remark 1. The predicted state estimate $\hat{\theta}\,(1|0)$ is a priori state estimate at step k= 1 given knowledge of the process prior to step k= 1.

remark 2. The update state estimate $\hat{\theta}\,(1|1)$ is a posteriori state estimate at step k= 1 given the measurement $\psi\,(1)$

remark 3. The predicted error covariance $P(1|0)$ is the mean squared error in the estimate $\hat{\theta}(1|0)$

remark 4. The updated error covariance $P(1|1)$ is the mean squared error in the estimate $\hat{\theta}\,(1|1)$, after including the first measurement $\psi(1)$.

remark 5. For convenience, we change the notation in Equation 3.20 by replacing $\hat{\theta}\,(k)$ with $\psi(k)$ as the notation $\hat{\theta}\,(k)$will be used for the Kalman predicted state estimate in the text.

[2] It is however possible to model the time loop as a modified Kalman filter

The measurement process is central in Kalman filter. The random variable $\theta(k)$ can be in either predicted or updated state. Before a single measurement $\psi(k)$ is included $\theta(k)$ should be predicted. After including $\psi(k)$, the random variable $\theta(k)$should be updated with the new information.

3.3.2.4 State-space phase-noise model

Instead of considering a random process as the output of a linear time-invariant system with a white noise input (e.g. ARMA, AR, MA processes), the state-space models can be used for signal estimation from noisy measurement. Together, the phase noise model given by Equation 3.19 and the linear phase noise estimator given by Equation 3.20 make up the state space phase noise model:

$$\theta(k) = a\,\theta(k-1) + v(k) \tag{3.19}$$

$$\psi(k) = c\,\theta(k) + n'(k) \tag{3.20}$$

Equation 3.19 models the dynamic of the state variable $\theta(k)$ and Equation 3.20 models the observed $\psi(k)$. In the standard Kalman filter terms they are known as state equation and observation equation respectively. Fig. 3.7 shows the signal model for the discrete scalar Kalman filter.

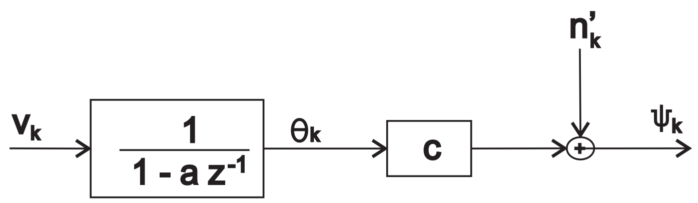

Figure 3.7: Signal model for scalar Kalman filter

3.3.2.5 Problem Statement

The task of the the Kalman filter is the following: “Given the state-space model that includes the state equation 3.19 and the observation equation 3.20 , how can we filter $\psi(k)$ so as to estimate the phase noise $\theta(k)$ at time k while minimizing the effects of $v(k)$ and $n'(k)$?.

In our development of the scalar Kalman filter we will let $\hat{\theta}(k|k)$ denote the best linear estimate of $\theta(k)$ at time k given the observations $\hat{\theta}(i)$ for $i = 1, ..., k$, and we will let $\hat{\theta}(k|k-1)$ denote the best estimate of $\theta(k)$ given the observations up to time $k-1$.

3.3.2.6 Derivation step by step

The Kalman filter algorithm works in two steps. In the prediction step, the Kalman filter algorithm produces the estimate of the current true phase noise along with its related variance. Once the outcome of the next measurement, corrupted with some among of error, becomes available, this estimate is updated using a weighted average with more weight being given to estimates with higher certainty.

Before deriving of the discrete scalar Kalman filter equations, we need to make the following assumptions:

- The Gaussian system noise $v(k)$ and Gaussian measurement noise $n'(k)$ are additive white Gaussian noises with

$$E[v(k)] = 0, \quad E[n'(k)] = 0$$

$$Var[v(k)] = \sigma_\theta^2, \quad Var[n'(k)] = \sigma^2.$$

The two noise sources are independent of each other and independent of the input.

- We assume that the system starts at $k = 0$. Hence it is necessary to specify an initial system state $\theta(0)$ which is independent of $v(k)$ and $n'(k)$.

- For the random variable $\theta(0)$ the following apriori information should be provided:

$$E[\theta(0)] = \theta_0 \tag{3.21}$$

$$Var[\theta(0)] = P_0 \tag{3.22}$$

Let $k = 0$ be the initial time. Clearly, when no measurement is available, the LMMSE(Linear Minimum Mean Square Error) estimator of $\theta(0)$ is equal to the initial guess θ_0

$$\hat{\theta}(0|0) = \theta_0$$

and its related error variance is

$$Var\left[\tilde{\theta}(0|0)\right] = E\left[(\theta(0|0) - \theta_0)^2\right] = P_0$$

In addition, $\theta(0)$ is unbiased because $E[\theta(0)] = \hat{\theta}(0|0)$.

PREDICTION STEP

Before collecting a sample $\psi(1)$ we need to predict $\theta(1)$. So lets $\hat{\theta}(1|0)$ be the one-step-ahead predictor of $\theta(1)$. $\hat{\theta}\,(1|0)$ can be determined as a function of the old $\hat{\theta}(0|0)$ by the following linear equation:

$$\hat{\theta}(1|0) = \alpha\hat{\theta}(0|0) + \beta \tag{3.23}$$

Now taking expectations of both sides of the Equation 3.19 for time instance $k = 1$ we get

$$E[\theta(1)] = E[a\theta(0) + v(1)] = a\hat{\theta}(0|0) \tag{3.24}$$

The next step is to choose α and β so that the LMMSE $E[(\theta(1) - \hat{\theta}(1|0))^2]$ is minimized. Comparing Equation 3.23 with Equation 3.24 we see that for unbiasedness we should choose $\alpha = a$ and $\beta = 0$. Thus the best prediction of $\theta(1)$ is given by:

$$\hat{\theta}(1|0) = a\hat{\theta}(0|0) \tag{3.25}$$

We can verify LMMSE property using orthogonality principle a.k.a projection theorem. Consider the Fig 3.8. We are looking along the vector $\hat{\theta}\,(0|0)$ for a point closest to $\theta\,(1)$. From the geometric interpretation it is intuitive that the smallest error vector $\tilde{\theta}(1|0)$ is the orthogonal to the vector $\hat{\theta}\,(0|0)$.

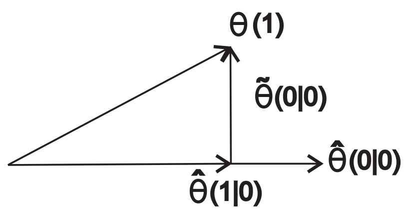

Figure 3.8: Orthogonality principle

From Fig. 3.8 we can see that vector error $\tilde{\theta}(1|0)$ is simply the difference between $\theta(1)$ and $\hat{\theta}(1|0)$ that is,

$$\tilde{\theta}(1|0) = \theta(1) - \hat{\theta}(1|0) \tag{3.26}$$

and the error variance $Var[\hat{\theta}(1|0)]$ is minimized if the vector $\tilde{\theta}\,(1|0)$ is perpendicular to the vector $\hat{\theta}\,(0|0)$. Now in order to determine the predicted estimation error $\tilde{\theta}\,(1|0)$ we should first determined $\theta\,(1)$. Equation 3.19 for $k = 1$ yields

$$\theta\,(1) = a\,\theta\,(0) + v(1) \tag{3.27}$$

Replacing Equation 3.25 and Equation 3.27 into Equation 3.26 gives

$$\tilde{\theta}(1|0) = a\theta(0) + v(1) - a\hat{\theta}(0|0) = a\left(\theta(0) - \hat{\theta}(0|0)\right) + w(1)$$

Thus the predicted estimation error is given by:

$$\tilde{\theta}(1|0) = a\tilde{\theta}(0|0) + v(1) \tag{3.28}$$

where $\tilde{\theta}(0|0) = \theta(0) - \hat{\theta}(0|0)$.
The next step is to compute the predicted error covariance $P(1|0)$

$$P(1|0) = Var\left(\tilde{\theta}(1|0)\right) = E\left[\left(a\tilde{\theta}(0|0) + v(1)\right)\left(\tilde{\theta}^T(0|0)\,a^T + v^T(1)\right)\right]$$

Using the fact that $v(k)$ is white Gaussian with covariance σ_θ^2 we finally obtain the prediction error covariance,

$$P(1|0) = aP(0|0)\,a^T + \sigma_\theta^2 \tag{3.29}$$

UPDATE STEP

Having found the predicted state estimate $\hat{\theta}(1|0)$ and the predicted error covariance $P(1|0)$, in order to update $\hat{\theta}(1|0)$, we can now incorporate the first sample $\psi(1)$ corrupted with some amount of error . Here we either assume that the updated state estimate $\hat{\theta}(1|1)$ is a linear weighted sum of the prediction and the new observation $K\psi(1)$ or assume that the linearity of the linear phase estimator model, Equation 3.20, implies that we can add any linear combination of $K\psi(1)$ to Equation 3.30 since this is a zero mean stochastic variable. Thus

$$\hat{\theta}(1|1) = \alpha\hat{\theta}(1|0) + K\psi(1) \tag{3.30}$$

Note that for notational simplicity we sometimes denote the Kalman gain $K(k)$ by K. The updated estimation error is given by

$$\tilde{\theta}(1|1) = \theta(1) - \hat{\theta}(1|1) = \theta(1) - \alpha\hat{\theta}(1|0) - K\left(c\theta(1) + n'(1)\right)$$

$$\tilde{\theta}(1|1) = (1 - Kc)\,\theta(1) - \alpha\hat{\theta}(1|0) - Kn'(1) \tag{3.31}$$

Again we are looking for $E\left[\tilde{\theta}(1|1)\right] = 0$. So after taking expected value of the Equation 3.31 and setting it to zero we get:

$$(1 - Kc)\,E[\theta(1)] = \alpha\hat{\theta}(1|0) \tag{3.32}$$

Clearly Equation 3.32 holds if

$$\alpha = 1 - K\,c$$

Replacing $\alpha = 1 - K\,c$ into Equation 3.31

$$\tilde{\theta}\,(1|1) = (1 - K\,c)\,\theta\,(1) - (1 - K\,c)\,\hat{\theta}\,(1|0) - K\,n'\,(1)$$

$$= (1 - K\,c)\left(\theta\,(1) - \hat{\theta}\,(1|0)\right) - K\,n'\,(1)$$

Now expressing $\tilde{\theta}\,(1|1)$ as a function of $\tilde{\theta}\,(1|0)$ we have

$$\tilde{\theta}\,(1|1) = (1 - Kc)\,\tilde{\theta}\,(1|0) - Kn'\,(1) \tag{3.33}$$

Now our problem is reduced to finding the optimal Kalman coefficient or blending factor K in Equation 3.33 that minimizes the predicted error covariance $P(1|1) = Var\left(\tilde{\theta}\,(1|1)\right)$ with $K(k)$ as decision variable. We could do it using orthogonal principle but this time we do it using the completing-of-squares method.

$$E\left[\tilde{\theta}\,(1|1)\,\tilde{\theta}\,(1|1)\right] = E\left[\left\{(1 - K\,c)\,\tilde{\theta}\,(1|0) - K\,n'\,(1)\right\}\left\{\tilde{\theta}\,(1|0)\,(1 - K\,c) - n'\,(1)\,K\right\}^T\right]$$

$$=(1 - K\,c)\,P\,(1|0)\,(1 - K\,c)^T + K\sigma^2 K$$

Since the transpose of a scalar is the same scalar, the transpose operator can be ignored , thus

$$=P\,(1|0) - K\,c\,P\,(1|0) - P\,(1|0)\,c\,K + K\,c\,P\,(1|0)\,c\,K + K\,\sigma^2\,K$$

$$= P\,(1|0) - KcP\,(1|0) - P\,(1|0)\,c\,K + K\,[cP\,(1|0)\,c + \sigma^2]K$$

$$= P\,(1|0) - KcP\,(1|0) - P\,(1|0)\,c\,K + K\,S\,K$$

where S is the innovation covariance, which is defined as

$$S = cP\,(1|0)\,c + \sigma^2 \tag{3.34}$$

Now we are looking for a square structure, rearranging the above equation we get:

$$= K\,S\,K - P\,(1|0)\,c\,K - K\,c\,P\,(1|0) + P\,(1|0)$$

$$P(1|1) = (K - P(1|0)cS^{-1})S(K - S^{-1}cP(1|0)) + P(1|0) - P(1|0)cS^{-1}cP(1|0) \geq P(1|0) - P(1|0)cS^{-1}cP(1|$$

$$P(1|1) = (K - P(1|0)cS^{-1})S(K - S^{-1}cP(1|0)) + P(1|0) - P(1|0)cS^{-1}cP(1|0) \geq P(1|0) - P(1|0)cS^{-1}cP(1|0) \tag{3.35}$$

$$P(1|1) = (K - P(1|0)cS^{-1})S(K - S^{-1}cP(1|0)) + P(1|0) - P(1|0)cS^{-1}cP(1|0) \geq$$

$$P(1|0) - P(1|0)cS^{-1}cP(1|0) \tag{3.36}$$

Where the Equation 3.36 follows by completing the square. We need to find $K(k)$ such that $P(1|1)$ becomes as small as possible which can be done by choosing $K(k))$ so that $P(1|1)$ decreases by the maximum amount possible at each instant in time. This is accomplish by setting in Equation 3.36

$$K = P\,(1|0)\;c\;\left(c\,P\,(1|0)\;c + \sigma^2\right)^{-1} \tag{3.37}$$

From Equation 3.34 and Equation 3.37 a very simple relation between Kalman gain K and the innovation covariance$S$ can be established as follows

$$K = P\,(1|0)\;c\,S^{-1} \tag{3.38}$$

Once the Kalman gain $K(k)$ has been determined it is straightforward to obtained the updated estimate covariance $P(1|1) = Var(\tilde{\theta}(1|1))$, from Equation 3.36

$$P\,(1|1) = P\,(1|0) - P\,(1|0)\,c\left(cP\,(1|0)\,c + \sigma^2\right)^{-1} cP\,(1|0) \tag{3.39}$$

Equation 3.39 can be written in a more compact form as

$$P\,(1|1) = P\,(1|0)\,(1 - K\,c) \tag{3.40}$$

We use Equation 3.40 in our MATLAB simulation.

Having determined the Kalman gain $K(k)$ we can now incorporate the first measurement $\psi(1)$. Replacing K and $\alpha = 1 - K\,c$ into the equation Equation 3.30 we obtain the updated state estimate as

$$\hat{\theta}\,(1|1) = (1 - Kc)\,\hat{\theta}\,(1|0) + K\psi\,(1) \tag{3.41}$$

$$\hat{\theta}\,(1|1) = \hat{\theta}\,(1|0) + K\left(c\,\theta\,(1) - c\hat{\theta}\,(1|0)\right) \tag{3.42}$$

where the first term $\hat{\theta}\,(1|0)$ in Equation 3.42 is the old information and the difference between $\psi(1) = c\,\theta(1)$ and $c\hat{\theta}\,(1|0)$ is the new information which is called innovation or the residual. The innovation reflects the discrepancy between the predicted measurement $c\hat{\theta}\,(1|0)$ and the actual measurement $\psi\,(1) = c\,\theta\,(1)$.

We can now proceed with $\hat{\theta}\,(2|1)$, $\hat{\theta}\,(2|2), \ldots$ using $\psi\,(1)$, $\psi\,(2)$, ... respectively. The above derivation contains the essence of the scalar Kalman filter, the only change required for the more general case being the replacement of the scalar $\theta(k)$ by a vector state and with the covariance becoming matrices whose reciprocals are inverses and one must keep track of the order of operations.

3.3.3 Kalman filter smoother

There are basically three types of smoothing filtering which need to be distinguished: Fixed point smoothing, Fixed-lag smoothing and Fixed interval smoothing which in turn has two main recursive approaches namely Forward-backward algorithm and the Two-pass smoother also known as RTS (Rauch-Tung-Striebel) smoother. It was decided that the best type of smoothing filtering for this research was RTS smoother because of its proven accuracy.

The Fixed interval smoothing algorithm recalculates each estimate generated by its associated Kalman filter based on information obtained over the entire interval of data being analyzed. In this sense, it is useful only as a post data analysis tool, since the entire set of data over the given interval must be known and Kalman filter estimates and variances of error between estimates and observations must be generate previously. In Fixed interval smoothing, k is available and N is fixed. This corresponds to the situation where one collects some experimental data and then derives estimates of the state subsequent to the data collection.

Problem statement: the fixed-interval smoothing problem consists of estimating

$$\hat{\theta}\,(k|T) = E\left[\theta\,(k)\,|z(1)....z\,(N)\right] \tag{3.43}$$

for fixed N and for all k in the interval $1 \leq k \leq N$.

The smoothed estimate improves the standard estimates by adding future measurements. It is relevant for off-line estimation problems. In a recursive context, Fixed-interval smoothing is concerned with the smoothing of a finite set of data i.e. with obtaining $\hat{\theta}_{k|N}$ for fixed N and all k in the interval $k = 0,...N$. In this off-line filtering situation, we have access to N measurements and want to find the best possible state estimate $\hat{\theta}_{k|N}$.

The Two-pass smoother or RTS smoother approach requires the standard Kalman estimate and covariance to be computed in a forward pass and the smoothed quantities are then computed in the backward pass. The equations for the Fixed-interval smoothing algorithm used in this applications where obtained from [20] . Several sources where beneficial in understanding these equations . An excellent derivation of it can be found in [19] . Here the equations are given without proof. The RTS algorithm is the following:

Given the available observation $y\,(k) = 0,...,N$, run the standard Kalman filter and store both the time and measurement updates, $\hat{\theta}\,(k|k)$, $\hat{\theta}\,(k|k-1)$, $P\,(k|k)$, $P\,(k|k-1)$. Then apply the following time recursion backwards in time:

Gain for Kalman smoothing filtering,

$$K_{smooth} = P\left(k-1|k-1\right)a_{k-1}P^{-1}\left(k|k-1\right) \tag{3.44}$$

$$\hat{\theta}\left(k-1|N\right) = \hat{\theta}\left(k-1|k-1\right) + K_{smooth}\left(\hat{\theta}\left(k|N\right) - \hat{\theta}\left(k|k-1\right)\right) \tag{3.45}$$

The covariance matrix of the estimation error $P\left(k|N\right)$ is

$$P\left(k-1|N\right) = P\left(k-1|k-1\right) + K_{smooth}\left(P\left(k|N\right) - P\left(k|k-1\right)P^{-1}\left(k|k-1\right)P^{-1}\left(k|k-1\right)a_{k-1}P\left(k-1|k-1\right)\right) \tag{3.46}$$

This algorithm is simple to implement in MATLAB but it requires that all the states and covariances for both the prediction and filter errors are stored. The MATLAB implementation of both Kalman filter and Kalman smoother are available in Appendix E.

3.3.4 Summary of key equations for Kalman filter

At this point it is worth summarizing the equations which underlay the Kalman filter algorithm for scalar case. They will be useful when analyzing the simulation results.

STATE SPACE MODEL

$$\theta\left(k\right) = a\,\theta\left(k-1\right) + w\left(k\right) \tag{3.47}$$

$$\psi\left(k\right) = c\,\theta\left(k\right) + v(k) \tag{3.48}$$

$$E\left[v\left(k\right)\right] = 0, \quad E\left[n'\left(k\right)\right] = 0$$

$$Var[v(k)] = \sigma_{\theta}^{2}, \quad Var[n'(k)] = \sigma^{2}.$$

Initial

$$\hat{\theta}(0|0) = \theta_0 \text{ and } P(0|0) = P_0$$

PREDICTION STEP: This step predicts the state and variance at time k dependent on the information at time $k-1$

$$\hat{\theta}(k|k-1) = a\,\theta\left(k-1|k-1\right) \tag{3.49}$$

$$P\left(k||k-1\right) = aP(k-1|k-1)a^{T} + \sigma_{\theta}^{2} \tag{3.50}$$

KALMAN GAIN

$$K\,(k) = P\,(k|k-1)\,c^T\,(cP(k|k-1)c^T + \sigma^2)^{-1} \tag{3.51}$$

UPDATE STEP

$$\hat{\theta}(k|k) = \hat{\theta}(k|k-1) + K(k)(\psi(k) - c\hat{\theta}(k|k-1)) \tag{3.52}$$

$$P\,(k|k) = P(k|k-1) - P(k|k-1)c^T\,(cP(k|k-1)c^T + \sigma^2)^{-1}cP\,(k|k-1) \tag{3.53}$$

In compact form

$$P\,(k|k) = P\,(k|k-1)\,(1 - K(k)\,c\,P\,(k|k-1)) \tag{3.54}$$

3.3.5 Tuning scalar Kalman filter for performance

The performance of the Kalman filter is uniquely determined by the values of the phase noise variance σ_θ^2 and additive noise variance σ^2 which are the required inputs to the Kalman filter. As we can see from Equation 3.50 and Equation 3.51 , the Kalman gain increases with σ_θ^2 and decreases with σ^2, thus the convergence properties are dependent on the relative magnitudes of the phase noise variance and the additive noise variance. An intuitive understanding of the importance of these quantities may be acquired by the following argument. Looking at Kalman gain Equation 3.51 we see that as the additive noise variance σ^2 approaches zero, the Kalman gain $K(k)$ approaches c^{-1}, specifically

$$lim_{\sigma^2 \to 0} K(k) = c^{-1} \tag{3.55}$$

Consequently, the Kalman gain weights the residual $\psi(k) - c\hat{\theta}(k|k-1)$, that reflects the discrepancy between the predicted measurement $c\hat{\theta}(k|k-1)$ and the actual measurement $\psi(k)$, in Equation 3.52 more heavily. Next, let us consider the case when σ^2 becomes infinity. As σ^2 approaches infinity , Kalman gain $K(k)$ approaches 0 and $\hat{\theta}(k|k)$= $\hat{\theta}(k|k-1)$, specifically

$$lim_{\sigma^2 \to \infty} K(k) = 0 \tag{3.56}$$

and the updated state estimate $\hat{\theta}(k|k)$ becomes equal to the predicted state estimate $\hat{\theta}(k|k-1)$. On the other hand, from Equation 3.51, as the predicted error covariance $P(k|k-1)$ approaches zero, the Kalman gain $K(k)$ weights the residual less heavily, specifically

$$lim_{P(k|k-1) \to 0} K(k) = 0 \tag{3.57}$$

Now we will use the above results to make a key observation about the weighting by Kalman coefficient that give insight in applications. Let's rearrange the Equation 3.52, we get

$$\hat{\theta}(k|k) = \hat{\theta}(k|k-1)(1-K(k)c) + K(k)\psi(k) \tag{3.58}$$

Next, as σ^2 approaches zero then $K(k) = c^{-1}$, according to 3.55, and the term inside the parentheses in Equation 3.58 vanishes. Thus, we obtain the conclusion: as σ^2 approaches zero the actual observation $\psi(k)$ is trusted more and more and the predicted measurement $c\hat{x}(k|k-1)$ is trusted less and less.

On the other hand, as the prediction error covariance $P(k|k-1)$, which decreases with σ_θ^2 according to Equation 3.50, approaches zero, the Kalman gain $K(k)$ will go to zero according to the result 3.57 and the actual measurement $\psi(k)$ is trusted less and less while the predicted measurement $c\hat{x}(k|k-1)$is trusted more and more .

The discussion on Kalman filter tuning can be summarized as follows:

- Large variation of the noise in the process model i.e. high σ_θ^2 means that the prediction, according to the process model, is likely to be less accurate and the new measurement should be weighted heavier.
- Large variation in the measurement noise i.e. high σ^2 means that the new measurement is likely to be less accurate and the prediction should be weighted heavier. The other way around, less uncertainty in the observations i.e low σ^2 means we rely more on the observations.

For the purpose of simulation it is worth mentioning here a couple of key properties of the Kalman filter related to its performance. At some time the Kalman filter has to be given an initial value $\theta(0)$. The best $\theta(0)$ is probably the one equal to the unknown $\theta(k)$ of the system being filtered. However one must reckon with the possibility that an inappropriate initial $\theta(0)$ for the Kalman filter is selected. The point is that any damage caused by the inappropriate selection is forgotten exponentially fast. Another property is that the best estimate of $\theta(k)$ at the particular time k depends in an exponentially decaying fashion on prior measurements [3].

There is a final point worth mentioning on Kalman smoother related to its practical implementation. In [24] a new approach allowing the phase noise to be estimated by both prediction and smoothing filter that optimally estimate the phase noise in the minimum mean-squared error MMSE sense is proposed. It consists in modifying the receiver analog front-end by adding an additional signal path directly from the oscillator so as to provide a nondata modulated observation of the phase noise. The hardware overhead of introducing this additional path in the analog front-end, according to [24] , is modest, especially in the integrated circuit.

3.3.6 Kalman filter simulation

Typically Kalman filter assumes that the phase noise variance σ_θ^2 is known in advance. However this is usually not the case. Since we don't know the actual noise variance, it must be set by the user before the Kalman filter starts. The principle of selection is that a small σ_θ^2 should be chosen in order to obtain relatively small variances in the estimate state variables after the algorithm converges. However, this also implies that the algorithm will take a longer time to converge as compared with that of using a larger σ_θ^2.

The simulation of both scalar Kalman filter and Kalman filter smoother was performed in MATLAB environment for the following initial values: phase noise variance $\sigma_\theta^2 = 10^{-6}$, additive noise variance $\sigma^2 = 10^{-2}$, the initial condition on the state θ_0 i.e initial value for true $\theta(k)$ was set to 1.2 , the initial guess for $\theta(k)$, labeled θ_{-0} , is random and in this particular case θ_{-0}=1.3147 and initial guess for covariance $P_{-0} = 0.01$.

Fig. 3.9 shows the effect of filtering the data. Note that the Kalman filtered signal $\hat{\theta}(k)$ (green solid line) and the smoothed signal $\hat{\theta}_{smooth}(k)$ (black solid line) follow the true value $\theta(k)$ quite closely. Consequently, we make the following observation: the updated state estimate $\hat{\theta}(k+1|k+1)$ tends to follow the true value $\theta(k)$ quite closely.

The plot of the Kalman filter gain is shown in Fig. 3.11, it drops as the updated estimate is trusted more. Fig. 3.12 shows the covariance associated with the Kalman smoothing algorithm (black solid line) together with the covariance achieved by the standard Kalman filter (read solid line). From the plot, it can be seen that their initial values are large but they quickly drop as the length of simulation k gets larger . As expected, the value of the smoothed variance is a definite improvement over the standard Kalman filter result. We note the steady state behavior in the middle of the plot. This happen often when the data span is large and the process is stationary [20]. In the steady state region, the gains of the filter and the smoother and the error variance are constant .

Finally, a couple of interesting observations can be made about the length of simulation k. From the Fig. 3.11 , as the number of trials k gets larger, the Kalman gain $K(k)$ tends to a steady state and so does the updated error covariance $P(k|k)$ as shown in Fig. 3.12. Consequently, the following conclusion can be drawn: As $k \to \infty$, Kalman filter reaches steady state and becomes a linear time-invariant filter i.e $K(k)$ constant and $P(k|k)$constant.

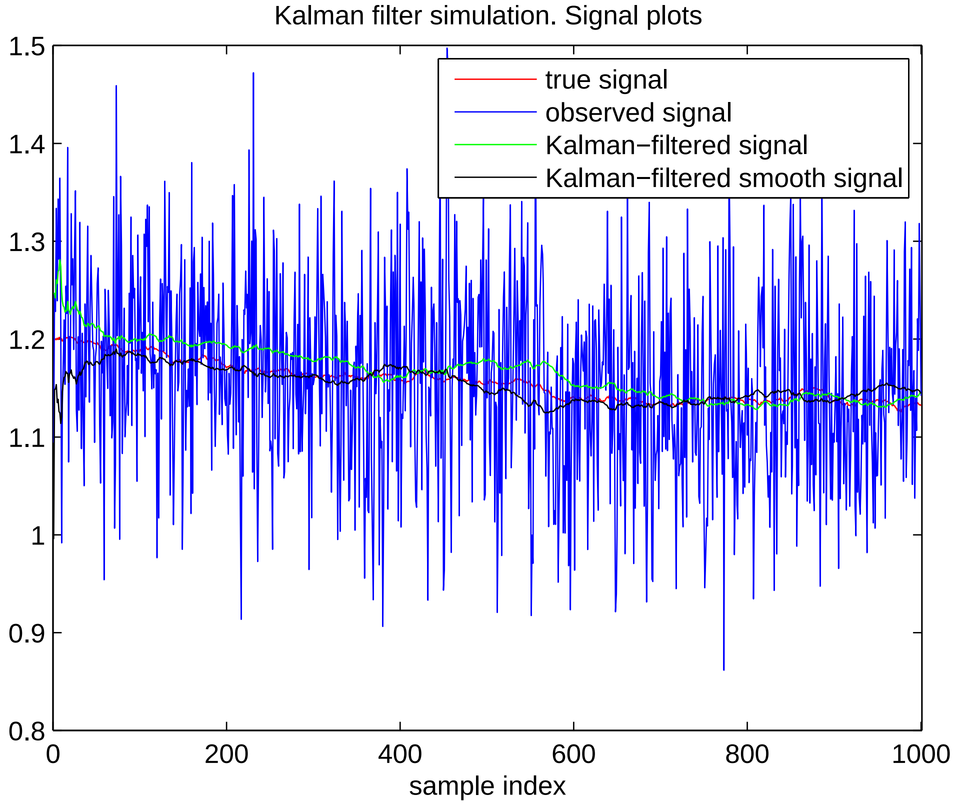

Figure 3.9: Kalman filter simulation, true signal θ_k , observed signal ψ_k, Kalman-filtered signal $\hat{\theta}_k$, Kalman-filtered smooth signal $\hat{\theta}_{k\_smooth}$

3.3.7 Performance evaluation of Kalman filter

In this section a simple consistency check between the Kalman smoother estimate versus and unfiltered estimate is provided. It can be used not only as a verification procedure for the filtering correctness, but also as a approach for making trade-off in designing a suitable Kalman filter.

We begin by defining the quality of the filtered estimate V_k as the mean squared error of the Kalman smoother phase noise $\hat{\theta}_k$ with respect to the true phase noise θ_k as given by the equation 3.59.

$$V_k = E\left[\left(\hat{\theta}_k - \theta_k\right)^2\right] \tag{3.59}$$

We also define the quality of the unfiltered estimated S_k as the mean squared error of the soft phase estimate ψ_k with respect to the true phase noise θ_k as given by the Eq. 3.60

$$S_k = E\left[\left(\psi_k - \theta_k\right)^2\right] \tag{3.60}$$

Parameters V_k and S_k are of the direct interest because they provide a simple relationship that ensure us the correctness of the Kalman filter design by verifying the following inequality

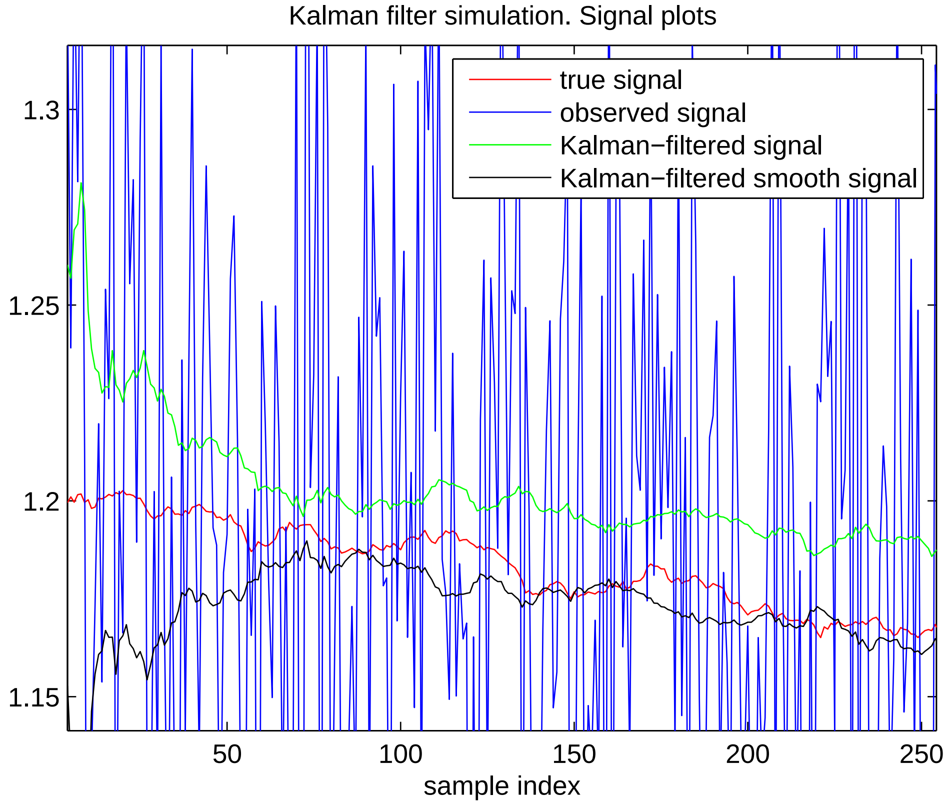

Figure 3.10: Kalman filter simulation. This is a zoomed in version of the Fig. 3.9

$$S_k > V_k \tag{3.61}$$

Fig. 3.13 plots vector V_k and vector S_k against nine different values for the phase noise variance σ_θ^2 and channel noise variance σ^2 denoted in MATLAB by Q and R respectively. The length of the simulation was set equal to 10000. All vectors in the simulation namely V_k, S_k, σ_k^2 and σ^2 are 1x9 row vectors whose values are increasing by a factor of 100.

There is a striking feature of the bottom two panels of Fig. 3.13. As shown in the bottom right plot, no matter how accurate the process model may be (i.e whether σ_θ^2 is small or large), the value of the quality of the unfiltered estimates in S_k can only be changed by varying the value of σ^2. Similarly, in the bottom left plot the value of S_k is independent of σ_θ^2 and its value changes by varying σ^2. That make a lot of sense since the effect of varying the variance measurement process or channel noise variance σ^2 effect primarily on the spread of measurement from the real value as they show up in the Kalman filter formulas.

The bottom two plots of Fig. 3.13 depict Q and R versus the quality of the filtered estimate V. From both graphs, we can clearly see that the quality of the filtered estimates deteriorates as the statics of the process noises and measurement noises increase and vice verse, which is consistent with the theory. It is immediately apparent that the inequality 3.61is fulfilled and

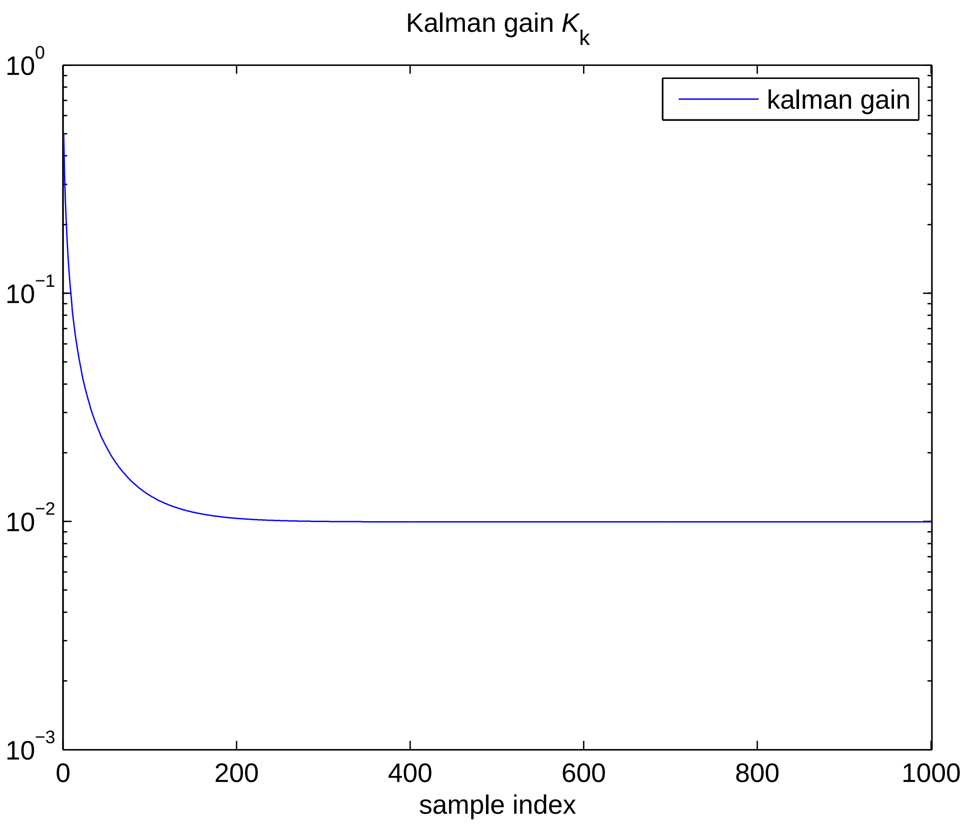

Figure 3.11: Kalman gain.

thereby the filtering correctness after comparing the two top graphs with the two bottom graphs of the Fig. 3.13.

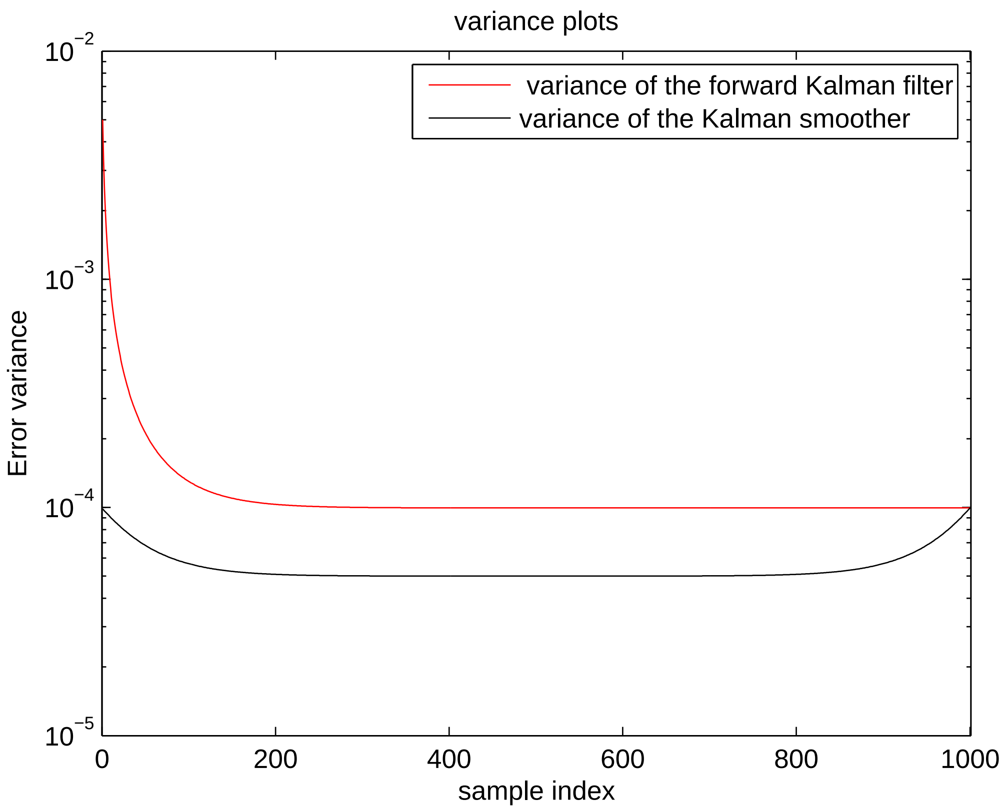

Figure 3.12: Standard Kalman filter and Kalman filter smoother covariances. Notice the reduction of the covariance by smoothing

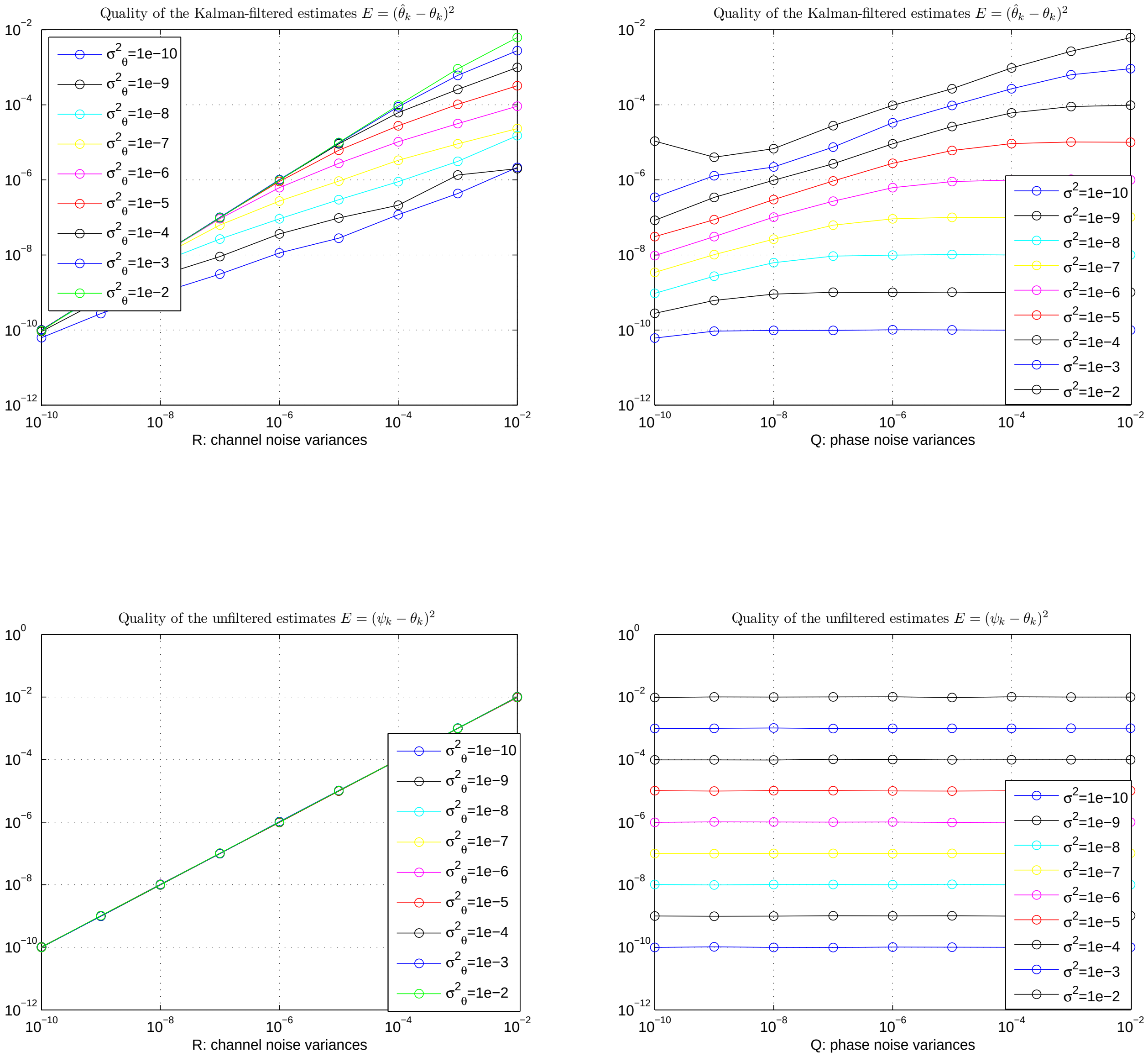

Figure 3.13: Kalman filter, tradeoff analysis. Quality of the filtered estimate, $V_k = E\{(\hat{\theta}_k - \theta_k)\}$. Quality of the unfiltered estimate, $S_k = E\{(\psi_k - \theta_k)^2\}$. R, row vector of channel noise variances σ^2 . Q, row vector of phase noise variances σ_θ^2.

Chapter 4

Experimental results and performance evaluation

This chapter focus on analyzing the FF carrier phase estimation algorithm developed in chapter 3. The estimator's performance is simulated and evaluated for a QAM modulation scheme. The mean square error MSE between the filtered estimates of both filters and the true phase noise was computed for comparison purpose in a similar way to those presenting in subsection 3.3.7, for performance evaluation of the Kalman filter.

4.1 Simulation model

The main design objective of this project is to implement and validate a phase noise estimator using linear filters to reduce the error variance of the phase noise obtained after the unwrapping stage. To achieve this objective, FIR Wiener filter, standard Kalman and Kalman smoother were designed. The detailed setup block diagram of the stages of the algorithm for the carrier phase estimator is illustrated in Fig. 4.1.

The phase noise variance σ_θ^2 and the additive noise variance σ^2 are two design parameters used to tune both Wiener and Kalman filter. In addition to that, Wiener filtering requires two extra inputs, namely L and $delta$. The phase noise variance σ_θ^2 is unknown as we typically do not have the ability to directly observe the process we are estimating. Sometimes a relatively poor process model can produce acceptable results if one injects enough uncertainty into the process via the selection of σ_θ^2 . Certainly in this case one would hope that the process measurements are reliable. For this project the level of reliability, measured by SER, should be less than 10^{-3}. It is achieved by choosing $\sigma^2 = 0.021$ in the SER versus SNR diagram. Here we set the phase noise covariance σ_θ^2 to a value of $1e-5$.

In MATLAB simulation, the Kalman filter is initialized to a suitable random value by guessing the initial true phase noise and its variance. To make sure that the estimates converges quickly and the influence of the initial guess soon will be negligible, we make the value of he variance of the initial true phase noise large enough.

Greek variable names used in the text and their correspondent representation in MATLAB environment are presented in Table 4.1.

Greek character variable name	Variable name in MATLAB	meaning
σ_θ^2	sigma2_theta	Phase noise variance
σ^2	sigma2	Additive noise variance
θ_k	theta_true	True phase noise
$\hat{\theta}_k^r$	theta_hat_raw	Raw phase noise estimate
$\hat{\theta}_k^u$	theta_hat_unwrapped	Unwrapped phase noise estimate
ψ_k	psi	Soft phase estimate
$\hat{\theta}_k$	theta_hat	Kalman-filtered phase noise
$\hat{\theta}_k$	theta_Hat	Wiener-filtered phase noise
Δ	delay	Wiener filter delay

Table 4.1: Greek characters and their variable names in MATLAB

During this project, several MATLAB m.files were written, they all are available in the Appendix E. To make the description of the block diagram in Fig. 4.1 more understandable, the block names has the same name as the function name in MATLAB. A brief explanation of all MATLAB scripts is provided below.

1. main_program_4QAM.m. This is the main program of the project which calls below functions as needed.

2. generating_4QAM.m, generates a series of randomly picked-up QAM symbols.
   input arguments: N: length of the transmitted symbol sequence , Es_No_dB: SNR, M: number of symbols, $M = 2$ for 4-QAM, $M = 4$ for 16-QAM
   output arguments: a: transmitted QAM symbols, n: additive noise variance, σ^2

3. true_theta.m , returns theta true θ_k
   input arguments: N: length of the transmitted symbol sequence, phase noise variance σ_θ^2
   output argument: θ_k.

4. viterbi.m, generates the raw phase noise estimate $\hat{\theta}_k^r$
   input arguments: r: received signal, M_s: length of the block
   output argument: $\hat{\theta}_k^r$

5. unwrapping.m, generates the unwrapped phase noise estimate , $\hat{\theta}_k$
   inputs argument: $\hat{\theta}_k^r$: raw phase noise estimate
   output argument: $\hat{\theta}_k$

6. kalman_filter.m, produces the Kalman-filtered phase noise $\hat{\theta}_k$ using Kalman filter smoother
   input arguments: ψ_k, σ_θ^2, σ^2, N: length of the transmitted symbols
   output argument: $\hat{\theta}_k$

7. wiener_filter.m, produces the Wiener-filtered phase noise $\hat{\theta}_k$.
   input arguments: ψ_k, σ_θ^2 , σ^2, L: number of taps: , Δ: filter delay
   output argument: $\hat{\theta}_k$.

The big picture of the feedforward carrier phase estimation algorithm is shown in Fig. 4.1. Note that Kalman initial condition is not considered in the picture. A main goal of the simulation is to obtain the filtered estimate of the phase noise using Kalman smoother and FIR Wiener filters. A comparative assessment of the performance of Kalman smoother and Wiener filters can be achieved by comparing the quality of the filtered estimate with the quality of unfiltered estimates of both filters. Thus, the performance of Kalman and Wiener filters is evaluated by the quality of their phase noise estimation which is determined by the steady-state variance given by the Eq. 4.1

$$V_k = E\left[\left(\hat{\theta}_k - \theta_k\right)^2\right] \tag{4.1}$$

where $\hat{\theta}_k$ is the filtered output of either Kalman or Wiener filter and θ_k is the true phase noise. The quality of the unfiltered estimate is in turn calculated by Eq. 4.2

$$S_k = E\left[\left(\psi_k - \theta_k\right)^2\right] \tag{4.2}$$

where ψ_k soft phase estimate and θ_k is the true phase noise.

The actual simulation was carried out in MATLAB. Except for the block with no label on it, shown at the upper right side of the block diagram in the Figure 4.1, each block has a label indicating the name of the block which is also the MATLAB function name in the simulation. The inputs to a block that influence its functioning can be a parameter or input data.

From Fig. 4.1 the parameters M , Es_No_dB and N are applied directly to the block generating_4QAM which in turn generates a series of randomly picked-up 4-QAM transmitted symbols labeled a. The block generating_4QAM returns also the additive noise variance σ^2 or sigma2 in MATLAB, which is calculated considering the SER constraint that should be less than 10^{-3} , it also returns the channel noise n. The block true_theta with inputs phase noise variance σ_θ^2 and N , outputs true theta θ_k that is theta_true in MATLAB simulation. The additive noise variance σ^2 along with a and n and the true theta θ_k are feed into the block in the upper right part of the Figure 4.1 which generates the received signal r. This input as well as θ_k are directly feed into the block viterbi that returns the raw phase noise estimate $\hat{\theta}_k^r$. The block unwrapping takes $\hat{\theta}_k^r$ and produces the unwrapped output ψ_k . The last step in the simulation process is the optimal filter taking the form of FIR Wiener filter or Kalman smoother. The inputs σ^2, σ_θ^2 and ψ_k are common to both FIR Wiener and Kalman smoother. Wiener filter required additionally two inputs, the number of taps L and the filter delay Δ. The output obtained from both filters can be compared in Table 4.3 . For simplicity, in Fig. 4.1, the argument N, length of the transmitted sequence has been ignored.

Table 4.2 presents the parameter setting for the simulation of the FF carrier phase estimation.

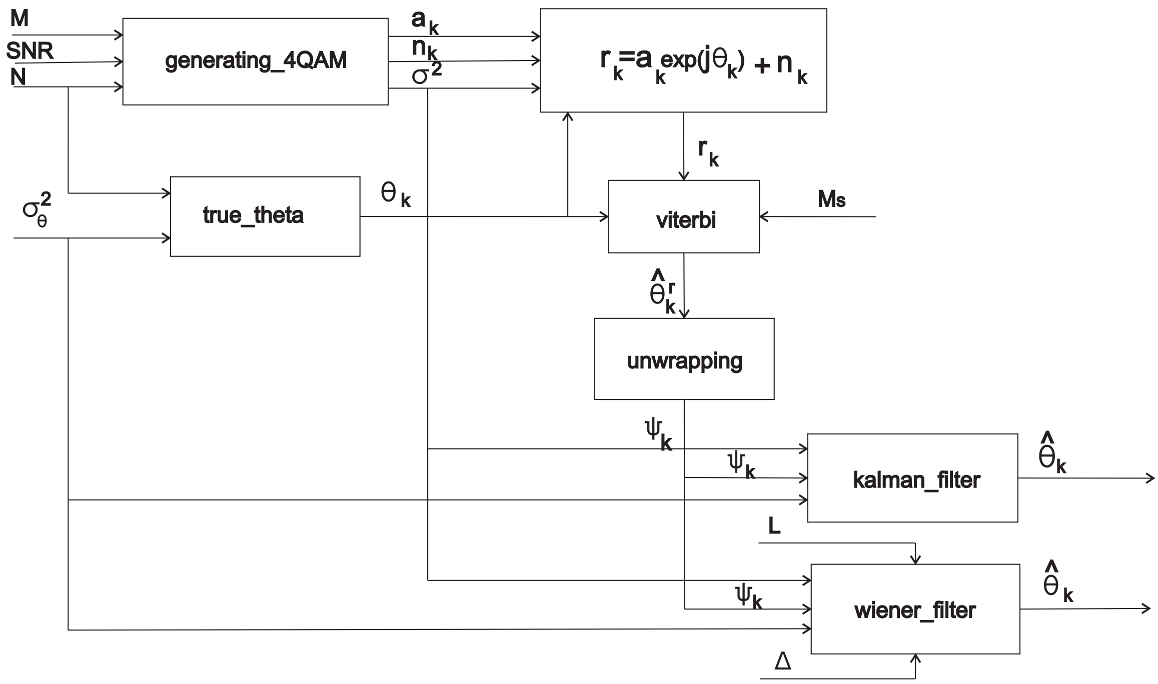

Figure 4.1: Diagram of the MATLAB simulation setup for FF carrier phase estimation algorithm. Note the initialization of the Kalman filter is not considered in this picture.

We compare the performance of both Kalman smoother and Wiener filter and their results are shown in Table 4.3. The results presented in the Table 4.3 display eight outcomes from their respective simulation runs, carried out for identical parameter settings as presented in Table 4.2.

We cutoff the first 50 indexes of ψ_k in Kalman and Wiener simulation respectively, to avoid the initial condition problem that shows up when comparing the performance of two different filters. As can be seen from the data in Table 4.3. Wiener-filtered estimates are slightly smaller than Kalman-filtered estimates and the unfiltered phase estimates respectively. Surprisingly, there is no significant differences between Kalman-filtered and unfiltered estimates as can be

Parameter	Value	Notation
N	10000	Length of the transmitted symbol sequence
N_s	32	Length of the block of the Viterbi and Viterbi estimator
M	4	Number of symbols for 16-QAM
Es_No_dB	17.5	SNR
BER	1e-3	Bit error rate
σ_θ^2	1e-5	Phase noise covariance
σ^2	0.0178	Additive noise covariance
L	51	Wiener filter length
Δ	20	Wiener filter delay

Table 4.2: Parameter setting simulation. The additive noise covariance σ^2 is determined by SER

observed from the first and third columns. This is counter to analytical expectations as we would expect that the filtered estimates to be much less than unfiltered estimates

Kalman-filtered estimateV_k	Wiener-filtered estimateV_k	Quality of the unfiltered estimate S_k
0.6151	1.084	0.6203
0.6127	0.3221	0.6290
0.6166	0.2756	0.6230
0.6072	0.1973	0.6227
5.5274	3.9221	2.2283
0.6176	0.9021	0.6151
5.5516	5.1108	5.5603
5.5378	5.6562	5.5577

Table 4.3: Quality of the Kalman and Wiener filter estimates

MATLAB plot simulations from phase noise estimation are shown in Fig. 4.2. From the top-left plot, it can be seen that the discontinuity of θ_k^r appears when an extreme value $\frac{\pi}{2}$ is reached, the raw phase estimate θ_k^r then jumps to the other end of the interval $-\frac{\pi}{2}$ and vice verse. As explained above, reconstructing the physically continuous phase noise can be done by adding or subtracting multiples of $\frac{\pi}{2}$ and thereby suppressing the phase noise jumps as can be observed in the top-right plot in Fig. 4.2. However, difficulties arise in MATLAB simulation due to cycle slips arising from ambiguities in phase unwrapping. The occurrence of cycle slips, refers to jumps by $\pm\pi/2$, breaks the continuity of ψ_k as can be observed at index 1800. So, the right way to compute the quality of the unfiltered estimate is to detect the position of the cycle slips and then compute the MSE of the data set determined by two contiguous cycle slips without considering them.

The bottom-left plot shows for comparison purpose the true phase noise θ_k and soft or unwrapped phase noise ψ_k. The most important simulation result is shown in the next plot, from

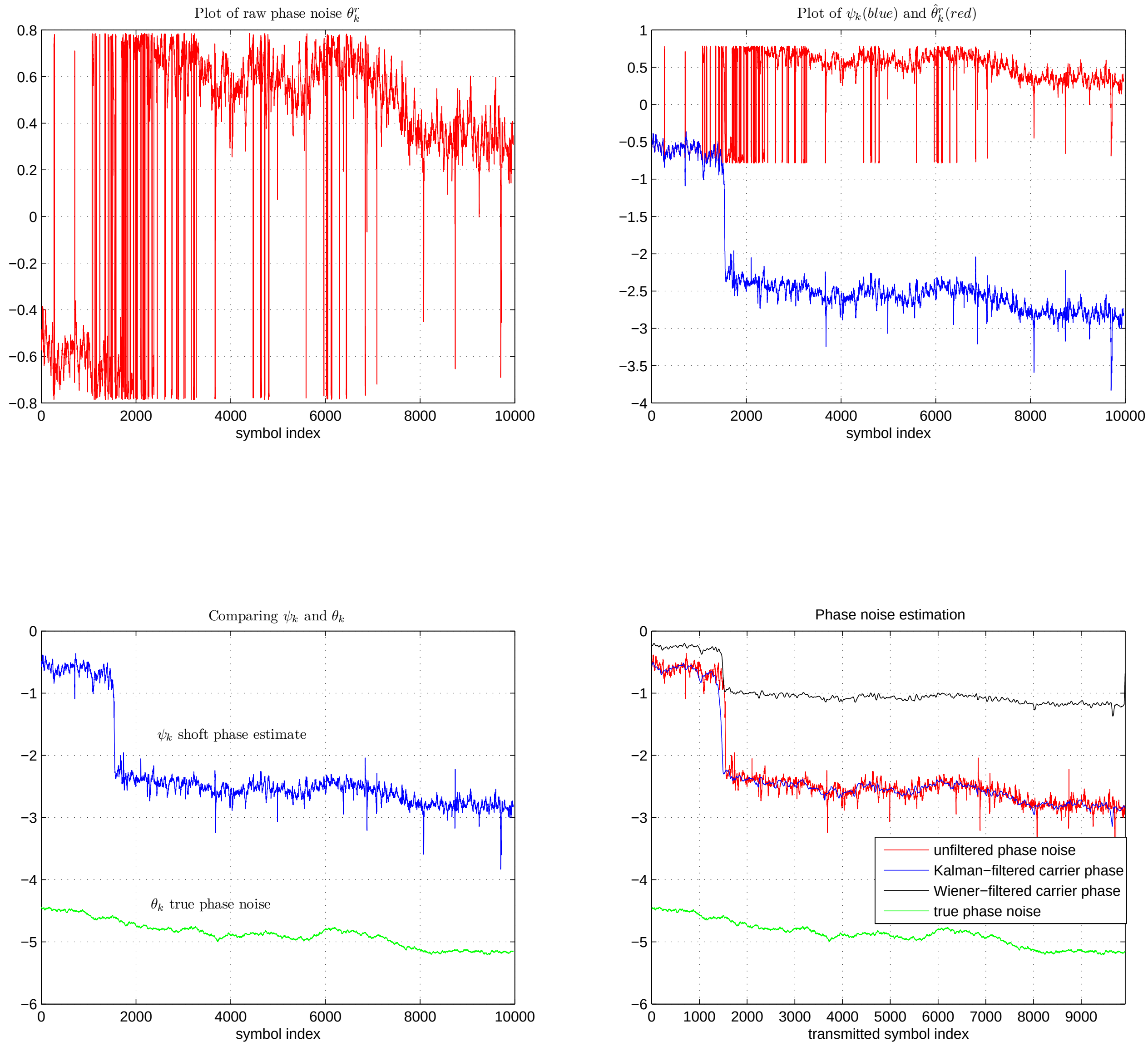

Figure 4.2: MATLAB simulation results of phase noise estimation.

which it is apparent that, in this particular run simulation, Kalman filter will produce a bit better estimate than Wiener filter.

From the simulation results, we can see that the unfiltered phase noise ψ_k is too noisy. Usually it is necessary to fine-tune the Kalman filter by tuning σ_θ^2 to avoid noisy estimates (the larger σ_θ^2 the larger the Kalman gain $K(k)$ and the stronger the updating of the estimates). However, since both σ_θ^2 and σ^2 are considered fixed input parameters in this project, we should consider another way to obtained a more accurate unfiltered phase noise. A possible solution is to use a better Viterbi and Viterbi phase estimator that could generate a more accurate rough phase estimate $\hat{\theta}_k^r$ of the carrier phase as the quality of the filtered estimate depend on the accuracy of the first stage estimator which can be significantly lower than the symmetry angle of the QAM constellation. Further proposals to improve it can be founded in Chapter 5.

Chapter 5

Concluding comments and suggestions for further work

The work described in this thesis was motivated by the desire of improving the feedforward phase noise estimates obtained after unwrapping for a QAM modulation scheme by reducing their MSE . Linear filters optimal in the MSE sense such as Kalman filter, Kalman smoother and Wiener filter were designed and tested for this purpose. Simulation results were carried out in MATLAB for a 4-QAM modulation scheme which can be easily modify and use for16-QAM.

The MSE was the criterion used to assess the quality of the filtered estimates. Surprisingly, Wiener filter was found to produce a bit better phase noise filtering results than Kalman filter which appears inconsistent with those of the Kalman and Wiener simulation results obtained in Chapter 3. Moreover, there was no significant difference between Kalman-filtered and unfiltered estimates. These results suggest that a more accurate estimation of the phase noise previous filtering, which could produce a more accurate estimate, need to be implemented.

Finally, it is recommended that future work, that may lead to an improved simulation result, should include the following:

- Include the effect of cycle slips,
- Consider new estimators,
- Extension to dual polarization
- Initialization of the Kalman filter

Bibliography

[1] A. Mehrotra Alper Demir and Jaijeet Roychowdhury. Phase noise in oscillators: A unifying theory and numerical methods for characterization. *Bell Laboratories. Murray Hills*.

[2] Amit Mehrotra Alper Demir and Jaijeet Roychowdhury. Phase noise oscillators: A unifying theory and numerical methods for characterization. *IEEE Transactions on Circuit and Systems, Fundametal Theory and Applications*, 47:655–674, 2000.

[3] Brian D. O. Anderson. From wiener to hidden markov models. 199.

[4] Daniel J.F. Barros Joseph M. Khan Ezra Ip, Alan Pak Tao Lau. Coherent detection in optical fiber systems. *Optics Express*, 16,:753–791, 2008.

[5] A. Sedar et al Fangrong Peng, Henk Wymeersch. Operational regime of symbol-by-symbol phase noise estimation for polmux 16-qam. *Optical Communication (ECOC)*, 2010.

[6] Ezra Ip and Joseph Kahn. Feedforward carrier recovery for coherent optical communications. *Journal of Lightwave Technology*, 25, 22007.

[7] D. Manolakis J.G. Proakis. *Dgital Signal Processing: Principles, Algorithms and Applications*. 1995.

[8] kazuro Kikuchi. Coherent optical communications - history, state-of-the-art technologies and challenges for the future.

[9] Nakazawa Kikuchi and Tetsuya. *High Spectral Density Optical Communication Technologies. Chapter 2*. Springer, 2010.

[10] Guifang Li. Recent advances in coherent optical communication. *Advances in Optics and Photonics*, 2009.

[11] I. Darwaseh L.M. Pessoa, H.M. Salgado. Efficient implementation of a phase estimation algorithm in coherent optical system.

[12] I. Darwaseh L.M. Pessoa, H.M. Salgado. Joint mitigation for optic impairments and phase estimation in coherent optical systems. pages 169–170, 2008.

[13] R. Lopez-Valcarcel. Cost minimization interpretation of fourth power phase estimator and links to multimodulus algorithm. *Electronic Letters*, 40, No 4, 2004.

[14] Robert H Morelos-Zaragoza. Feedforward carrier and symbol timing recovery and phase-invariant signal for software-defined radio. 2003.

[15] Georgios Ellina Neophytos Antoniades. *WDM Systems and Networks. Modelling, Simulation, Design and Engineering*. Springer, 2012.

[16] Reinhold Noé. Phase noise-tolerant synchronous qpsk/bpsk baseband-type intradyne receiver concept with feedforward carrier recovery. *IEE Journal of Lightwave Technology*, 32:802–808, 2005.

[17] Ara Patapoutian. On phase-locked loops and kalman filters. *IEEE Transactions on Communications*, 47, NO 5:670–673, 1999.

[18] Willian Shieh and Keang-Po Ho. Equalization-enhanced phase noise for coherent detection systems using digital signal processing. *Optical Society of America*, 2008.

[19] N. Shimking. Lectures notes, fall 2009. department of electrical engineering. estimation and identification in dynamical systems. lecture 7. *Israel Institute of Technology*, 2009.

[20] Gilbert Strang and Kei Borre. *Linear Algebra, Geodesy and GPS*. Wellesley Cambridge Press, 1997.

[21] Hissao Nakashima Takeshi Hoshida and Jens Rasmussen. Digital coherent receiver technology for 100/gb/s optical transport systems. *Fujitsu*, 46:63–71, 22009.

[22] M.G. Taylor. Coherent detection method using dsp for demodulation of signal and subsequent equalization of propagation impairments. *Photonics Technology Letters IEEE*, 16, issue 2:674–676, 2004.

[23] Chongjin Xie. Loca oscillator phase noise inducted penalties in optical coherent detection systems using electronic chromatic dispersion compensation. *Optical Society of America*, 2009.

[24] Liang Zhao. A novel phase-noise compensation scheme for communication receivers. *IEEE Transactions on Communication Technology*, 54, 2006.

Appendix A

The input-output PSD relation in linear systems with random signals.

Consider a stationary process $y(k)$ which by the Wold's decomposition theorem can be expressed as

$$y(k) = h_k(\theta) * n(k) \tag{A.1}$$

for some linear filter $h_k(\theta)$ where θ are the unknown parameters in h_k and $n(k)$ is a white noise process with spectrum Power spectral density$\Phi_{uu}(\omega)$. We know that if $n(k)$ is a discrete time white noise SSP then$\Phi_{uu}(\omega)$ =σ_n^2 where σ_n^2 is the covariance of $n(k)$. Since white noise $n(k)$ is stationary then

$$r_Y(k) = \sum_{l=-\infty}^{\infty} \sum_{p=-\infty}^{\infty} h(l)h(p)r_N(k+p-l) \tag{A.2}$$

Equation A.2 may seem to lead to very complicated computations. But fortunately Equation A.2 gives a very simple relation between power spectrum of input and output signal. Thus PSD or its popular name power spectrum of Equation A.2 is given by its Fourier transform,

$$\Phi_{yy}(\omega) = F\left[\sum_{l=-\infty}^{\infty} \sum_{p=-\infty}^{\infty} h(l)h(p)r_N(k+p-l)\right] \tag{A.3}$$

$$= \sum_{l=-\infty}^{\infty} \sum_{p=-\infty}^{\infty} h(l)h(p)\,F\left[r_N(k+p-l)\right] \tag{A.4}$$

time shifting in autocorrelation domain corresponds to rotation in frequency domain, hence

$$\Phi_{yy}(\omega) = \sum_{l=-\infty}^{\infty} \sum_{p=-\infty}^{\infty} h(l)h(p)\Phi_{nn}(\omega)e^{j\omega(p-l)} \tag{A.5}$$

rearranging the Equation A.5 gives

$$\Phi_{yy}(\omega) = \Phi_{nn}(\omega) \sum_{l=-\infty}^{\infty} h(l)e^{-j\omega l} \sum_{p=-\infty}^{\infty} h(p)e^{j\omega p} \tag{A.6}$$

$$\Phi_{yy}(\omega) = \Phi_{nn}(\omega) H\left(e^{j\omega}\right) H^{*}\left(e^{j\omega}\right) \tag{A.7}$$

From Eq A.7 it follows the output-input relation

$$\Phi_{yy}(\omega) = \Phi_{nn}(\omega) \left|H\left(e^{j\omega}\right)\right|^{2} \tag{A.8}$$

Thus, the frequency response completely determines the shape of the output PSD. This result also shows that any PSD must be non negative. Equation A.8 is a powerful result used in filtering and system identification.

Appendix B

Frequency response for noncausal Wiener filter

Suppose we have observed the signal $\psi(k), k = 0, 1, 2, ...N-1$, over a fine time interval and we have a signal and noise model given by

$$\psi(k) = \theta(k) + n(k) \tag{B.1}$$

We want to apply the filter $H(q)$

$$\hat{\theta}(k) = H(q)\,\psi(k) = H(\theta(k) + n(k)) \tag{B.2}$$

that attenuate the noise $n(k)$ and keep the signal as intact as possible. We want to minimize

$$\varepsilon(k) = \hat{\theta}(k) - \theta(k) \tag{B.3}$$

The DTFT of this difference can be written as

$$\varepsilon\left(e^{j\omega}\right) = \hat{\Theta}\left(e^{j\omega}\right) - \Theta\left(e^{j\omega}\right) = H\left(e^{j\omega}\right)\Psi\left(e^{j\omega}\right) - \Theta\left(e^{j\omega}\right) \tag{B.4}$$

$$\varepsilon\left(e^{j\omega}\right) = \left(H\left(e^{j\omega}\right) - 1\right)\Theta\left(e^{j\omega}\right) + H\left(e^{j\omega}\right)N\left(e^{j\omega}\right) \tag{B.5}$$

where Ψ, Θ and N denote the DTFT of ψ, θ and n respectively and H is the transfer function of the filter. Now applying Parseval's formula,

$$\sum_{k=0}^{\infty} \varepsilon^2(k) = \frac{1}{2\pi}\int_{-\pi}^{\pi} |\varepsilon\left(e^{j\omega}\right)|^2 d\omega \tag{B.6}$$

$$= \frac{1}{2\pi}\int_{-\pi}^{\pi} |H\left(e^{j\omega}\right) - 1|^2 \Phi_{\theta\theta}(\omega) + |H\left(e^{j\omega}\right)|^2 \Phi_{nn}(w)\, d\omega \tag{B.7}$$

Consider the Equation B.7, in order to minimize $\varepsilon(k)$ we should design the filter $H(z)$ such that

- $|H\left(e^{j\omega}\right)-1|$ is small for ω where $|\Theta\left(e^{j\omega}\right)|$ is large
- $|H\left(e^{j\omega}\right)|$ is small for ω where $|N(e^{j\omega})|$ is large

If the spectra$\Phi_{\theta\theta}(\omega)$ and $\Phi_{nn}(\omega)$ are both known, the optimal filter that minimized the Equation B.7 can be derived by optimizing $H\left(e^{j\omega}\right)$ for each frequency. It can be shown that the filter must be real (all other functions are real and symmetric). For each frequency we are facing the optimization problem

$$min_H V(H) = min_H (H-1)^2 \Phi_{\theta\theta} + H^2 \Phi_{nn} \tag{B.8}$$

that has the optimal solution

$$H^{opt} = \frac{\Phi_{\theta\theta}}{\Phi_{\theta\theta}+\Phi_{nn}} \tag{B.9}$$

consequently, B.9 in B.8 gives

$$V\left(H^{opt}\right) = \frac{\Phi_{\theta\theta}^2\Phi_{nn}+\Phi_{\theta\theta}\Phi_{vv}^2}{\left(\Phi_{\theta\theta}+\Phi_{nn}\right)^2} = \frac{\Phi_{\theta\theta}\Phi_{nn}}{\Phi_{\theta\theta}+\Phi_{nn}} \tag{B.10}$$

The frequency response $H\left(e^{j\omega}\right)$ of the filter is obtained straightforward. This is a very important result that we will use in the derivation of Wiener filter solution $W(z)$

$$H\left(e^{j\omega}\right) = \frac{\Phi_{\theta\theta}(\omega)}{\Phi_{\theta\theta}(\omega)+\Phi_{nn}(\omega)} \tag{B.11}$$

Notice that the filter defined by Eq. B.11 defines a non-causal filter, meaning that the output depends on future values of the input as well as the past i.e $h(\tau) \neq 0$ for some $\tau \leq 0$. To generate a causal filter, we need to define a filter such that it depends on the past and current values of the input, thus $h(\tau) = 0$ for all $\tau \leq 0$.

Appendix C

Derivation of the Wiener transfer function

The derivation of the Wiener transfer function in subsection 3.3.1.1 given by Equation 3.14 requires the used of the Wolf decomposition theorem as well as the equation A.8, derived in Appendix A. The Wolf decomposition theorem says that every stationary stochastic process can be realize as the convolution

$$\theta(k) = \sum_{l} v(l)\, h(k-l) \tag{C.1}$$

for some deterministic sequence $h(k)$ (filter), where $v(k)$ is white noise. The transfer function of the phase noise model is given by

$$H_\theta(z) = \frac{1}{1-z^{-1}} \tag{C.2}$$

We rewrite the Equation A.8 for convenience. Thus

$$\Phi_{\theta\theta}(\omega) = \left|H\left(e^{j\omega}\right)\right|^2 \Phi_{vv}(\omega) \tag{C.3}$$

Plugging $\Phi_{vv}(\omega) = \sigma_\theta^2$ into the Equation C.3 and expanding $|H\left(e^{j\omega}\right)|^2$ we get

$$\Phi_{\theta\theta}(\omega) = |H\left(e^{i\omega}\right)^2|\sigma_\theta^2 = \frac{\sigma_\theta^2}{\left(1-e^{-j\omega}\right)\left(1-e^{j\omega}\right)}, \tag{C.4}$$

replacing $z = e^{i\omega}$ in C.4 gives

$$\Phi_{\theta\theta}(z) = \frac{\sigma_\theta^2}{\left(z^{-1}-1\right)(z-1)} \tag{C.5}$$

Spectral factorization of the observation $\hat{\theta}(k)$ is determine by

$$\Phi_{\hat{\theta}\hat{\theta}}(\omega) = \Phi_{\theta\theta}(\omega) + \Phi_{nn}(\omega) \tag{C.6}$$

Replacing Equation C.6 and the value for $\Phi_{nn}(\omega)$ into Equation C.7

$$\Phi_{\hat{\theta}\hat{\theta}}(z) = \frac{\sigma_\theta^2}{(z^{-1}-1)(z-1)} + \sigma^2 = \frac{\sigma_\theta^2 + \sigma^2\left(z^{-1}-1\right)(z-1)}{(z^{-1}-1)(z-1)} \tag{C.7}$$

Having found $\Phi_{\hat{\theta}\hat{\theta}}(z)$ and $\Phi_{\theta\theta}(z)$ we are now able to use the result given by the Equation B.11 in Appendix B. The relation between the frequency function $H(e^{j\omega})$ and the transfer function $W(z)$ is established by the relation between the DTFT and the Z-transform[1] . Now since the Wiener filter coefficients $w(n)$ assure a stable sequence, obtaining $W(z)$ is straightforward. By definition, the transfer function $W(z)$ is determined by the ratio of $\Phi_{\theta\theta}(z)$ to $\Phi_{\hat{\theta}\hat{\theta}}(z)$,

$$W(z) = \frac{\Phi_{\theta\theta}(z)}{\Phi_{\hat{\theta}\hat{\theta}}(z)} \tag{C.8}$$

Replacing Equation C.5 into Equation C.7 into Equation C.8

$$W(z) = \frac{\Phi_{\theta\theta}(z)}{\Phi_{\hat{\theta}\hat{\theta}}(z)} = \frac{\frac{\sigma_\theta^2}{\left(z^{-1}-1\right)(z-1)}}{\frac{\sigma_\theta^2+\sigma^2\left(z^{-1}-1\right)(z-1)}{\left(z^{-1}-1\right)(z-1)}} = \frac{\sigma_\theta^2}{\sigma_\theta^2 + \sigma^2(z^{-1}-1)(z-1)} \tag{C.9}$$

Now pulling out σ^2 in the denominator in the above equation we get

$$W(z) = \frac{\sigma_\theta^2}{\sigma^2\left(\frac{\sigma_\theta^2}{\sigma^2} + (z^{-1}-1)(z-1)\right)}. \tag{C.10}$$

replacing $\frac{\sigma_\theta^2}{\sigma^2}$ by r gives

$$W(z) = \frac{r}{r+(z^{-1}-1)(z-1)} = \frac{r}{r+(1-z^{-1}-z+1)} = \frac{r}{r+(2-z^{-1}-z)} \tag{C.11}$$

$$W(z) = \frac{r}{r+\frac{z^{-1}}{z^{-1}}(2-z^{-1}-z)} = \frac{rz^{-1}}{rz^{-1}+2z^{-1}-2z^{-2}-1} \tag{C.12}$$

which leads to the Wiener transfer function given by the Equation 3.14 given in section 3

$$W(z) = \frac{rz^{-1}}{-1+(2+r)z^{-1}-z^{-2}} \tag{C.13}$$

[1]The DTFT can be interpreted as the Z-transform evaluated on the unit circle $z = e^{j\omega}$

Appendix D

Derivation of Wiener coefficients

The coefficients $w(k)$ are obtained by taking the inverse Z-transform of Equation C.13. The first step is to find the poles in $W(z)$. Let $\alpha_1$and α_2 be the poles, then it follows that

$$W(z) = \frac{rz}{(z-\alpha_1)(z-\alpha_2)} = \frac{rz}{\alpha_1-\alpha_2}\left(\frac{1}{z-\alpha_1} - \frac{1}{z-\alpha_2}\right) \tag{D.1}$$

Now since α_1 and α_2 are inverse of each other we can make $\alpha_1 = \frac{1}{\alpha}$ and α_2=α, this math trick gives

$$W(z) = \frac{\alpha r}{1-\alpha^2}\left(\frac{z}{z-\frac{1}{\alpha}} - \frac{z}{z-\alpha}\right) \tag{D.2}$$

$$W(z) = \frac{\alpha r}{1-\alpha^2}\left(\frac{1}{1-\frac{z^{-1}}{\alpha}}\right) + \frac{\alpha r}{1-\alpha^2}\left(\frac{-z}{\alpha-z}\right) \tag{D.3}$$

the radius of convergence ROC of the first term is:

$$\text{ROC } |\tfrac{\alpha}{z}| < 1 \text{ or } |\alpha| < |z|$$

and the ROC of the second term is

$$\text{ROC } |\tfrac{z}{\alpha}| < \text{ or } |z| < |\alpha|$$

Thus

$$= \frac{\alpha r}{1-\alpha^2}\left(\frac{1}{1-\frac{\alpha}{z}}\right) + \frac{\alpha r}{1-\alpha^2}\left(1-\frac{\alpha}{\alpha-z}\right) \tag{D.4}$$

$$= \frac{\alpha r}{1-\alpha^2}\left(\frac{1}{1-\frac{\alpha}{z}}\right) + \frac{\alpha r}{1-\alpha^2}\left(1-\frac{1}{1-\frac{z}{\alpha}}\right) \tag{D.5}$$

Finally

$$W(z) = \frac{\alpha r}{1-\alpha^2}\sum_{k=0}^{\infty}\alpha^k z^{-k} + \frac{\alpha r}{1-\alpha^2}\sum_{-\infty}^{-1}\alpha^k z^{-k} \tag{D.6}$$

By inspection we can clearly see that the first term is a Right sided exponential sequence and the second term is a Left sided exponential sequence. Therefore both sequences may be written as

$$w(k) = \frac{\alpha r}{1-\alpha^2}\alpha^k, \quad k \geq 0 \tag{D.7}$$

and

$$w(k) = \frac{\alpha r}{1-\alpha^2}\alpha^{-k}, \quad k < 0 \tag{D.8}$$

where

$$\alpha = \left(1+\frac{r}{2}\right) - \sqrt{\left(1+\frac{r}{2}\right)^2 - 1} \tag{D.9}$$

or written in compact form

$$w(k) = \begin{cases} \frac{\alpha r}{1-\alpha^2} & \alpha^k, \quad n \geq 0 \\ \frac{\alpha r}{1-\alpha^2} & \alpha^{-k}, \quad n < 0 \end{cases} \tag{D.10}$$

We can conclude from Equation D.10 that the optimum decay rate depends on the ratio between phase noise variance σ_θ^2 and additive noise variance σ^2.

Appendix E

MATLAB codes for simulation

This appendix contains the MATLAB code for each simulation block in Fig. 4.1.