Multipath Mitigating Correlation Kernels

by

Steven Miller

A Dissertation Presented in Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy

Approved November 2013 by the
Graduate Supervisory Committee:

Andreas Spanias, Chair
Cihan Tepedelenlioglu
Junshan Zhang
Konstantinos Tsakalis

ARIZONA STATE UNIVERSITY

December 2013

ABSTRACT

Autonomous vehicle control systems utilize real-time kinematic Global Navigation Satellite Systems (GNSS) receivers to provide a position within two-centimeter of truth. GNSS receivers utilize the satellite signal time of arrival estimates to solve for position; and multipath corrupts the time of arrival estimates with a time-varying bias. Time of arrival estimates are based upon accurate direct sequence spread spectrum (DSSS) code and carrier phase tracking. Current multipath mitigating GNSS solutions include fixed radiation pattern antennas and windowed delay-lock loop code phase discriminators. A new multipath mitigating code tracking algorithm is introduced that utilizes a non-symmetric correlation kernel to reject multipath. Independent parameters provide a means to trade-off code tracking discriminant gain against multipath mitigation performance. The algorithm performance is characterized in terms of multipath phase error bias, phase error estimation variance, tracking range, tracking ambiguity and implementation complexity. The algorithm is suitable for modernized GNSS signals including Binary Phase Shift Keyed (BPSK) and a variety of Binary Offset Keyed (BOC) signals. The algorithm compensates for unbalanced code sequences to ensure a code tracking bias does not result from the use of asymmetric correlation kernels. The algorithm does not require explicit knowledge of the propagation channel model. Design recommendations for selecting the algorithm parameters to mitigate precorrelation filter distortion are also provided.

DEDICATION

I dedicate this dissertation to my wife, Kathleen, and to my children, Michael, Erika and Ryan. Your kindness and support made this dissertation possible.

ACKNOWLEDGMENTS

I express my sincere gratitude to Dr. Andreas Spanias, my dissertation advisor, for his continuous guidance and patience. Appreciation and thanks are also extended to my committee members, Dr. Cihan Tepedelenlioglu, Dr. Konstantinos S. Tsakalis, and Dr. Junshan Zhang.

I greatly appreciate my family's support and sacrifice. Kathleen, Michael, Erika and Ryan, thank you for your love.

Finally, I thank Dr. Michael Whitehead, a leader in the Global Navigation Satellite System field and Chief Technical Officer at Hemisphere GNSS, for his professional support and willingness to share his knowledge.

TABLE OF CONTENTS

Page

LIST OF TABLES

Table Page

LIST OF FIGURES

Figure Page

Figure Page

Figure Page

LIST OF SYMBOLS

Symbol Page

Chapter 1

INTRODUCTION

A Staten Island ferry struck a pier as it was docking on January 9th, 2013 injuring 57 people. This was not the first accident for the ferry. On October 15, 2003, a similar accident killed 11 people [1]. This disaster may have been avoided if the ferry was autonomously controlled or a vehicle trajectory warning system was installed. Global Navigation Satellite Systems (GNSS) provide the enabling technology for real-time, autonomous vehicle navigation and control in such diverse applications as construction, mining, farming, and fishing. Consequently, there is commercial interest to increase measurement accuracy and integrity while simultaneously reducing the system cost. Modern GNSS receivers provide real-time position accuracy of a few centimeters. All GNSS receivers estimate a satellite's signal time of arrival to solve for position and time. The quality of these time-of-arrival (TOA) estimates are directly dependent upon accurate tracking of the direct sequence spread spectrum (DSSS) code and carrier phase. Unfortunately, *multipath is a dominant error source* within these systems since it corrupts the signal phase estimates with a time-varying bias.

The first GNSS system was designed by the US and consists of over 24 satellites at a height of 20,000 Km and an orbital period of about 12 hours [2]. The satellites are positioned such that at least four are in view from any position on the Earth with possibly greater than 10 visible depending upon the receiver's location.

The remainder of this chapter introduces fundamental GNSS concepts and develops the notion that position and time estimation are fundamentally a signal phase estimation problem. It then introduces basic types of range calculations and GNSS systemic errors

that degrade accuracy and robustness. Finally, it motivates how multipath is a dominant error and outlines the unique contributions of this thesis.

1.1. *GNSS Fundamentals and Motivating Multipath Mitigation*

GNSS is fundamentally a time estimation problem utilizing signal carrier and code phase estimates. The receiver calculates the signal propagation time for each satellite in view by extracting the signal origination time and satellite orbital position from the satellite message. An over-constrained linear equation is solved to minimize the receiver position error with respect to all visible satellites. The four variables of the linear equation are receiver position, *(x, y, z)*, and local time, *t*. When greater than four satellites are visible, additional unknowns, such as atmospheric effects, maybe estimated.

The requirement to estimate the local time is illustrated with a 2D position estimation example. Consider a ship navigating through a channel with three time-synchronized beacons positioned on shore. Each beacon simultaneously transmits a message that contains a time-stamp and its location. The ship can determine its position, as illustrated in Fig. 1-1, by calculating the propagation time from each beacon to the ship. The ship is located at the intersection of the three measurement circles since it is the only solution that is consistent with the distance observations, P1, P2 and P3. Now consider the timing uncertainty of the user local time. This creates a position uncertainty around each beacon as illustrated by the signal propagation rings in Fig. 1-2. Now the solution does not consist of a single point, but a volume of possible points.

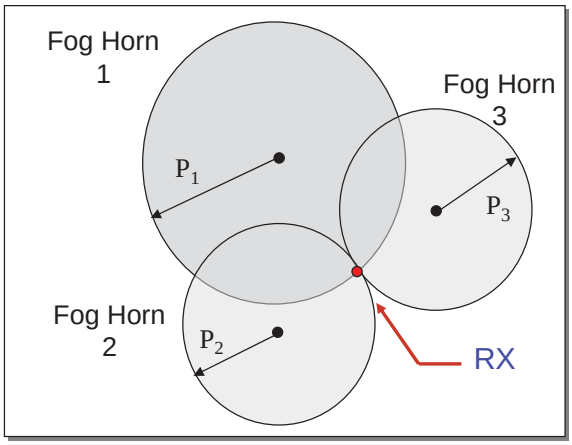

Fig. 1-1: 2D Range Estimation.

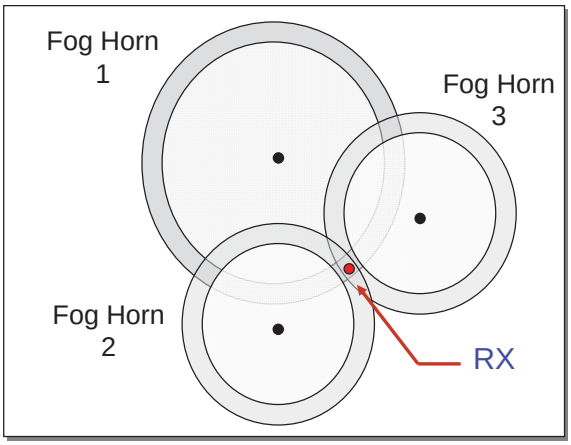

Fig. 1-2: 2D Range Estimation with Timing Ambiguity.

Unlike the beacon example, the GNSS system employs satellites that transmit direct sequence spread spectrum (DSSS) signals over multiple L-band carriers. Transmission over multiple L-Band carriers enables receivers to measure the propagation delay through the dispersive Ionosphere for more precise positioning. Due to the carefully constructed DSSS signal and message structures, the time estimation problem becomes a code and carrier phase estimation problem where the performance of the code and carrier tracking loops determine the phase estimate resolution and ultimately the position and time resolution. Reasonable performance for modern tracking loops sets the code and carrier phase estimates at the sub-meter and centimeter level respectively. Phase measurements from all visible GNSS signals are simultaneously captured by the receiver. This enables the removal of common mode clock errors and fixed delays due to the antenna cable, down-converter analog group delay, and digital signal processing delays.

Each satellite transmits carefully constructed messages that include the satellite's position, GNSS system time and other system parameters. Since each bit of the message perfectly aligns with a DSSS code chip; and the receiver counts whole and fractional code chips; an absolute, unambiguous time reference is established. Therefore, the receiver can

calculate the time of flight from when the satellite transmitted a certain code chip phase and when it was received. Unlike, the code phase measurement which provides absolute time measurements to the transmitting satellite; the carrier phase measurement is ambiguous. Although the receiver counts the number of whole and fractional carrier cycles (modulo one phase measurement period), there is no datum for carrier cycles. Therefore, the receiver does not absolutely measure the number of whole carrier cycles occurring during the message time-of-flight; rather it measures the number of carrier cycles that accumulate between measurement observations. This measurement is referred to as the delta carrier phase and the uncertainty in whole carrier cycles to the transmitting satellite is referred to as carrier phase ambiguity. Advanced GNSS receivers utilize the code phase estimates to limit the search for the carrier phase ambiguity. Consequently, code and carrier tracking errors, such as from multipath, degrades the phase measurement data; and therefore, the time of arrival estimate and the user location estimate.

The pseudo-range from the *i-th* satellite to the receiver is non-linear and given by,

$$\widehat{P_i} = \sqrt{(x_i - x_{rx})^2 + (y_i - y_{rx})^2 + (z_i - z_{rx})^2} + ct_{BIAS} + \varepsilon_i, \qquad (1\text{-}1)$$

where x_i, y_i and z_i are the positions of the *i-th* satellite given in Earth Centered Earth Fixed (ECEF) co-ordinates; x_{rx}, y_{rx}, and z_{rx} are the current predicted receiver position in ECEF coordinates; c is the speed of light; t_{BIAS} is the satellite and receiver clock bias with respect to the GNSS system time; and ε are systemic errors which will be described in more detail below [3]. Eq. (1-1) is called the *pseudo-range* since it includes the true geometric range between the receiver and the satellite as well as error terms.

Many methods to solve Eq. (1-1) exist, [3] and all suffer from multipath. An incremental approach will be presented to further introduce the fundamental concepts. Eq. (1-1) is linearized by taking the first term of a Taylor series expansion to yield,

$$\mathbf{H} = \begin{bmatrix} \frac{\partial p_1}{\partial x} & \frac{\partial p_1}{\partial y} & \frac{\partial p_1}{\partial z} & 1 \\ \frac{\partial p_2}{\partial x} & \frac{\partial p_2}{\partial y} & \frac{\partial p_2}{\partial z} & 1 \\ \vdots & \vdots & \vdots & \vdots \\ \frac{\partial p_N}{\partial x} & \frac{\partial p_N}{\partial y} & \frac{\partial p_N}{\partial z} & 1 \end{bmatrix}, \tag{1-2}$$

where the partials are the unit vectors in the direction of the *N* space vehicles in view. This matrix is also referred to the directional cosine matrix and shows how the solution depends on the geometry of the visible satellites with respect to the receiver. The receiver solves for position and time utilizing the following steps:

Step 1: Initially predict the receiver state, **X**, given by,

$$\mathbf{X} = [\hat{x}_{rx} \quad \hat{y}_{rx} \quad \hat{z}_{rx} \quad c\hat{t}_{BIAS}] \tag{1-3}$$

Step 2: Compute predicted pseudo ranges for all satellites in view, $\hat{p}_{i..N}$. This is based upon the valid assumption that the satellites are far away and the directional cosines, **H**, are the same for both the true receiver position and the current estimated position.

Step 3: Obtain the pseudo range from the code and carrier tracking loops, $p_{i..N}$. Update the receiver state by minimizing the error between the predicted and measured pseudo range,

$$\underbrace{\begin{bmatrix} P_1 \\ P_2 \\ \vdots \\ P_N \end{bmatrix} - \begin{bmatrix} \hat{P}_1 \\ \hat{P}_2 \\ \vdots \\ \hat{P}_N \end{bmatrix}}_{\mathbf{Z}} = \underbrace{\begin{bmatrix} \frac{\partial P_1}{\partial x} & \frac{\partial P_1}{\partial y} & \frac{\partial P_1}{\partial z} & 1 \\ \frac{\partial P_2}{\partial x} & \frac{\partial P_2}{\partial y} & \frac{\partial P_2}{\partial z} & 1 \\ \vdots & \vdots & \vdots & \vdots \\ \frac{\partial P_N}{\partial x} & \frac{\partial P_N}{\partial y} & \frac{\partial P_N}{\partial z} & 1 \end{bmatrix}}_{\mathbf{H}} \underbrace{\begin{bmatrix} \Delta\hat{x}_{rx} \\ \Delta\hat{y}_{rx} \\ \Delta\hat{z}_{rx} \\ c\hat{t}_{BIAS} \end{bmatrix}}_{\Delta\mathbf{X}}. \tag{1-4}$$

If solved by least squares, then,

$$\mathbf{Z} = \mathbf{H}\Delta\mathbf{x},$$

$$\Delta\mathbf{x} = (\mathbf{H^T W H})^{-1}\mathbf{H^T W Z}, \text{ and,} \tag{1-5}$$

$$\hat{\mathbf{x}}_{\mathbf{k+1}} = \hat{\mathbf{x}}_{\mathbf{k}} + \Delta\mathbf{x}.$$

where **W** is a weight matrix with each element corresponding to the confidence of each satellite observation. Typically, **W** is adaptive and is, at a minimum, a function of the satellite elevation angle: The lower elevation angle satellites are de-weighted since they have a lower signal-to-noise (SNR) and typically suffer a greater multipath bias; and consequently, a larger observation variance.

Recall that Eq. (1-2) implies the position solution depends on the geometry of the satellites with respect to the receiver. When the number of satellites is 4, the solution to Eq. (1-4) becomes, $\Delta\mathbf{x} = \mathbf{H}^{-1}\mathbf{Z}$, and the error covariance is,

$$\begin{aligned} \text{cov}(\Delta\mathbf{x}) &= \text{E}[\Delta\mathbf{x}\Delta\mathbf{x}^{\text{T}}] = \mathbf{H^{-1}}\underbrace{\text{E}[\mathbf{Z}\mathbf{Z}^{\text{T}}]}_{\sigma_{RX}^2}\mathbf{H}^{-\text{T}}, \\ \text{cov}(\Delta\mathbf{x}) &= \sigma_{RX}^2\underbrace{\mathbf{H^{-1}}\mathbf{H}^{-\text{T}}}_{\text{DOP}}, \end{aligned} \tag{1-6}$$

where σ_{RX}^2 is the measurement variance. The $\mathbf{H}^{-1}\mathbf{H}^{-T}$ term directly scales the measurement variance and is a function of the directional cosines. This term is called the Dilution of Precision (DOP) with diagonal elements consisting of the East, North, Vertical

and Time dilution of precision terms respectively. It is desirable for receivers to simultaneously utilize multiple GNSS constellations to provide a richer set of satellite geometry to decrease the DOP.

Recall that the GPS time estimation problem was transformed to a phase estimation problem. The performance of the code and carrier tracking loops determine the phase estimate resolution and ultimately the position resolution. The measured phase range from the the *i-th* satellite to a receiver, rx, is given by,

$$P_{i,rx} = R_{i,rx} + \varepsilon_{Orbit,i} + \varepsilon_{Iono,i,rx} + \varepsilon_{Tropo,i,rx} + c\left(\varepsilon_{t,i} - \varepsilon_{t,rx}\right) + \varepsilon_{MP,i,rx} + \varepsilon_{\sigma,i,rx}, \tag{1-7}$$

where,

$P_{i,rx}$ = the measured phase range from the *i-th* satellite to the receiver, rx;

$R_{i,rx}$ = the true range from the *i-th* satellite to the receiver, rx;

$\varepsilon_{Orbit,i}$ = the *i-th* satellite orbital error;

$\varepsilon_{Iono,i,rx}$ = the *i-th* satellite signal propagation delay through the ionosphere;

$\varepsilon_{Tropo,i,rx}$ = the *i-th* satellite signal propagation delay through the troposphere;

$\varepsilon_{t,i}$ = the *i-th* satellite clock error (seconds) with respect to the GNSS time datum;

$\varepsilon_{t,rx}$ = the receiver clock error (seconds) with respect to the GNSS time datum;

c = the speed of light in meters/second;

$\varepsilon_{MP,i,rx}$ = the code phase error due to multipath from the *i-th* satellite to the receiver; and,

$\varepsilon_{\sigma,i,rx}$ = Additive White Gaussian Noise (AWGN) and other un-modeled errors.

Note that if not otherwise stated, all units are in meters.

The methods to mitigate these errors drive receiver technology, cost and complexity. A *Pareto* chart of the error contributions is shown in Fig. 1-3. The error terms include: (1) Multipath; (2) ionosphere and troposphere atmospheric errors; (3) satellite orbital errors; and (4) the receiver and satellite clock bias with respect to the GNSS system time. Each contribution is discussed below.

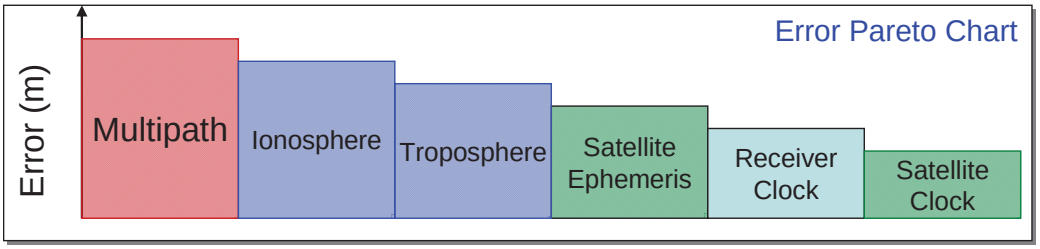

Fig. 1-3: Relative Positional Error Contributions.

Multipath is the vector sum of additional non-line-of-sight signal paths that cause a time varying bias. Although multipath inherently has a longer time-of-flight, the resulting satellite range error can be positive or negative. GNSS transmitted signals utilize nearly time-shift orthogonal code sequences, so the code multipath error's upper bound is set by the chip period: Tens of meters to hundreds of meters. Multipath is primarily mitigated by digital signal tracking algorithms and receiver antenna designs as discussed further in Chapters 2 and 3. Digital signal multipath mitigating techniques are the focus of this dissertation.

Ionosphere and troposphere errors are a consequence of the satellite signals propagating through the atmosphere and are independent of the signal structure. These time varying biases are dependent upon both the satellite and receiver positions and can be tens of meters [4]. The *ionosphere* contains charged particles that cause a frequency dependent signal propagation bias dependent upon the sunspot activity and the Earth's

crust. GNSS systems exploit the dispersive nature of the Ionosphere by transmitting on multiple L-Band frequencies to enable receivers to measure and mitigate the Ionosphere delays. The *troposphere* extends from the ground to an altitude of about 10Km and contains the Earth's weather layer. Troposphere errors are dependent upon water vapor and altitude and are frequency independent. Differential receiver techniques are typically used to mitigate atmospheric errors.

GNSS satellites are typically not geostationary, so their transmitted messages contain their orbital position. Earth's gravitational tides and atmospheric drag can cause errors between the stated trajectory and the true satellite trajectory by about 0.5m. These *ephemeris* errors will continuously decrease with advanced ground tracking techniques.

Satellite time is based upon atomic clocks; therefore, they are stable and accurate and any time error can be modeled as receiver local time perturbations. The stability of the local receiver time is determined by its reference clock which drifts with time, temperature, and vibration. Temperature compensated oscillators (TCXO) create a sub-meter time-varying range bias. These and other errors that are common to *all* GNSS measurements are estimated within t_{BIAS} as shown in Eq. (1-1).

Commercial GNSS receivers are primarily of two types: (1) Stand alone; or (2) Differential Real-Time Kinematic (RTK). The stand-alone receiver provides real-time positioning of about 0.5m and is the dominant type due to simplicity and cost. As the name implies, it does not require additional support equipment. In contrast, differential receivers must not only process signals from the GNSS satellites but must also receive additional signals from a base-station. Differential receivers provide 2cm real-time positioning by removing correlated errors between the base-station and receiver but are considerably more

complex and higher cost. We shall show that multipath is the dominant error source for both types of receivers.

Stand-Alone Carrier Smoothed Code Range Measurement Model

The variance of the position estimate is reduced by smoothing the code phase estimate with the carrier phase estimate. The technique, referred to as carrier-smoothing, combines two measurements: The unambiguous, but higher variance, code phase measurement and the ambiguous, but lower variance, carrier phase measurement. This method, also referred to as the Hatch Filter [4], achieves decimeter level position resolution. It formulates the problem as a weighted one-pole IIR filter given by,

$$\tilde{P}_k = \alpha_k P_{k-1} + (1-\alpha_k)\left(\tilde{P}_{k-1} + c_k - c_{k-1}\right), \tag{1-8}$$

where k is the iteration index; α_k is the time varying IIR weight; c_k and p_k are the carrier and code phase measurements at time k, respectively. $\tilde{P}_k$ is the smoothed pseudo-range that is applied to Eq. (1-1). The weight, α_k, is also given by an IIR structure that initializes the value near 0.99. This initially heavily weighs the code measurements to drive the solution to the unambiguous range mean contained within the noisy code measurement. The weight is gradually decreased until it saturates near 0.01 which heavily weighs the lower variance delta carrier phase measurements; thus yielding smoothed range measurements. Note that a code phase bias caused by multipath will still exist within the smoothed range, $\tilde{P}_k$.

Differential Receivers

A differential system consists of a base station and a rover as shown in Fig. 1-4. The base-station consists of a GNSS receiver and a radio transmitter. The base-station

broadcasts its code and carrier phase measurements along with its location; which is typically surveyed and known to within a few millimeters. The user's receiver, called a rover, uses differencing techniques to remove base-station and rover correlated errors such as clock, ephemeris and atmospheric errors, without explicitly estimating each error contribution. This is analogous to how a differential amplifier removes common mode errors. Consequently, errors that are temporally and spatially coherent are removed. Since base and rover multipath biases are independent; differential techniques tend to worsen the multipath bias. The inability of a differential system to mitigate multipath is the primary motivation to develop advanced multipath mitigating processing techniques.

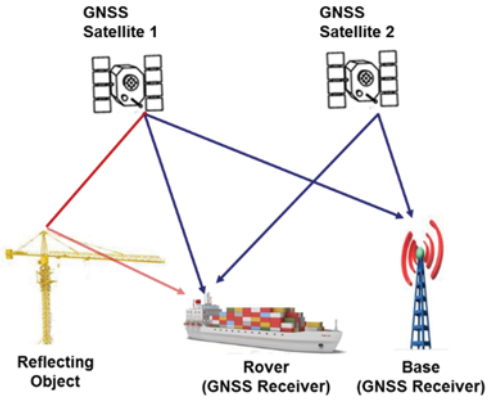

Fig. 1-4: Base and Rover Differential Scheme to Mitigate Position Error.

Within Eq. (1-7), replace the subscript *RX* with either a *B* or an *R* to denote a base or rover receiver respectively. Consider only satellites that are visible by both the base and the rover and assume the base and rover are close enough that they experience the same atmospheric conditions (spatially correlated atmospheric errors), then,

$$\varepsilon_{atmos,i} = \varepsilon_{Iono,i,B} + \varepsilon_{Tropo,i,B} = \varepsilon_{Iono,i,R} + \varepsilon_{Tropo,i,R}. \tag{1-9}$$

The single difference is calculated to subtract out common mode errors. The single difference is,

$$P_{i,\Delta RB} = P_{i,R} - P_{i,B}. \tag{1-10}$$

Substituting in Eq. (1-7) into Eq. (1-10) yields,

$$\begin{aligned} P_{i,\Delta RB} = {} & \left(R_{i,R} - R_{i,B}\right) \\ & + \left(\varepsilon_{Orbit,i} + \varepsilon_{atmos,i} + c\left(\varepsilon_{t,i} - \varepsilon_{t,R}\right) + \varepsilon_{MP,i,R} + \varepsilon_{\sigma,i,R}\right) \\ & - \left(\varepsilon_{Orbit,i} + \varepsilon_{atmos,i} + c\left(\varepsilon_{t,i} - \varepsilon_{t,B}\right) + \varepsilon_{MP,i,B} + \varepsilon_{\sigma,i,B}\right), \end{aligned} \tag{1-11}$$

which simplifies to,

$$\begin{aligned} P_{i,\Delta RB} = {} & \left(R_{i,R} - R_{i,B}\right) + \left(c\varepsilon_{t,R} + \varepsilon_{MP,i,R} + \varepsilon_{\sigma,i,R}\right) \\ & - \left(c\varepsilon_{t,B} + \varepsilon_{MP,i,B} + \varepsilon_{\sigma,i,B}\right), \end{aligned} \tag{1-12}$$

Note that the atmospheric and satellite errors are removed. The remaining errors are multipath, base and rover clock errors, as well as AWGN. An additional difference is performed again to remove the receiver clock errors. This double difference is taken between different satellites, designated as *1* and *2*, which yields,

$$P_{\nabla\Delta RB} = \left(P_{1,R} - P_{1,B}\right) - \left(P_{2,R} - P_{2,B}\right). \tag{1-13}$$

Substituting Eq. (1-12) into Eq. (1-13) yields,

$$\begin{aligned} P_{\nabla\Delta RB} = {} & \underbrace{\left\{\left(R_{1,R} - R_{1,B}\right) - \left(R_{2,R} - R_{2,B}\right)\right\}}_{Delta\ Range} \\ & + \underbrace{\left\{\left(\varepsilon_{MP,1,R} - \varepsilon_{MP,1,B}\right) - \left(\varepsilon_{MP,2,R} - \varepsilon_{MP,2B}\right)\right\}}_{Unmitigated\ Multipath} \\ & + \underbrace{\left\{\left(\varepsilon_{\sigma,1,R} - \varepsilon_{\sigma,1,B}\right) - \left(\varepsilon_{\sigma,2,R} - \varepsilon_{\sigma,2,B}\right)\right\}}_{noise}, \end{aligned} \tag{1-14}$$

Note that the remaining errors are multipath and noise.

Base and rover correlated errors, such as atmospheric errors, satellite clock errors and ephemeris errors, are removed by the differencing operation. However, since the base and rover are spatially separated, their respective multipath profiles are independent; and

therefore the dominating error remains multipath. This is exactly what motivates advanced multipath mitigation. Multipath mitigation techniques require both digital processing techniques and an advanced antenna design.

1.2. *Statement of Problem*

GNSS systems provide the enabling technology for real-time, autonomous vehicle navigation and control. These systems estimate position and time based upon time-of-arrival (TOA) estimates derived from tracking a satellite's code and carrier phase. Unfortunately, multipath is a dominant error source within either stand-alone or differential GNSS systems since it corrupts the signal phase estimates with a time-varying bias. Commercial GNSS receiver manufacturers are under increasing cost pressure to provide real-time 2-cm level position solutions suitable for both fixed and mobile applications. Consequently, this dissertation investigates multipath mitigating techniques that are power efficient, optimized for digital implementation and suitable for either binary phase shift keyed or binary offset carrier signaling techniques that may or may not include interleaved pilot and data channels.

1.3. *Contributions of Dissertation*

The unique contributions to multi-path mitigation include:

1. Development and analysis of a novel class of asymmetric correlation kernels for GNSS multipath mitigation resulting in US and international patents [5] and [6]. This kernel is compared to existing techniques and assessed on signal tracking bias, variance and robustness. Efficient digital implementation schemes are also presented. One scheme is utilized within several generations of application specific integrated circuits with over 300,000 units deployed

2. The effect of the receiver's pre-correlation analog filter on code-phase estimation is also analyzed in the context of the novel correlation kernel. A novel scheme to mitigate some filter distortion via selection of the correlation kernel parameters is provided.
3. The discovery that some GPS L1CA spreading codes are unbalanced which can cause a code phase bias when asymmetric correlation kernels are utilized. The development of an algorithm to compute an unbiased code phase estimate in the presence of unbalanced spreading codes.

1.4. *Organization of Dissertation*

The dissertation is organized into six chapters and three appendices. The first chapter motivates the GNSS multipath problem. The second chapter establishes signal, channel and receiver mathematical models. The third chapter describes existing mitigating techniques concentrating on correlation based processing methods. The fourth chapter proposes a novel, non-adaptive multipath mitigation technique based upon a non-symmetric kernel along with simulation and field results. The fifth chapter analyzes the effect of a receiver's precorrelation analog filter on correlation kernel design and proposes a novel mitigation technique. The sixth chapter provides concluding remarks on the proposed algorithms and directions for future work. Appendix A analyzes the GPS L1CA spreading code and demonstrates that some codes have unbalanced transition properties which must be accounted for to prevent a biased code phase estimate. Appendix B provides an analysis of delay lock loop discriminators. Appendix C discusses the design parameters for a digital carrier NCO and analyses the resulting correlation self-noise effects the code detection and tracking thresholds.

Chapter 2

GNSS SIGNAL MODELS

GNSS has now evolved to include multiple systems: (1) GPS operated by the US [2]; (2) GLONASS developed by Russia [7]; (3) Beidou in development by China; and (4) Galileo in development by the European Union [8]. Additional regional satellite based augmentation systems (SBAS) also exist and are operated by the United States, European Union, Japan, India, and China. The SBAS signals are similar to the GPS L1CA signal structure.

Consumer receivers utilize a limited set of the available signals; however precision systems, such as for survey and vehicle control, adopt a "greedy" approach and utilize all of the signals available. This "greedy" approach provides: (1) More observables to reduce estimation variance in a least squares sense; (2) better geometric observability to reduce position variance due to DOP; (3) more observable frequencies for Ionsospheric and Tropospheric observability and subsequent removal; and (4) more observable frequencies to increase the RTK carrier "ambiguity wavelength" thereby reducing the number of required searchable discrete points for RTK ambiguity resolution. Moreover, to compete in some national markets, a receiver must support the corresponding national GNSS system. Table 2-1 summarizes the planned and available commercial GNSS signals [4-6].

Table 2-1: Commercial GNSS Signal Characteristics by System [4-6]

System	Carrier (MHz)	Transmit Filter BW (MHz)	Signal Name	Modulation	Chip Rate (Mcps)	Code Length (Chips)
GPS (USA)	L1: 1575.420	30 [4]	L1CA	BPSK	1.023	1023
			L1P(Y)	BPSK	10.23	7 days
			L1C	AltBOC(m,n)	10.23	10230
	L2: 1227.60	30 [4]	L2CM-Data	TDM BPSK	0.5115	10230
			L2CL-Pilot		0.5115	767250
			L2P(Y)	BPSK	10.23	7 days
	L5: 1176.450	30 [4]	L5I-Data	QPSK	10.23	10230
			L5Q-Pilot	QPSK	10.23	10230
Galileo (European Union)	E1: 1575.42	[6]	E1bOS	BOC(1,1)	1.023	4092
			E1cOS	BOC(1,1)	1.023	4092 * 25
	E5ab: 1191.795	[6]	Composite	AltBOC(15,10)	10.23	10230
	E5a: 1176.450		E5a-I-Data	BPSK	10.23	10230
			E5a-Q-Pilot	BPSK	10.23	10230
	E5b: 1207.140		E5b-I-Data	BPSK	10.23	10230
			E5b-Q-Pilot	BPSK	10.23	10230
GLONASS (Russia)	G1-Band $f(n) = nf_o+1602.0$ $f_o = 0.562$ $n = \{-7..+13\}$	[5]	G1K	BPSK	0.511	511
	G2-Band $f(n) = nf_o+1246.0$ $f_o = 0.4375$ $n = \{-7..+13\}$	[5]	G2K	BPSK	0.511	511
Beidou (China)	B1: 1561.1	Not Avail.	B1	BPSK	1.023	1023
	B2: 1176.450	Not Avail.	B2	QPSK	10.23	10230
	B3: 1207.1		B3	QPSK		

Many satellites, frequencies, and modulation schemes exist. Commercial grade receivers will focus on carriers near 1572.450 MHz and 1176.450 MHz since a dual frequency receiver can process GPS, Galileo and Beidou signals: L1CA, L1C, L5IQ, E1bOS, E1cOS, E5aIQ, B1 and B2IQ. Precision GNSS receivers will process all of the

available signals; and consequently, a multi-path mitigation scheme must be suitable for all of the signaling types such as BPSK, QPSK, and Binary Offset Carry (BOC). The characteristics of these signaling types are described in detail below.

The Russian system utilizes frequency division multiplexing (FDMA). This has the disadvantage that all satellite signals experience a different group delay (due to the different frequency as a consequence of channelization) and therefore this complicates the positioning calculation due to this non-common mode code phase bias. It is anticipated that future Russian systems will utilize carrier division multiple access (CDMA). It should be noted that the Russian system is the only other globally deployed and fully functioning GNSS system beyond GPS and differentially processing GLONASS signals improves receiver performance.

This chapter develops mathematical models for the transmitted satellite signals, the propagation channel and receiver.

2.1. *Signal Structures*

GNSS signals are specifically designed to facilitate code-phase estimation and can be grouped into four distinct phase-shift keyed signal structures:

(1) Rectangular pulse weighted signals such as GPS L1CA;

(2) Binary-offset-carrier (BOC) pulse weighted such as Galileo E1bOS;

(3) Rectangular pulse weight with time interleaved data and pilot channels such as GPS L2C;

(4) Alt-BOC signals such as GPS L1C and Galileo E5ab.

The characteristics that influence signal tracking, such as energy spectral density and autocorrelation are investigated for each signal type. The baseband signal has the form:

$$x(t) = f(t) * g(t) \sum_{k=-\infty}^{\infty} C_n \, \delta(t - kT_c), \tag{2.1}$$

where C_n denotes the n-th spreading chip with magnitudes {+1,-1} and period T_c; $f(t)$ represents the impulse response of the band limited transmit filters; and $g(t)$ represents the pulse waveform of a single chip. The signal-in-space (SIS) specification for each system provides $f(t)$. For power constrained receivers, the receiver band-width is narrower than the transmit band-width. Fig. 2-1 plots $g(t)$ for *Rect(n)*, *SinBOC(m,n)* and *CosBOC(m,n)* pulse waveforms. Rect(n) describes a rectangular pulse at rate $f_c = nf_0$; with $f_0 \equiv 1.023MHz$. BOC(m,n) describes a Rect(n) signal further modulated by a square wave with fundamental frequency, $f_s = mf_0$, where m is the number of square wave cycles within one chip period, subject to *m*, *n* and $M = \frac{2m}{n}$ are positive integers [9].

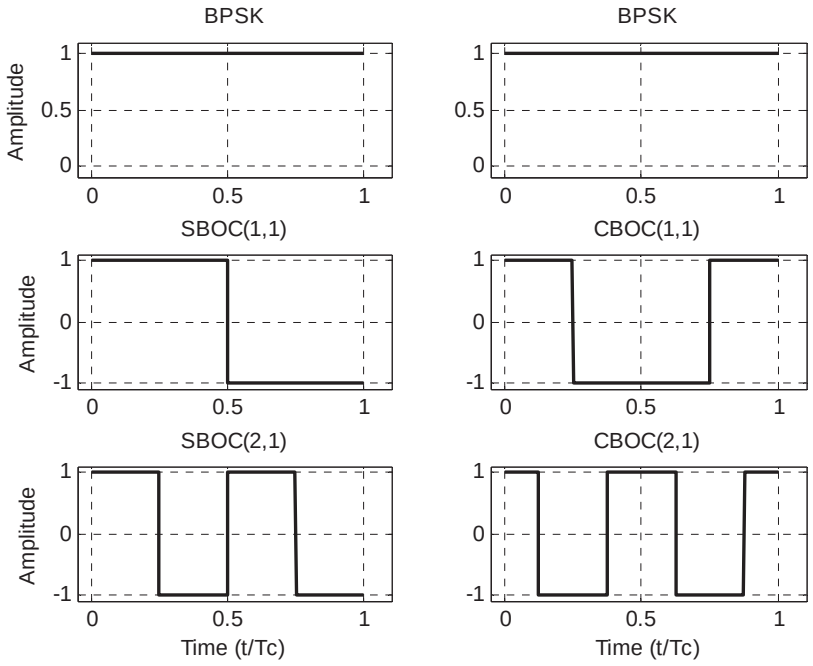

Fig. 2-1: $g(t)$ Waveforms for Rect(n), SinBOC(m,n) and CosBOC(m,n).

For a signal defined with a rectangular pulse, *Rect(n)*, then $g(t)$ has the form,

$$g(t) = \sqrt{f_c} Rect_{T_c}(t) = \begin{cases} \sqrt{f_c}, & \frac{-T_c}{2} \le t \le \frac{T_c}{2}, \\ 0, & Otherwise \end{cases} \tag{2.2}$$

and the power spectrum,

$$|S_{RECT}(f)|^2 = T_c sinc^2(\pi T_c f) = T_c \left[\frac{sin(\pi T_c f)}{\pi T_c f}\right]^2. \tag{2.3}$$

For a signal defined with a SinBOC(m,n) pulse, $g(t)$ has the form,

$$g(t) = \begin{cases} \sqrt{f_c} sign\big(sin(2\pi m f_c t)\big), & \frac{-T_c}{2} \le t \le \frac{T_c}{2}, \\ 0, & Otherwise \end{cases} \tag{2.4}$$

Which is equivalent to,

$$g(t) = \begin{cases} \sqrt{f_c} \sum_{k=0}^{m} (-1)^k Rect_{T_s}(t - kT_s), & \frac{-T_c}{2} \le t \le \frac{T_c}{2}. \\ 0, & Otherwise \end{cases} \tag{2.5}$$

Where $T_s = \frac{T_c}{2m}$ is half the square wave period for the secondary code. The corresponding power spectrum is [9],

$$S_{SBOC}(f) = \begin{cases} 4T_c sinc^2(\pi f T_c) \left[\frac{sin^2\left(\frac{\pi f}{4f_s}\right)}{cos\left(\frac{\pi f}{2f_s}\right)}\right]^2, & k\ even \\ 4T_c \frac{cos^2(\pi f T_c)}{(\pi f T_c)^2} \left[\frac{sin^2\left(\frac{\pi f}{4f_s}\right)}{cos\left(\frac{\pi f}{2f_s}\right)}\right]^2, & k\ odd \end{cases} \tag{2.6}$$

With $k = \frac{2m}{n} = \frac{2f_s}{f_c} = \frac{T_c}{T_s}$; again with T_s half the square wave period.

Similarly for CosBOC(m,n) signals, $g(t)$ have the form,

$$g(t) = \sqrt{f_c} \sum_{i=0}^{2m-1} \sum_{k=0}^{1} (-1)^{i+k} Rect_{T_s}(t - iT_S - \frac{kT_S}{2}). \tag{2.7}$$

The power spectrum of CBOC(m,n) is given by [9],

$$S_{CBOC}(f) = \begin{cases} 4T_c sinc^2(\pi f T_c)\left[\dfrac{sin^2\left(\frac{\pi f}{4f_s}\right)}{cos\left(\frac{\pi f}{2f_s}\right)}\right]^2 & , k\ even \\ 4T_c \dfrac{cos^2(\pi f T_c)}{(\pi f T_c)^2}\left[\dfrac{sin^2\left(\frac{\pi f}{4f_s}\right)}{cos\left(\frac{\pi f}{2f_s}\right)}\right]^2 & , k\ odd \end{cases} \quad (2.8)$$

With $k = \frac{2m}{n} = \frac{2f_s}{f_c} = \frac{T_c}{T_s}$; again with T_s half the square wave period.

The normalized Rect(1) power spectrum is plotted in Fig. 2-2.

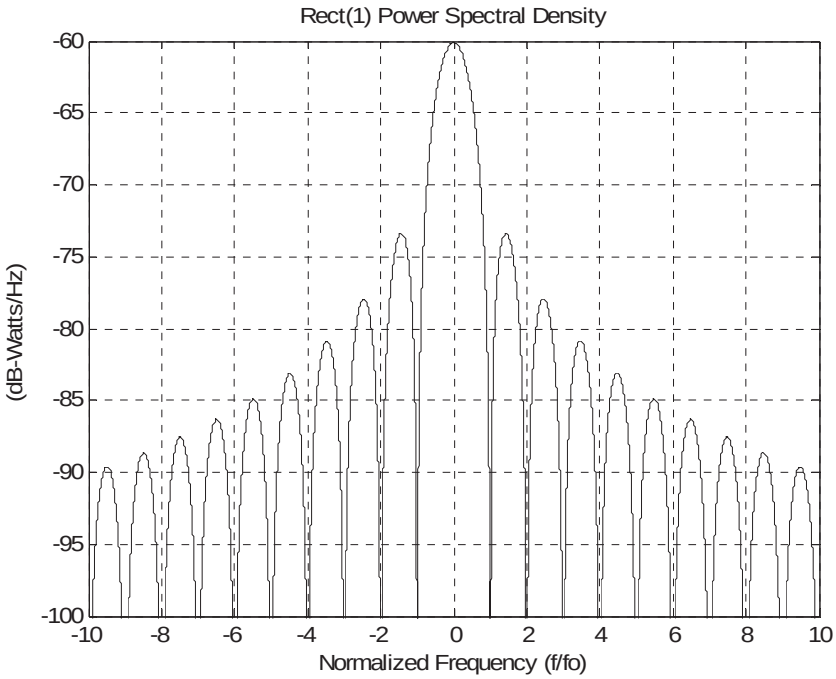

Fig. 2-2: $S_{Rect(1)}$ Normalized Power Spectrum.

The normalized power spectrum for SinBOC(m,n) and CosBOC(m,n) for various m and n are plotted in Fig. 2-3. In comparison to Rect(n) signals that have a dominant main-lobe, BOC(m,n) signals two dominant main-lobes with a separation determined by $m \cdot f_0$. This characteristic enables multiple systems to share the same carrier frequency while reducing co-channel interference due to cross-correlation [9].

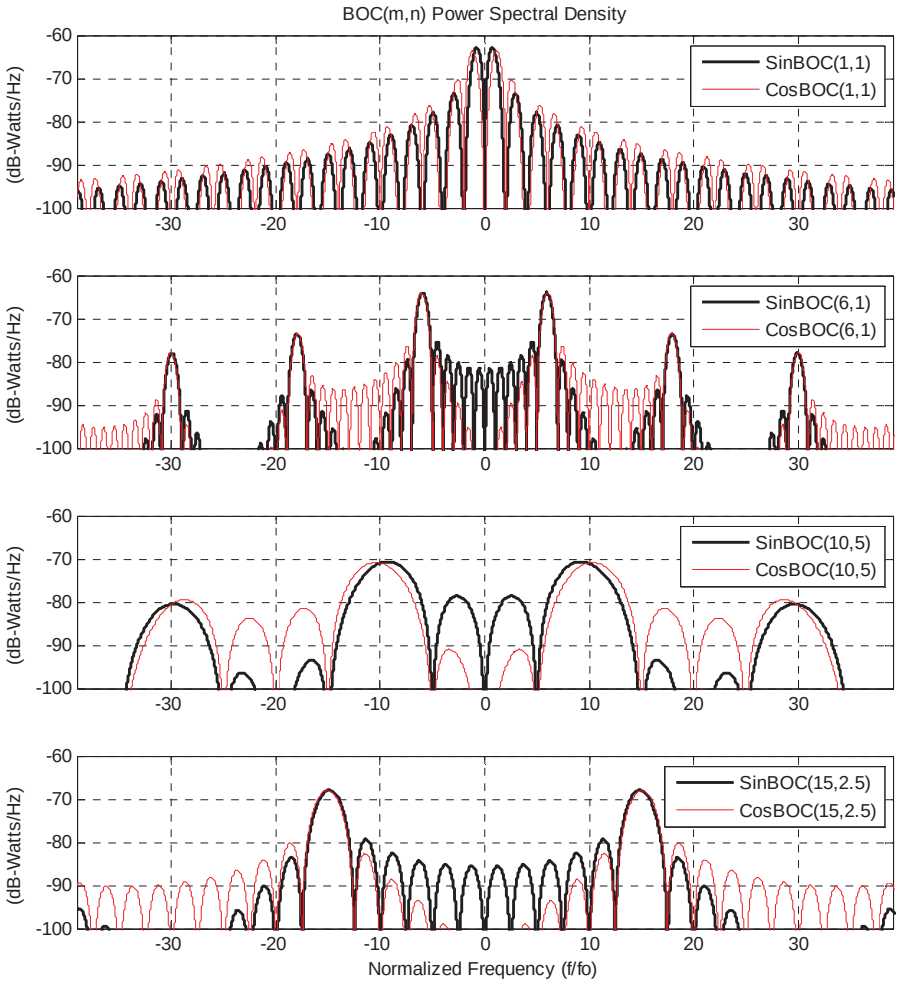

Fig. 2-3: Normalized Power Spectrums for Various SinBOC(m,n) and CosBOC(m,n) Signals.

A composite normalized power spectrum for Rect(1), SinBOC(1,1) and CosBOC(1,1) are plotted in Fig. 2-4 for comparison. Note the increasing signal bandwidths for Rect(1), SinBOC(1,1) and CosBOC(1,1) respectively.

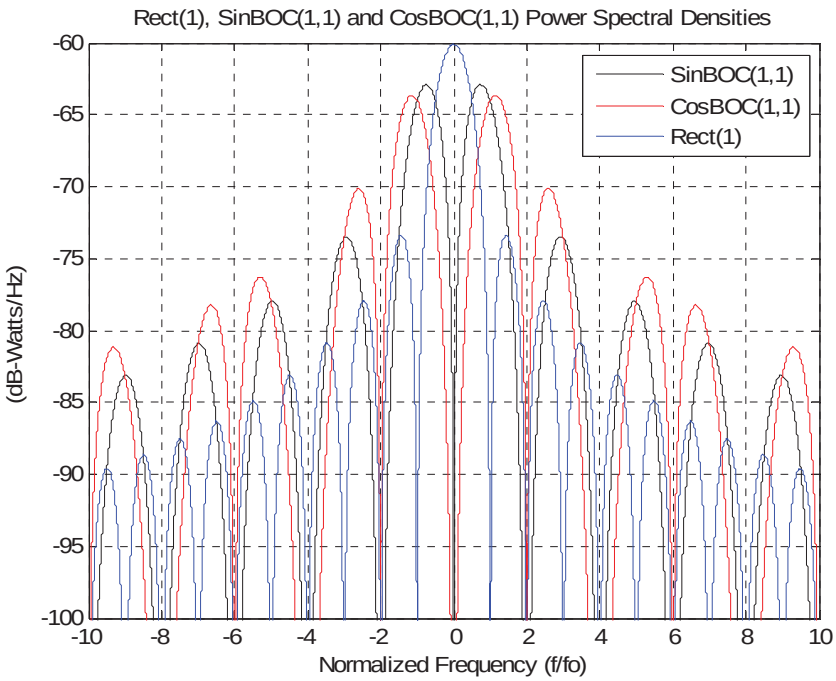

Fig. 2-4: Composite Normalized Power Spectrum for Rect(1), SinBOC(1,1) and CosBOC(1,1).

Fig. 2-5 plots the normalized power profiles, with respect to f_o, for Rect(1), SinBOC(1,1) and CosBOC(1,1). A receiver must have an input bandwidth of at least $4.5f_0$, $12f_0$, and $18f_0$ respectively; to capture 98% of the total available signal power. Therefore, the disadvantages of BOC signals include an increase in the required receiver bandwidth, sample-rate and power-consumption.

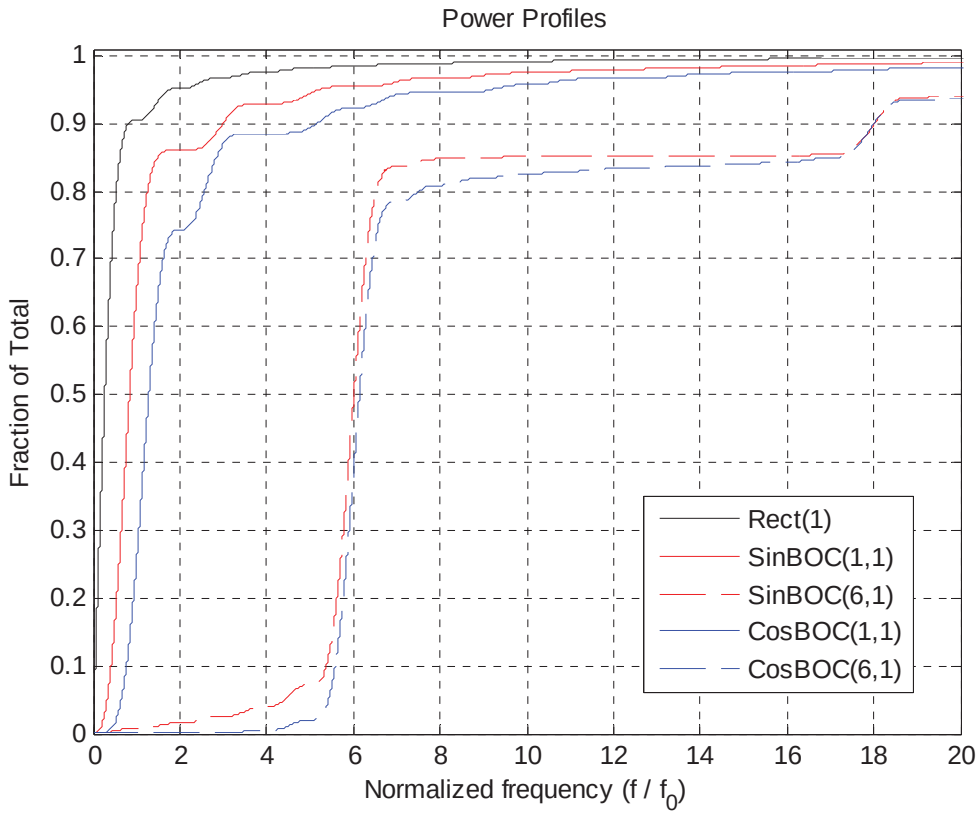

Fig. 2-5: Normalized Power Profiles for Rect(1), SinBOC(1,1) and CosBOC(1,1).

The Rect(1), SinBOC(1,1) and CosBOC(1,1) autocorrelation functions are potted in Fig. 2-6. The width of the main correlation peak is a measure of time-shift orthogonality (coherence time) and is inversely proportional to the required signal bandwidth. CosBOC(1,1) has a narrower peak than either SinBOC(1,1) or Rect(1). This means that time shifted replicas of the BOC(1,1) signals decorrelate more than a Rect signal given the same time shift; thereby providing an intrinsic level of multi-path mitigation. However, unlike the Rect signal with one correlation peak, the BOC signals have multiple correlation peaks which can cause false lock points if not properly handled. Both of these points will be explained in detail below.

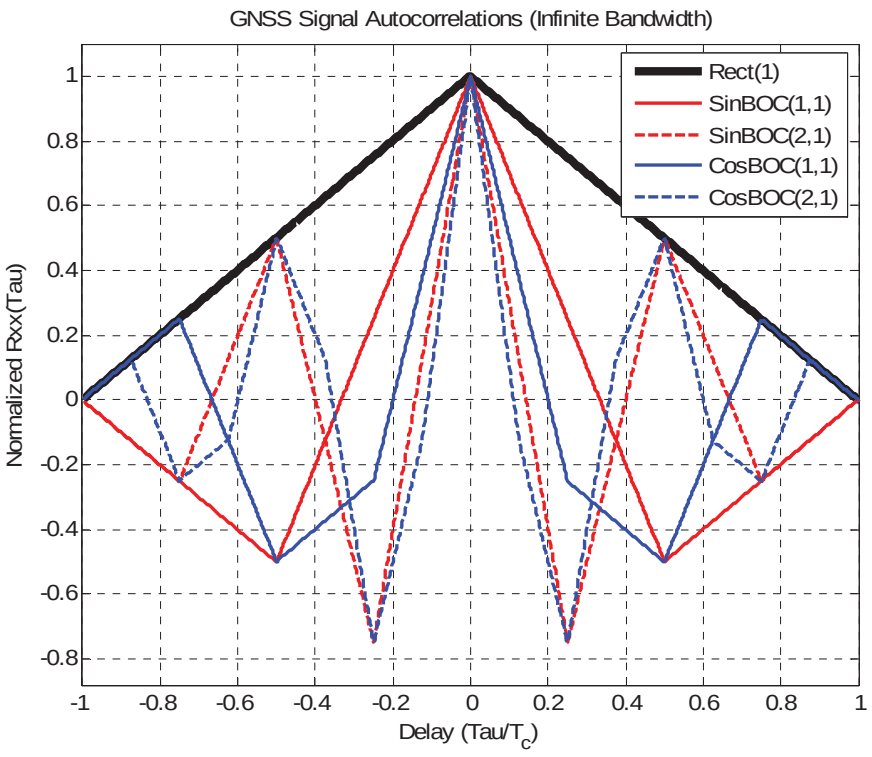

Fig. 2-6: Normalized Autocorrelations.

2.2. *Channel Model and Signal Propagation*

The GNSS signals occupy the same time and bandwidth. The incident signal will be a sum of the signals from the various transmitters present, interference and additive noise. Let N denote the numbers of transmitters and L_n denote the number of multipath signal components of the nth transmitter. The received signal vector at antenna element, k, may be expressed as

$$\mathbf{r}_k(t) = \sum_{n=1}^{N}\sum_{l=1}^{L_n}\mathbf{a}(\theta_{n,l,k})\alpha_{n,l,k}e^{j\phi_{n,l,k}}x(t-\tau_{n,l,k}) + \mathbf{n}(t), \tag{2.9}$$

where $\theta_{n,l,k}$, $\tau_{n,l,k}$, $\alpha_{n,l,k}$, and $\phi_{n,l,k}$ are the angle of arrival, delay time, path attenuation, and path phase of the l^{th} path of the n^{th} transmitter, respectively, impinging on antenna element

k. And $\mathbf{a}(\theta_{n,l})$ is the antenna response vector to a signal impinging upon the element with an direction of arrival (DOA), $\theta_{n,l}$. The receiver noise, $\mathbf{n}(t)$, is Additive White Gaussian Noise (AWGN) with zero mean and variance, σ_n^2. Fig. 2-7 illustrates different path delay scenarios within the channel model. Each ellipsoid represents equal path delay.

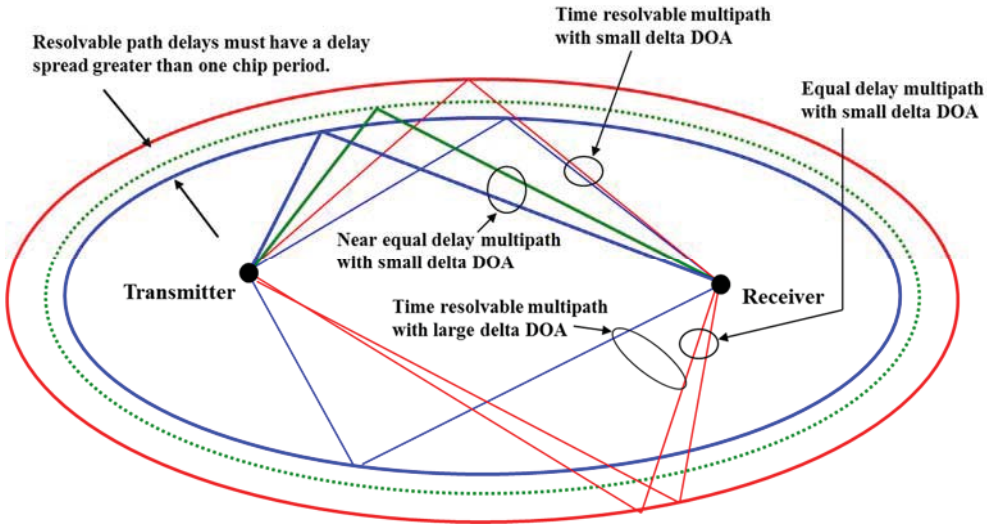

Fig. 2-7: Propagation Path Scenarios.

When only one transmitter and one receiver antenna element (with an ideal response) are considered, a simplified multipath model for the receiver becomes,

$$r(t) = \sum_{l=0}^{L} \alpha_l e^{j\phi_l} x(t-\tau_l) + n(t)\,. \tag{2.10}$$

Let $\alpha_0 \equiv 1$, $\phi_0 \equiv 1$, and $\tau_0 \equiv 0$ be the normalized line-of-sight signal parameters and parameters with *l>0* be with respect to the line-of-sight without loss of generality. From (2.10) it is clear that the ability to resolve multipath or estimate the channel impulse response is dependent upon the signal coherence time (autocorrelation).

All GNSS systems utilize Right Hand Circular Polarized (RHCP) signals since it is more robust to polarization changes that can occur as signals propagate through the Earth's Ionosphere [10]. Propagation models for various conditions are summarized in [11].

2.3. *Receiver Structures*

GNSS receivers utilizes a coherent correlator for signal detection and data demodulation. A complex signal model for a DSSS receiver operating on a real signal, *r(t)*, is shown in Fig. 2-8. The analog filter, H(ω), is located after the last down conversion stage and its bandwidth sets the system performance. It is referred to as the pre-correlation (or pre-detection) filter. The LMQ block includes an automatic gain controller and an adaptive Lloyd-Max Quantizer [12]. To reduce cost, digital sub-sampling is employed so the signal coming out of the analog process block is real with a non-zero digital carrier offset.

The following discussion assumes the signal, *z(t)*, has been down converted to near-baseband using conventional digital carrier tracking methods such as a Costas Loop. Appendix C provides an analysis of the design trade-offs for the digital carrier NCO and phase model suitable for GNSS applications and derives the resulting receiver self-noise associated. The remaining discussion concentrates on correlative code phase discriminators used within code tracking control loops.

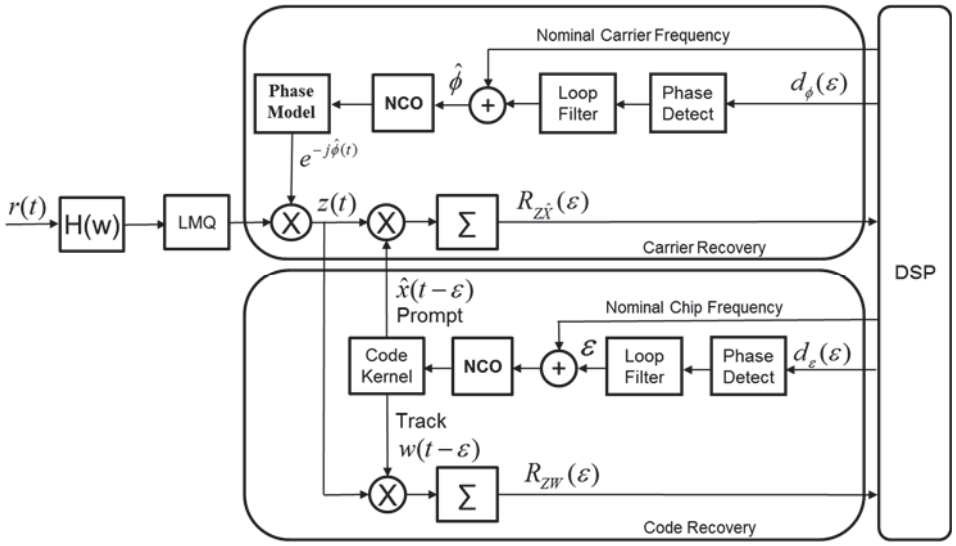

Fig. 2-8: Receiver Complex Signal Analysis Model.

The code recovery block operates on the complex near-baseband signal, *z(t)*, to track the incoming signal's code phase by driving the code phase error, ε, to zero. The block consists of a phase discriminator, a loop filter, a Numerically Controlled Oscillator (NCO) and a code kernel generator. Two kernels are generated: (1) A model, $\hat{x}(t)$, of the transmitted code sequence, *x(t)*; and (2) a code tracking kernel, *w(t)*. The kernel $\hat{x}(t)$ is correlated with *z(t)* to form the statistic $R_{Z\hat{X}}(\varepsilon)$. This complex valued statistic is used to form the carrier tracking loop discriminant, $d_{\emptyset}(\varepsilon)$ and decode the satellite's data bits. The kernel *w(t)* is also correlated with *z(t)* to form the statistic $R_{ZW}(\varepsilon)$. This kernel can be designed with multi-path mitigation properties and used as part of the code phase error discriminant, $d_{\varepsilon}(\varepsilon)$. The code phase discriminator estimates the phase error between the incoming measured signal and the code model. The filtered phase error is output from the loop filter and provides a delta-phase which is accumulated in the NCO to adjust the model phase and drive the phase error to zero. The control loop will advance or retard the model in time until it aligns with the incoming signal.

Consider the no-multipath case and using (2.1) to obtain *z(t)* yields,

$$z(t) = \{x(t) + n(t)\} * h(t)e^{-j\hat{\varphi}(t)}. \tag{2.11}$$

Solve for the correlation statistic, $R_{Z\hat{X}}(\varepsilon)$, using a suitable correlation period, *T*, which is determined by the data period and the required tracking dynamics. For GPS L1CA, the correlation period is typically set to the code epoch period of 1ms. $R_{Z\hat{X}}(\varepsilon)$ is found by,

$$R_{Z\hat{X}}(\varepsilon) = \frac{1}{T}\int_0^T z(t)\hat{x}^*(t-\varepsilon)dt, \tag{2.12}$$

and substituting (2.11) into (2.12) and expanding terms yields,

$$R_{Z\hat{X}}(\varepsilon) = \frac{1}{T}\int_0^T \left[\{x(t) + n(t)\} * h(t)e^{-j\hat{\varphi}(t)}\right]\hat{x}^*(t-\varepsilon)dt. \tag{2.13}$$

Represent the carrier tracking phase estimate, $e^{-j\hat{\varphi}(t)}$, as $e^{-j(2\pi f_e t+\varphi_e)}$, then the signaling term for (2.13) becomes,

$$R_{Z\hat{X}}(\varepsilon) = \frac{e^{-j\varphi_e}}{T}\int_0^T \left[x(t)\hat{x}^*(t-\varepsilon)e^{-j2\pi f_e t}\right] * h(t)dt, \tag{2.14}$$

and,

$$R_{Z\hat{X}}(\varepsilon) = \frac{e^{-j\varphi_e}}{T}\frac{e^{-j2\pi f_e t}}{j2\pi f_e t}\Bigg|_0^T R_{X\hat{X}}(\varepsilon) * h(t), \tag{2.15}$$

and,

$$R_{Z\hat{X}}(\varepsilon) = \left(R_{X\hat{X}}(\varepsilon) * h(t)\right)sinc(f_e T)e^{-j\left(2\pi f_e\frac{T}{2}+\varphi_e\right)}, \tag{2.16}$$

If the receiver is tracking, the frequency and phase error are small, and (2.16) can be simplified to,

$$R_{Z\hat{X}}(\varepsilon) = R_{X\hat{X}}(\varepsilon) * h(t). \tag{2.17}$$

The $R_{X\hat{X}}(\varepsilon)$ term is the auto-correlation of the spreading sequence. Plots of the auto-correlations and power spectrums for Rect(n), SinBOC(1,1) and CosBOC(1,1) were provided in Fig. (2-3) and Fig. (2-5).

The noise term of (2.13) is,

$$\mathrm{N}_{N\hat{X}}(\varepsilon) = \frac{e^{-j\varphi_e}}{T}\int_0^T \left[n(t)\hat{x}^*(t-\varepsilon)e^{-j2\pi f_e t}\right] * h(t)dt, \tag{2.18}$$

Since Gaussian noise is circularly symmetric, the complex rotations of (2.18) have no impact and can be removed. Moreover, since the model code chips are stationary, zero mean, equal probable with constant modulus, the noise term reduces to,

$$\mathrm{N}_{N\hat{X}}(\varepsilon) = \frac{1}{T}\int_0^T n(t) * h(t)dt. \tag{2.19}$$

This yields Gaussian noise with zero mean and variance, $\sigma^2_{N\hat{X}} = \sigma^2|H|^2$.

A similar analysis is conducted to solve for $R_{ZW}(\varepsilon)$ and yields a signaling term of,

$$\begin{aligned} R_{ZW}(\varepsilon) &= \frac{1}{T}\int_0^T z(t)w^*(t-\varepsilon)dt \\ &= \left(R_{XW}(\varepsilon) * h(t)\right)sinc(f_eT)e^{-j\left(2\pi f_e\frac{T}{2}+\varphi_e\right)}. \end{aligned} \tag{2.20}$$

When the receiver is carrier locked, the carrier frequency and phase errors are small, then (2.20) simplifies to,

$$R_{ZW}(\varepsilon) = R_{XW}(\varepsilon) * h(t). \tag{2.21}$$

The corresponding noise term is,

$$\mathrm{N}_{NW}(\varepsilon) = \frac{e^{-j\varphi_e}}{T}\int_0^T \left[n(t)W^*(t-\varepsilon)e^{-j2\pi f_e t}\right] * h(t)dt, \tag{2.22}$$

which yields Gaussian noise with zero mean and variance, $\sigma_{NW}^2 = \sigma^2|WH|^2$. A detailed analysis of the correlation process self-noise for a digital implementation is provided in Appendix C.

The design of the code tracking kernel, *w(t)*, depends on how the statistics $R_{Z\hat{X}}(\varepsilon)$ and $R_{ZW}(\varepsilon)$ are used to form the code phase discriminator, $d_\varepsilon(\varepsilon)$. The function of the code-phase discriminator is to estimate the signal's time-of-arrival and form a code-phase error estimation. A Maximum Likelihood time-of-arrival estimator is derived below [13] [14]. In general, given an independent identically distributed (iid) observation vector, $x_1, x_2, \ldots, x_N$, which is conditioned on the estimation parameters, θ, the likelihood function is given by,

$$L(\theta) = f_\theta(\mathbf{x}) = \prod_{i=1}^{N} P(x_i|\theta), \tag{2.23}$$

where *P(•)* is the conditional pdf. The log-likelihood becomes,

$$\Lambda_\theta(\mathbf{x}) = ln\{L(\theta)\} = \sum_{i=1}^{N} ln\{P(x_i|\theta)\}. \tag{2.24}$$

Therefore, the maximum likelihood is found by,

$$\max_\theta ln\{L(\theta)\}, \tag{2.25}$$

which becomes,

$$\frac{\partial}{\partial\theta} ln\{L(\theta)\} = \sum_{i=1}^{N} \left\{ \frac{1}{P(x_i|\theta)} \cdot \frac{\partial}{\partial\theta} P(x_i|\theta) \right\} = 0. \tag{2.26}$$

Now consider time of arrival maximum likelihood estimation with a baseband received signal,

$$\mathbf{x}(t) = s(t-\tau) + \mathbf{n}(t), \tag{2.27}$$

where x(t) is the delayed version of the reference signal, *s(t)*; τ is the path delay; and *n(t)* is additive white Gaussian noise with zero mean and variance, σ^2. The estimation parameter is $\theta = [\tau]$. The likelihood function is,

$$L(\theta) = f_\theta(\mathbf{x}) = \prod_{i=1}^{N} (2\pi\sigma^2)^{-1/2} e^{\frac{-1}{\sigma^2}(x[i]-s[i;\theta])^2}, \tag{2.28}$$

and

$$L(\theta) = f_\theta(\mathbf{x}) = (2\pi\sigma^2)^{-\frac{N}{2}} e^{\frac{-1}{\sigma^2}\sum_{i=1}^{N}(x[i]-s[i;\theta])^2}. \tag{2.29}$$

Note that the reference signal is taken as the distribution's mean. The log-likelihood function is,

$$\Lambda_\theta(\mathbf{x}) = -\frac{N}{2}\ln(2\pi\sigma^2) - \frac{1}{\sigma^2}\sum_{i=1}^{N}(x[i]-s[i;\theta])^2. \tag{2.30}$$

Maximizing $\Lambda_\theta(\mathbf{x})$ with respect to θ is the same as,

$$\hat{\theta} = \max_{\theta}\left\{\frac{1}{\sigma^2}\sum_{i=1}^{N}(x[i]-s[i;\theta])^2\right\}, \tag{2.31}$$

and after expanding the summation yields,

$$\hat{\theta} = \max_{\theta}\left\{\frac{1}{\sigma^2}\sum_{i=1}^{N}(x[i]^2 - 2x[i]s[i;\theta] + s[i;\theta]^2)\right\}. \tag{2.32}$$

Note that the reference signal for Rect(n), SinBOC(m,n) and CosBOC(m,n) have constant modulus, so $s[i;\theta]^2$=1, for all θ. This constant modulus assumption does not hold for all Alt-BOC(m,n) signals. Given constant modulus,

$$\hat{\theta} = \max_{\theta} \left\{ \sum_{i=1}^{N} x[i]s[i;\theta] \right\}. \tag{2.33}$$

Solving (2.26) using (2.33) for $\hat{\theta}$ yields,

$$\sum_{i=1}^{N} x[i]s[i;\theta] = \sum_{i=1}^{N} x[i] \frac{\partial}{\partial \theta} s[i;\theta] = 0, \tag{2.34}$$

With a change of variables, $\frac{\partial}{\partial \theta} = -\frac{\partial}{\partial t}$, then (2.34) becomes,

$$\hat{\theta} = \sum_{i=1}^{N} x[i] \frac{\partial}{\partial t} s[i;t] = 0. \tag{2.35}$$

Which shows the optimal estimate for the signal delay τ, in the maximum likelihood sense, is found by solving for when the cross correlation of the incoming signal and the derivative of the spreading code reference model is zero. This provides the basis for the delay locked loop (DLL) code phase discriminator with the tracking kernel equal to the code reference derivative [14],

$$\frac{\partial}{\partial t} s[i;t] \approx \mathrm{w}_{\delta}(\mathrm{t}) = \frac{\mathrm{x}(\mathrm{t}+\delta) - \mathrm{x}(\mathrm{t}-\delta)}{2\delta}. \tag{2.36}$$

Note as $\delta \to 0$, then (2.36) better approximates the derivative.

Since the spreading code sequence is known in advance, the DLL is naturally extended to DSSS signals via the δ-delay Early-Late (EML) discriminator described by [15] and [16]. Refs [17] and [18] analyzed loop performance with respect to δ for both low and high SNR and it was determined that the δ values other than ½ could be optimal. Moreover, [19] describes a scheme to vary δ, ($0 < \delta < 1$), to improve acquisition time and tracking bandwidth.

The code kernel block within Fig. 2-7 generates an EML tracking kernel,

$$w_\delta(\mathrm{t}) = \frac{\hat{\mathrm{x}}(\mathrm{t}+\delta) - \hat{\mathrm{x}}(\mathrm{t}-\delta)}{2}. \tag{2.37}$$

where $\hat{\mathrm{x}}(\mathrm{t}+\delta)$ and $\hat{\mathrm{x}}(\mathrm{t}-\delta)$ are the δ-chip early and δ-chip late spreading code model respectively. Fig. 2-9 shows the δ=½ chip early sequence, an on-time (prompt) sequence, a δ=½ chip late sequence, and the resulting correlation kernel, $w_\delta(\mathrm{t})$. Note how the EML signal approximates the derivative of the prompt signal.

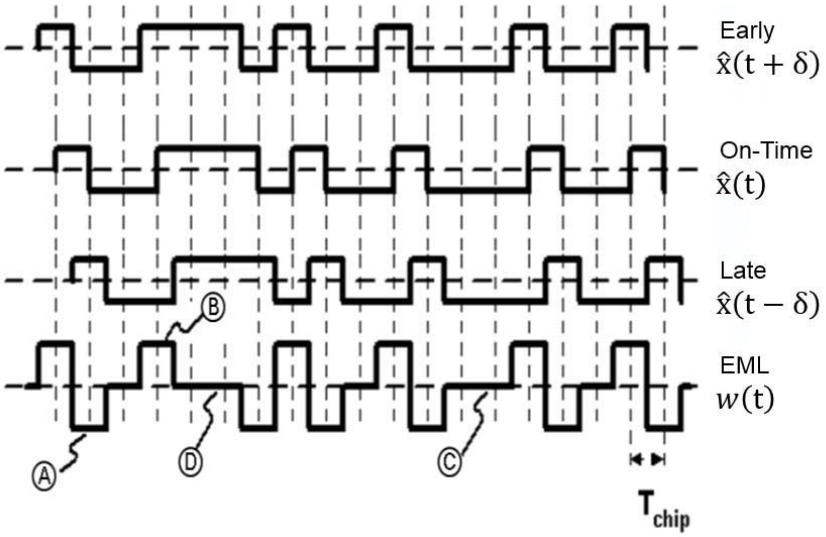

Fig. 2-9: Generation of the Prompt and Early Minus Late (EML) Correlation Signals for GPS L1CA with δ=½T_c.

The prompt, in this case, represents the signal to track, *z(t)*. Consider a discriminant, $d_\varepsilon(\varepsilon) = R_{ZW}(\varepsilon)$, and no multipath. Then $d_\varepsilon(\varepsilon)$ is found by substituting in (2.37) into (2.20) to yields,

$$d_\varepsilon(\varepsilon) = R_{ZW}(\varepsilon) = \frac{1}{T}\int_0^T z(t)\frac{\hat{x}^*(t+\delta-\varepsilon) - \hat{x}^*(t-\delta-\varepsilon)}{2}dt, \tag{2.38}$$

and simplifies to,

$$d_\varepsilon(\varepsilon) = R_{ZW}(\varepsilon) = \frac{R_{X\hat{X}}(\varepsilon+\delta) - R_{X\hat{X}}(\varepsilon-\delta)}{2} * h(t). \tag{2.39}$$

Note that the code phase discriminant is a linear combination of the spreading code auto-correlation, $R_{X\hat{X}}(\varepsilon)$. This is expected, since the EML tracking kernel is a linear combination of the spreading code. Fig. 2-10 shows the EML correlation tracking kernel, *w(t)*, applicable for Rect(n) signals and the resulting phase-discriminator output, *Rxw(ε)*, for variable δ. Note that *w(t)* is even symmetric and *Rxw(ε)* is odd symmetric about $\varepsilon = 0$.

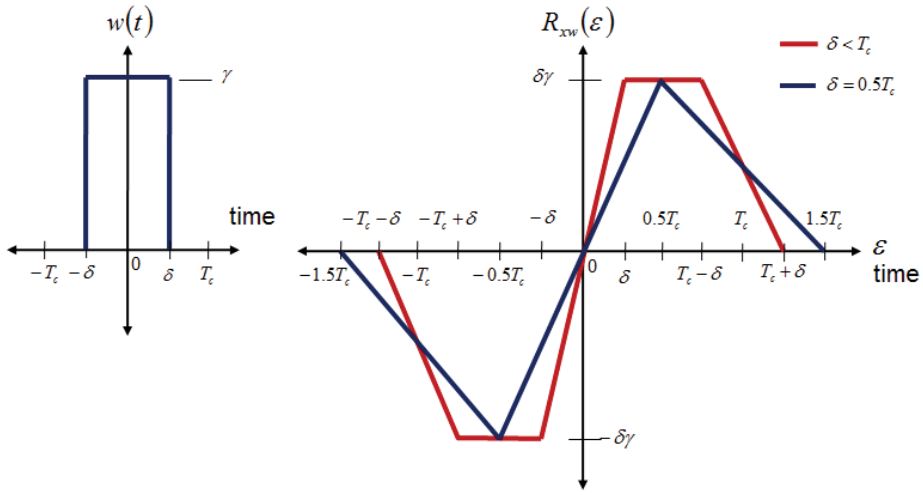

Fig. 2-10: ML-DLL Rect(n) Kernel Shape, *w(t)*, and Discriminator Gain, *Rxw(ε)*.

The correlation function is defined such that prompt is the reference and is held stationary. The axis is such that a delay in time alignment is considered positive and to the right. When the signal arrives earlier than the prompt model, $\hat{x}(t)$, its resulting correlation function is shifted to the right since the signal must be delayed before it aligns with prompt. When the signal arrives later than prompt, its correlation function is shifted to the left since it must advance to align with prompt. Note that the EML correlation function crosses zero at $\varepsilon = 0$ when early and late codes are balanced δ-chip on either side of a prompt code. A detailed analysis of the EML discriminator is presented in Appendix B. When the incoming signal arrives early with respect to the model, more energy is present in the early correlation result and a positive error is produced. The positive error advances the reference phase

and the EML model moves earlier in time. A negative error arises when the incoming signal arrives late and more energy shows up in the late sequence. This delays the reference phase. The error continuously drives the delay-lock-loop, which thereby adjusts the reference phase generator to maintain zero phase error.

Many factors affect the code phase discriminator: Noise, signal amplitude, carrier offset, transition probability and multipath. All factors will reduce the phase detector gain [13]. The dependence on amplitude can be eliminated by employing an automatic gain controller (AGC). The code tracking loop can function with a small carrier offset which is some fraction of the code rate. However, the carrier offset has the effect of reducing the average signal amplitude which reduces the phase detector gain. The code bit transition probability is set by the code bit sequence. A phase discriminator bias occurs due to multipath.

A normalized ML code tracking kernel for SinBOC(1,1) is shown in Fig. 2-11. Note that the kernel weight at t=0.5 has twice the amplitude as the weights at t=0 and t=1.0 since the SinBOC(1,1) signal has a guaranteed transition at t=0.5 and only a 50% transition probability at t=0 and t=1.0. The resulting code phase discriminant is shown in Fig. 2-12. Note the discriminant is zero at both $\varepsilon = 0$ and $\varepsilon = \pm T_c$ which creates a phase tracking ambiguity. This is characteristic of all BOC(m,n) signals and is due to the additional signal transitions within $g(t)$.

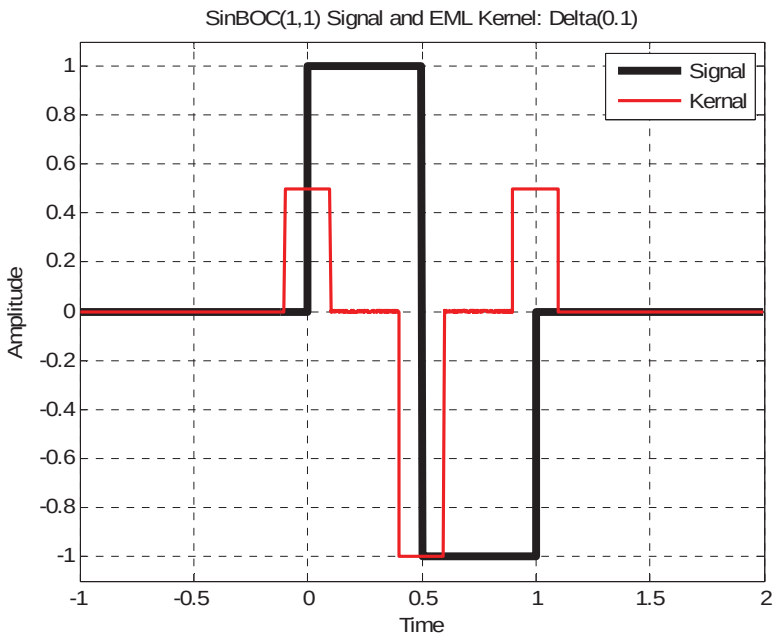

Fig. 2-11: ML SinBOC(1,1) Code Tracking Kernel, *w(t)*.

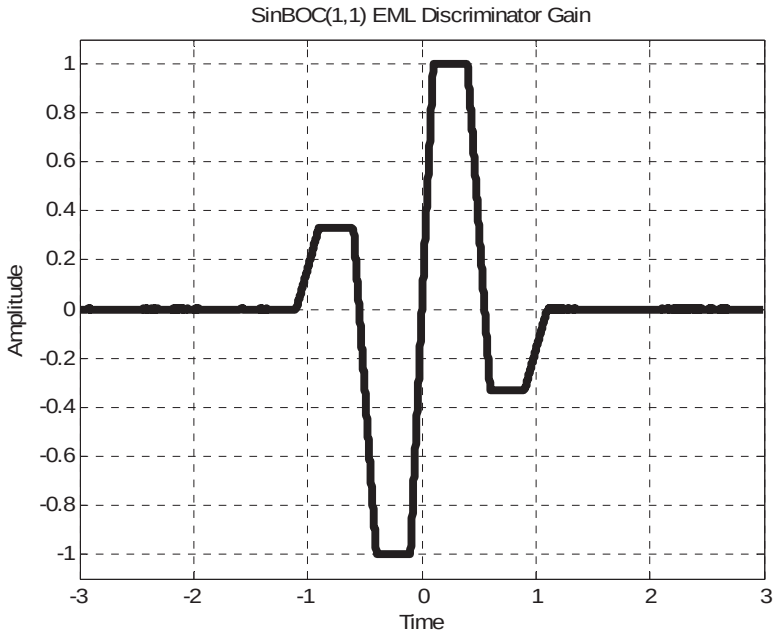

Fig. 2-12: ML SinBOC(1,1) Discriminator Gain, $R_{XW}(\varepsilon)$.

Chapter 3

EXISTING GNSS MULTIPATH MITIAGATION TECHNIQUES

Chapter 1 showed that multipath is a dominant error source for both stand-alone and differential GNSS receivers. Chapter 2 provided the mathematical models for GNSS signals, multipath channels and receivers. These models will be used within this Chapter to survey existing GNSS multipath mitigation techniques. These techniques can be placed into three categories: (1) Fixed Radiation Pattern Antennas; (2) Code tracking via digital signal processing; and (3) Multiple antenna systems. The concentration will be on signal processing techniques since this provides a cost and power effective solution to both consumer and precision GNSS receivers. Precision GNSS receivers for surveying and small platform autonomous control will also use a high quality fixed radiation pattern antenna. Large platform autonomous control can accommodate the high cost, power and size of a multiple antenna scheme.

3.1. *Fixed Radiation Pattern Antennas (FRPA)*

Recall that all GNSS signals are Right Hand Circular Polarized (RHCP) since it is more robust to polarization changes that can occur as signals propagate through the Earth's Ionosphere [10]. The FRPA design goals are: (1) Maximize the antenna gain for left-handed circular polarized signals in the skyward (up) direction; (2) Minimize the gain for signals impinging upon the antenna from below or with Left Hand Circular Polarization (LHCP) since these are reflected (multipath) signals; and (3) minimize phase *variation* across elevation and azimuth which manifests as a position dependent variation in satellite range [20]. The FRPA simply mitigates multipath by attenuating any signals impinging upon the antenna from below. Multipath impinging upon the antenna with a positive

elevation angle is passed. A FRPA element is pictured in Fig. 3-1 and the corresponding received signal gain pattern versus elevation angle is shown in Fig. 3-2.

Fig. 3-1: GNSS Fixed Radiation Pattern Antenna Element.

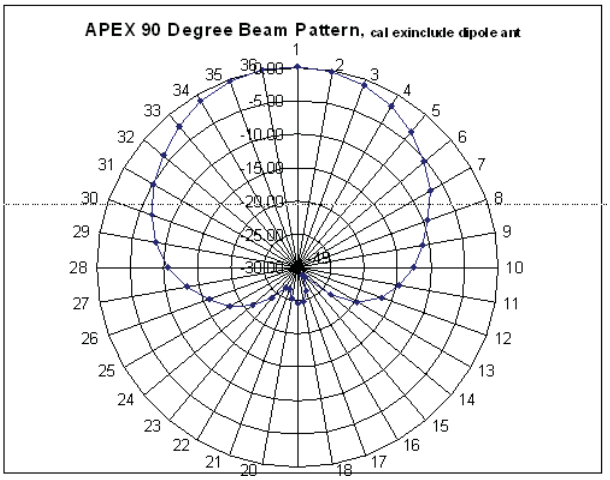

Fig. 3-2: GNSS FRPA Gain versus Elevation Angle (1=10˚, 36=360˚, Radius in dB).

3.2. *Digital Signal Processing Multipath Mitigation Methods*

Consider a coherent code phase discriminant defined as,

$$d_{\varepsilon}(\varepsilon) = R_{XW}(\varepsilon); \tag{3.1}$$

Consider the multipath channel represented in (2.10) such that the received signal consists of a direct path and *L-1* multi-path signals with delay τ_i and path gain α_i *relative* to the direct path,

$$r(t) = \sum_{l=0}^{L} \alpha_l e^{j\phi_l} x(t - \tau_l) + n(t), \tag{3.2}$$

and equivalently,

$$r(t) = x(t) + \sum_{l=1}^{L} \alpha_l e^{j\phi_l} x(t - \tau_l) + n(t). \tag{3.3}$$

Substituting (3.3) into (2.20) and following the computations and assumptions outlined in Chapter 2, the correlation statistics, $\tilde{R}_{XW}(\varepsilon)$ is,

$$\tilde{d}_{\varepsilon}(\varepsilon) = \tilde{R}_{XW}(\varepsilon) = R_{XW}(\varepsilon) + \sum_{i=1}^{L} \alpha_i R_{XW}(\varepsilon - \tau_i) + \mathrm{N}_{NW}. \tag{3.4}$$

Where the summation term in (3.4) is the phase discriminant distortion due to multipath; the tilde denotes signals that include multipath; and the noise term, N_{NW}, is that of (2.22). An EML $\delta = \frac{1}{2}T_c$ code discriminator output when the received signal, *r(t)*, includes multipath is shown in Fig. 3-3. Note how the multipath creates a discriminant bias such that $\tilde{d}_{\varepsilon}(\varepsilon) \neq 0$ when $\varepsilon = 0$. Also note that the magnitude of the bias generally increases with multipath delay, τ_i, but depends upon the magnitude and sign of α_i. It is important to note that the code phase bias can either be positive or negative. This code tracking timing bias generates a time of arrival estimation bias and consequently a GNSS position bias.

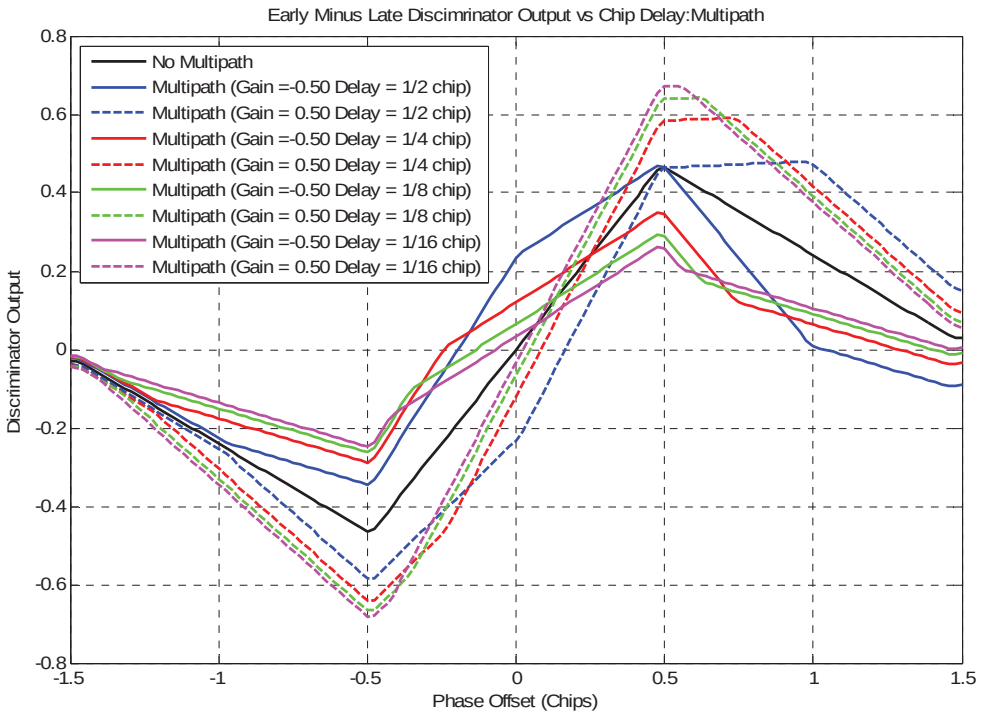

Fig. 3-3: EML Discriminator Multipath Performance with δ=½T_c.

Two primary solution classes exist to remove the multipath distortion via digital signal processing: (1) Parameter estimation of α_0, ϕ_0, and τ_0; and (2) Correlation kernel design of *w(t)*.

Parameter estimators utilize the $R_{Z\hat{X}}(\varepsilon)$ correlation statistic to perform a deconvolution or a projection onto convex sets (POCS). Substituting (3.3) into (2.12) and following the computations and assumptions outlined in Chapter 2, the correlation statistics, $\tilde{R}_{Z\hat{X}}(\varepsilon)$ is,

$$\tilde{R}_{Z\hat{X}}(\varepsilon) = R_{Z\hat{X}}(\varepsilon) + \sum_{i=1}^{L} \alpha_i R_{Z\hat{X}}(\varepsilon - \tau_i) + \mathrm{N}_{N\hat{X}}, \tag{3.5}$$

which can be simplified to,

$$\tilde{R}_{Z\hat{X}}(\varepsilon) = R_{X\hat{X}}(\varepsilon) * h(t) + \sum_{i=1}^{L} \alpha_i R_{X\hat{X}}(\varepsilon - \tau_i) * h(t) + \mathrm{N}_{N\hat{X}}. \tag{3.6}$$

Where the summation term in (3.6) is the distortion of the autocorrelation due to multipath; the tilde denotes signals that include multipath; and the noise term, $\mathrm{N}_{N\hat{X}}$, is that of (2.18). Representing (3.6) in matrix notation yields,

$$\widetilde{\boldsymbol{R}}_{\boldsymbol{Z\hat{X}}} = [\boldsymbol{R}_{\boldsymbol{X\hat{X}}} * h(t)]\boldsymbol{q} + \mathbf{N}_{\boldsymbol{N\hat{X}}}, \tag{3.7}$$

where $\boldsymbol{R}_{\boldsymbol{X\hat{X}}}$ is an NxN sampled autocorrelation matrix that spans -T_c to T_c in intervals of any multiple of the receiver sampling period; N is the number of signal paths to estimate; $\boldsymbol{q}$ is an Nx1 column vector of estimated path gains, and $\mathbf{N}_{\boldsymbol{N\hat{X}}}$ is an Nx1 noise vector. The channel impulse response can be estimated as,

$$\hat{\boldsymbol{q}} = \left(\hat{\boldsymbol{\sigma}}^{\boldsymbol{2}}\boldsymbol{I} + \boldsymbol{R}_{\boldsymbol{X\hat{X}}}^{\boldsymbol{T}}\boldsymbol{R}_{\boldsymbol{X\hat{X}}}\right)^{-\mathbf{1}}\boldsymbol{R}_{\boldsymbol{X\hat{X}}}^{\boldsymbol{T}}\widetilde{\boldsymbol{R}}_{\boldsymbol{Z\hat{X}}}, \tag{3.8}$$

or the solution can be solved by iteration. $\hat{\boldsymbol{\sigma}}^{\boldsymbol{2}}$ is the estimated noise variance. The disadvantage of these solutions are their complexity and scale considering each GNSS receiver processes hundreds of channels and each code phase tracking correlation statistic will be replaced by a 2N+1 correlators to provide the sampled autocorrelation function. Moreover, a complex matrix inversion is required every loop update which is about 1ms for GPS L1CA. A commercial GPS L1CA receiver utilizing this method is the Multipath Estimating Delay Locked Loop (MEDLL) [21].

The Teager-Kaiser (TK) non-linear energy operator has been applied to the autocorrelation statistic as a means to estimate multipath delay. The real, discrete time TK operator with sample rate, T_s, is,

$$\Psi_{\delta}[x(n)] = x^2(n) - x(n-1)x(n+1). \tag{3.9}$$

A plot of the TK operator on SinBOC(1,1) with multipath is shown in Fig. 3-4.

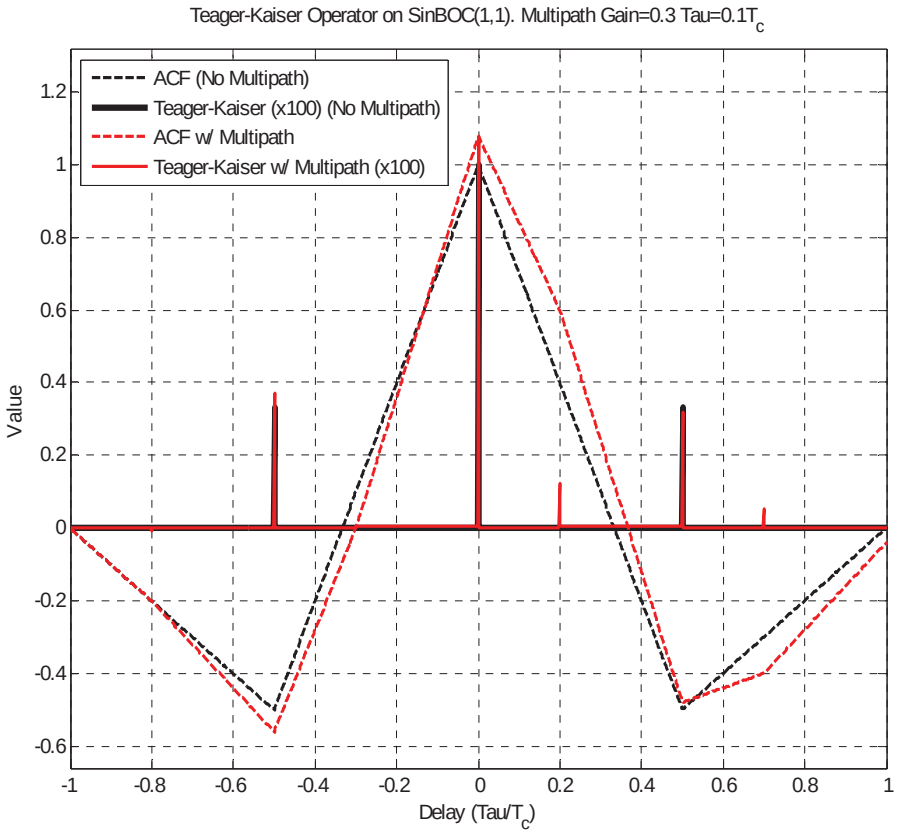

Fig. 3-4: Teager-Kaiser Operator on SinBoc(1,1).

Multipath components are resolvable to within the windowed resolution of the sample period. This scheme is more susceptible to noise due to its non-linear operation. Additional multipath mitigation schemes that utilize the sampled autocorrelation or a linear combination of the sampled autocorrelation. The most notable is [22] which included constraints on the discriminant to set minimum gain and resolve the BOC discriminator ambiguity.

The multipath distortion can also be reduced by limiting the ROS for the code tracking kernel, *w(t)*, such that $R_{XW}(\varepsilon - \tau_i) = 0$ for $\tau_i < \delta$ as in [23], [24], and [25]. The

Narrow Correlator sets the EML δ parameter to less than 0.5 chips with a limit set by the signal bandwidth (as described in section 3.3) [13,14]. The W-Correlator builds a W-shaped correlation waveform for a linear combination of the EML signal [18]. The Strobe Correlator blanks (sets to zero) the correlation waveform when the PN does not transition [19]. The waveform design correlator utilizes a linear combination of EML to produce a variety of correlation waveforms [21]. Most of the methods so far build the correlation waveform utilizing a linear combination of EML and are even symmetric. The multipath distortion can be reduced by limiting the ROS for the code tracking kernel *w(t)* as in the coherent discriminant case. Fig. 3-5 shows the EML discriminator multipath performance with $\delta=T_c/16$ for various multipath conditions.

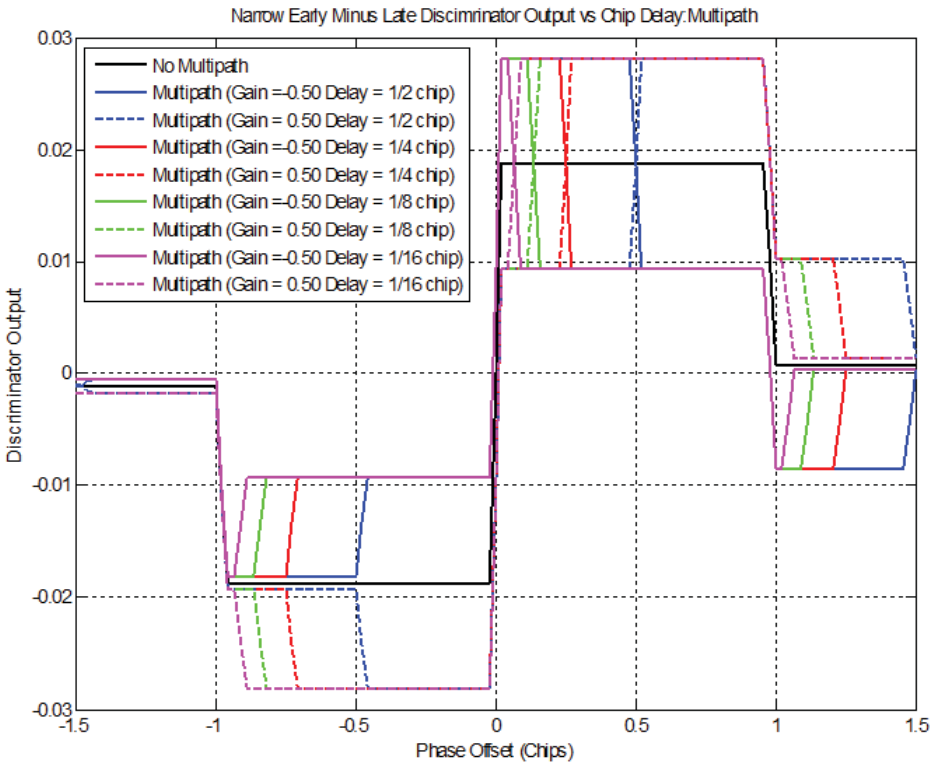

Fig. 3-5: EML Discriminator Multipath Performance with $\delta=T_c/16$.

These solutions have the disadvantage of lowering the discriminator gain to δ. Until [5]- [6], *w(t)*, has been limited to linear functions of the spreading code which produced an even functioned *w(t)*.

The multipath distortion can also be reduced by designing the code tracking kernel, *w(t)*, to meet specified objectives such as a non-ambiguous BOC discriminant [26] and or a desired discriminant shape [22]. The unambiguous S-curve shaping technique starts with an S-curve prototype and solves for the linear combination of the spreading code. This technique mitigates medium and long range multipath delay but has poorer result for delay spread of less than 0.4 chips. An advantage of this technique is that it generates an unambiguous correlation kernel for CosBOC; however, the resulting correlation kernel pulses are narrow and the effects of a band limited precorrelation filter were not considered. The ROS is greater than 3 chips.

Consider a dot-product discriminant defined as,

$$d_{\varepsilon}(\varepsilon) \;=\; R_{Z\hat{X}}(\varepsilon)R^{*}_{XW}(\varepsilon). \tag{3.10}$$

Substituting (3.10) into the non-coherent discriminant in (3.1) yields,

$$\begin{aligned}\tilde{d}_{\varepsilon}(\varepsilon) \;=\; & \tilde{R}_{Z\hat{X}}(\varepsilon)\tilde{R}^{*}_{XW}(\varepsilon) \\ & = \left\{ R_{Z\hat{X}}(\varepsilon) + \sum_{i=1}^{L_n} \alpha_i \, R_{Z\hat{X}}(\varepsilon - \tau_i) \right\} \\ & \cdot \left\{ R^{*}_{XW}(\varepsilon) + \sum_{i=1}^{L_n} \alpha_i^{*} \, R^{*}_{XW}(\varepsilon - \tau_i) \right\},\end{aligned} \tag{3.11}$$

and expanding yields,

$$\tilde{d}_{\varepsilon}(\varepsilon) = R_{Z\hat{X}}(\varepsilon)R^{*}_{XW}(\varepsilon) + R_{Z\hat{X}(\varepsilon)}\sum_{i=1}^{L_n}\alpha_i^{*}\,R^{*}_{XW}(\varepsilon-\tau_i)$$

$$+ R^{*}_{XW}(\varepsilon)\sum_{i=1}^{L_n}\alpha_i\,R_{Z\hat{X}}(\varepsilon-\tau_i) \qquad (3.12)$$

$$\pm\sum_{i=1}^{L_n}|\alpha_i|^2\,R_{Z\hat{X}}(\varepsilon-\tau_i)R^{*}_{XW}(\varepsilon-\tau_i).$$

The multipath distortion can be reduced by limiting the ROS for the code tracking kernel, *w(t)*, such that $R_{XW}(\varepsilon-\tau_i) = 0$ for $\tau_i < \delta$.

Chapter 4

PROPOSED NOVEL MULTIPATH MITIGATING ALGORITHM

Chapter 2 showed that the derivative of the spreading code is the maximum likelihood (ML) code bit transition estimator when not in the presence of multipath; and it sets the notion that the discriminator should only process the signal transitions [16]. However, this only holds for even symmetric correlation kernels or for equal probable transition and non-transitions for asymmetric correlation kernels. A new multipath mitigating algorithm is introduced that utilizes a non-symmetric correlation kernel that considers non-equal transition and non-transition probabilities. This algorithm is compared to existing techniques and assessed on signal tracking bias, variance and robustness.

4.1. *Non-Symmetric Correlation Kernel Design*

Four different correlation kernels exist corresponding to the four transitions states: (1) Low to high transition; (2) high to low transition; (3) no transition but model low; and finally (4) no transition but model high. These kernels do not have to be the same; nor are they constrained to have equal energy (sum of kernel weight squared) since separate accumulators can be maintained for each state and the results scaled based upon the ratio of kernel weights. Recall that the EML correlation signals are even symmetric about the model transition point as shown in Fig. 2-9. This symmetry is not required and other correlative kernels that are non-symmetric can be used. If not symmetric about the zero phase error point, a balancing mass is required otherwise a phase discriminator bias will result: The phase discriminator will have a non-zero output corresponding to zero phase error. Phase error bias is eliminated for non-symmetric transition kernels by introducing a

balancing mass such as a non-transition kernel. Consequently, non-symmetric transition kernels accompanied by non-transition kernels must account for transition and non-transition probabilities otherwise a phase discriminator bias will result. Note that in this case non equal transition and non-transition probabilities will result in a phase discriminator bias whereas in the EML case only the discriminator gain is reduced and no bias occurs.

Each correlation kernel can be formed using the upper bits of the code NCO, which defines the region of support, to address a look-up-table for the kernel weights. A design goal for a correlative discriminator with multipath mitigating properties is to: (1) Provide an unbiased code-phase error estimate insensitive to multipath; (2) Provide an unambiguous discriminator; and (3) Provide a wide linear-operating region. Note that multipath is inherently time delayed and can cause a positive or negative code phase error bias depending upon if the multi-path phase has the same or opposite sign as the main signal path respectively.

Fig. 4-1 and Fig. 4-2 show the code signal, ***B***, undergoing a low-to-high (L-H) and high-to-low (H-L) transition respectively. Furthermore, both figures show the applicable correlation kernels.

Referring to Fig. 4-1, the track reference signal contains the Non-Transition correlation kernel comprised of pulse A4, and L-H transition correlation kernel consisting of pulses A1, A2 and A3. The code takes a L-H transition coincident with regions A_1 and the A_3. Pulse A_4 is shown aligned to the end of a non-transition code chip (one for which the sign does not change in the next chip). Pulse A_2 follows pulse A1 and is of opposite

sign. When the track signal ***A*** is aligned in phase to the code signal ***B***, the area A_1 is equal to A_2 and the area A_3 is equal to A_4. This equalization is denoted as a mass balance.

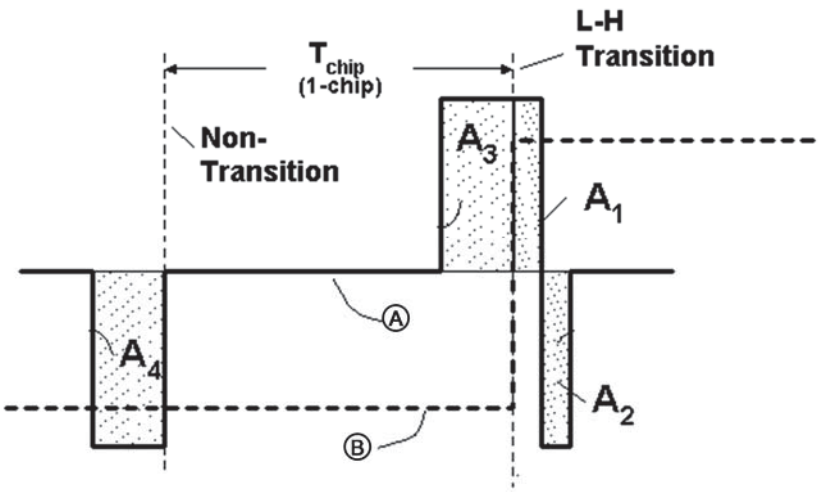

Fig. 4-1: Low to High Transition with Perfect Track Phase Alignment.

Referring to Fig. 4-2, we depict the code having opposite transition and non-transition signs relative to those shown in Fig. 4-1. Again, the area A_1 is balanced to A_2 and the area A_3 is balanced to A_4 when the phase of the track reference signal matches that of the code signal.

The track reference signal ***A*** acts as a multiplier and enable for the code signal ***B***. The product is zero wherever the track reference signal ***A*** is zero, and non-zero wherever the correlation kernels exist. The non-zero portion has amplitude that is the product of both the amplitude of the code and the amplitude of the track reference, including sign.

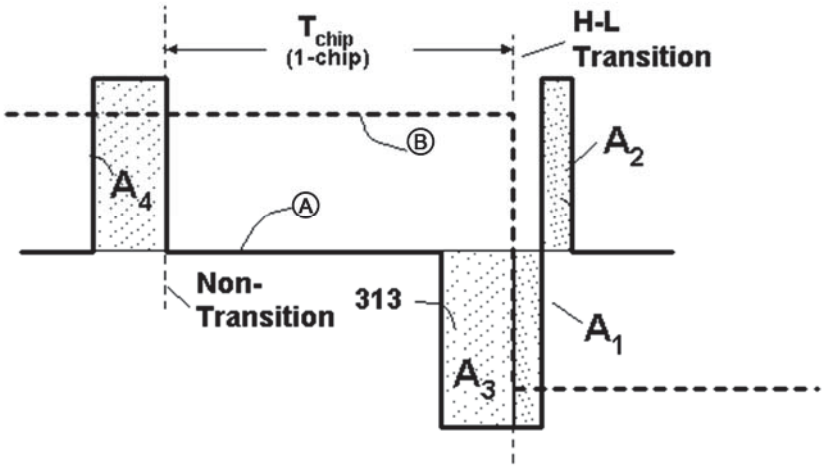

Fig. 4-2: High to Low Transition with Perfect Track Phase Alignment.

Fig. 4-3 shows the product of the track reference signal and the code signal when both are phase aligned. The product pulses have areas M_1, M_2, M_3 and M_4 respectively and are referred to masses. The areas of these masses are proportional to the corresponding areas A_1, A_2, A_3 and A_4 of the track reference signal, but are scaled in amplitude by the height of the code signal. The masses also contain some sign reversals relative to the original pulses due to multiplication by the code signal.

Fig. 4-3 holds for both L-H transitions, and H-L transitions. Regardless of the direction of the code transition (H-L or L-H), M_2 and M_3 are negative while M_1 and M_4 are positive. This is by design. A further property relates to the areas of M_1, M_2, M_3 and M_4. When the track reference signal and the code signal are aligned in phase, the areas cancel to zero when summed together. Specifically,

$$(M_2 = -M_1), \text{ and } (M_3 = -M_4), \tag{4.1}$$

so,

$$M_1 + M_2 + M_3 + M_4 = 0. \tag{4.2}$$

Thus, the track reference signal is mass-balanced about the code signal so as to yield a zero summation.

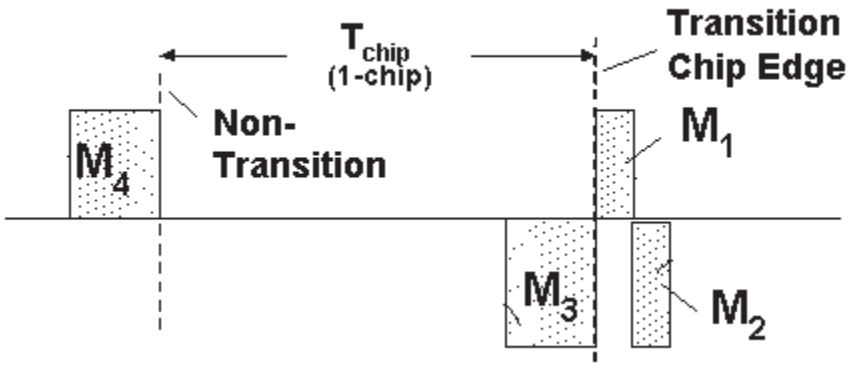

Fig. 4-3: Mass Balance Kernel with Perfect Track Signal Phase Alignment.

Fig. 4-4 shows the result of delaying the track reference signal relative to the code (the track reference signal lags the code signal). The masses M_1, through M_4 of the track signal are seen to shift right relative to the code's level transition. The mass gains positive area so that M_1 increases by a positive amount. The mass loses negative area, so that M_3 also effectively increases by a positive amount. Thus, there is a net positive increase in mass (or correlation) relative to Fig. 4-3. Precisely,

$$M_1 + M_2 + M_3 + M_4 = 2\varepsilon, \tag{4.3}$$

where ε is the increase in the area of M_1 and also the increase in area of M_3.

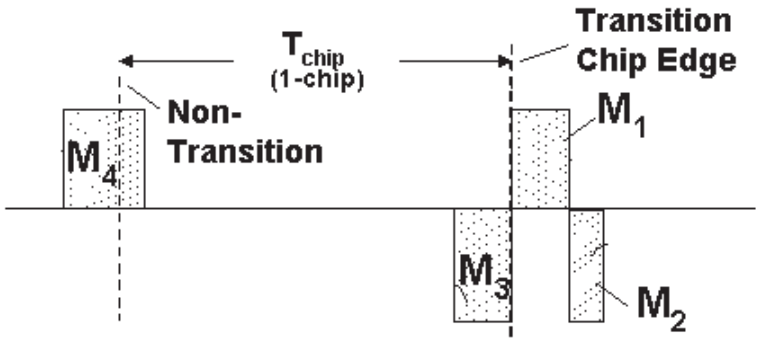

Fig. 4-4: Late Track Signal.

Fig. 4-5 shows the case when the track reference signal leads the code signal. The masses M_1, through M_4 are seen to shift left relative to the code sign transition. The mass loses positive area so that M_1 decreases in value. The mass M_3 gains negative area. Thus, there is a net decrease in mass relative to Fig. 4-3. Precisely,

$$M_1 + M_2 + M_3 + M_4 = -2\varepsilon, \tag{4.4}$$

where is ε the decrease in area of both M_1 and M_3.

The mass summations represent the process of computing a correlation between the track-reference signal and the code signal. This acts as a measure of phase error between the track reference signal and the code signal.

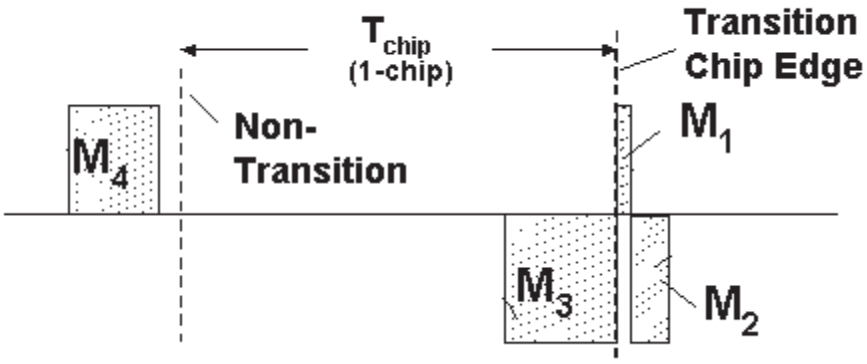

Fig. 4-5: Early Track Signal.

Referring to Fig. 4-6, the chip transition acts as the zero-phase reference. Phase error is thus measured by a displacement, ε, relative to zero. Positive, ε, is taken so the code signal leads the track reference signal. At zero phase error (ε=0), the masses M_1, through M_4 balance. For some positive constants γ and σ, the relationship is

$$\begin{aligned} M_1 &= \gamma/2, \\ M_2 &= -\gamma/2, \\ M_3 &= \sigma/2, \\ M_4 &= -\sigma/2, \end{aligned} \tag{4.5}$$

so that,

$$M_1 + M_2 + M_3 + M_4 = 0. \tag{4.6}$$

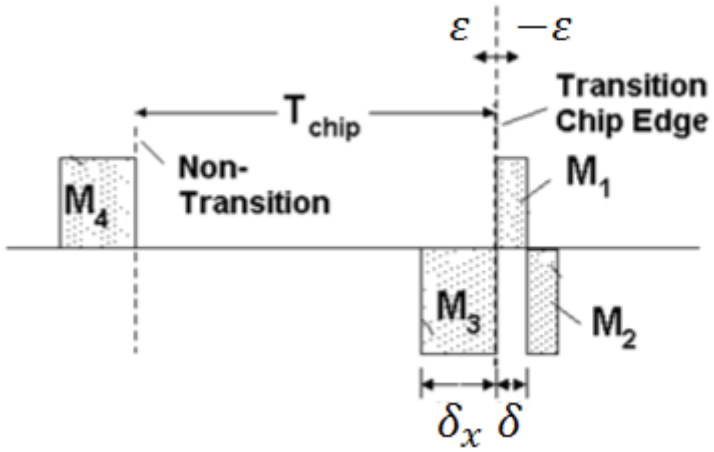

Fig. 4-6: Non-Transition and Transition Correlation Kernels.

Fig. 4-7 shows a plot of the Mass Balance (MB) discriminator as a function of true phase error. Notice that the point of zero phase error gives a zero mass balance and hence zero phase discriminator. The phase discriminator curve for small phase errors is of constant slope (linear) and is positive for positive phase error (code phase leading reference phase). This linear region is denoted as segment ***B***. It reaches a maximum of $\sigma\delta_x$ on the right-hand end of the linear region, at a phase error of $\varepsilon=\delta_x$. As true phase error continues in the positive direction, the discriminator levels off to a constant value as shown by segment ***A***. Similarly, moving left, it reaches a peak negative value of $-\gamma\delta$ at a phase error of $\varepsilon=\delta$ and then slopes back to zero as phase error becomes more negative, reaching zero at -2δ. The discriminator becomes zero over the region ***C*** outside of a phase error of -1 chip. Due to its asymmetric shape, it is referred to as an asymmetric correlative-phase-discriminator.

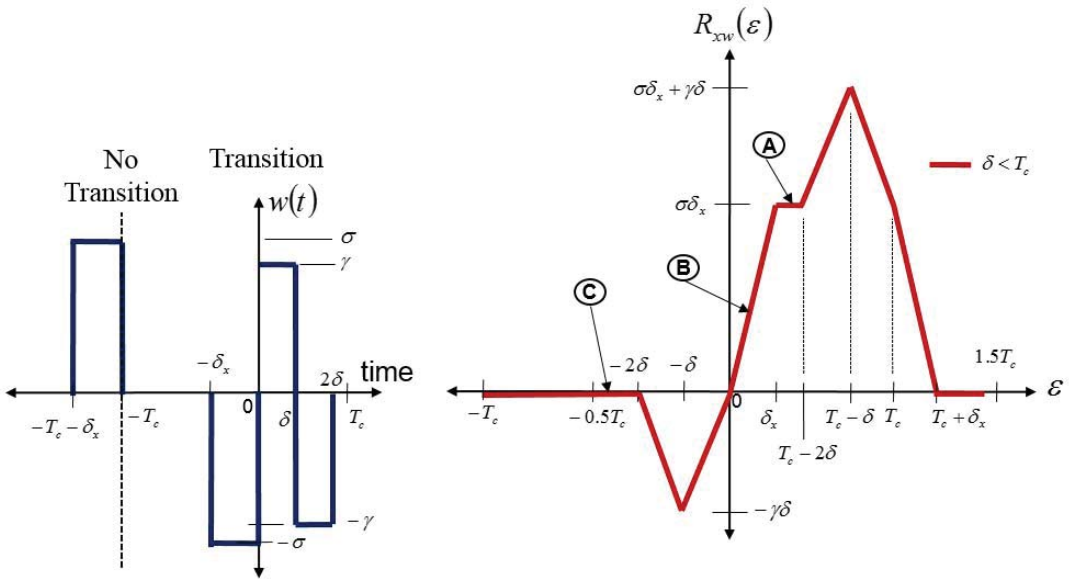

Fig. 4-7: Code Phase Discriminator Gain, *Rxw(ε)*.

The Mass Balance discriminant is an unbiased phase error estimator and conventional linear control techniques can be utilized to maintain near zero phase error. During tracking, the phase error will typically reside within the linear portion, and linear control assumptions hold true. Even larger positive phase errors, up to one chip in duration, are zeroed out by a feedback control loop since the phase discriminator remains non-zero and positive when phase error is positive. However, due to the asymmetric nature of the phase discriminator curve, reverse is not true. If phase error exceeds a value of -2δ, the phase discriminator goes to zero where it remains for negative phase errors up to -1 chip. In this region the feedback control will not be able to zero out the phase error. A solution for this problem is provided in [6] but a detailed theoretical performance analysis will be the focus for future work.

Recall that the phase discriminator is calculated by correlating over many code chips, some with transitions in sign and some without. Appendix A shows that the number of code transitions does not match the number of non-transitions for every GPS L1CA code

sequence. For an unbiased discriminant, the mass balance must take into account this imbalance.

Refer again to Fig. 4-6, recall the mass M_3 at the sign transition must balance the mass M_4 at the non-transition whenever the track reference signal and code signal are aligned. Consider all chips within the code sequence, then the mass balance becomes,

$$N_T \times M_1 + N_T \times M_2 = 0, \tag{4.7}$$

and

$$N_T \times M_3 + N_N \times M_4 = 0, \tag{4.8}$$

where N_T is the number of code sign transitions and N_N is the number of non-transitions.

When $M_2 = -M_1$ and $M_3 = -M_4$ a zero mass balance is not achieved unless $N_N = N_T$. Therefore, a scale factor α is introduced for which the final correlation of all non-transitions is multiplied. The scale factor α is given by,

$$\alpha = (N_T \times |M_3|) / (N_N \times |M_4|), \tag{4.9}$$

When $|M_3| = |M_4|$, which is the case described so far and also convenient for implementation, then,

$$\alpha = N_T / N_N. \tag{4.10}$$

The mass balance then becomes,

$$N_T \times M_1 + N_T \times M_2 = 0, \tag{4.11}$$

and

$$N_T \times M_3 + \alpha \times N_N \times M_4 = 0, \tag{4.12}$$

which holds true for any nonzero N_N and N_T. Generally, many arrangements of scaling the transition correlation to the non-transition correlation are possible. For example, choose any numbers β_1 and β_2 such that,

$$\beta_2 / \beta_1 = \alpha, \tag{4.13}$$

and then multiply the transition correlation by β_1 and the non-transition correlation by β_2 to satisfy the mass balance. This will give a mass balance,

$$\beta_1 \times (N_T \times M_1 + N_T \times M_2) = 0, \tag{4.14}$$

and

$$\beta_1 \times N_T \times M_3 + \beta_2 \times N_N \times M_4 = 0, \tag{4.15}$$

In summary, to produce an unbiased correlative-phase-discriminator with a non-symmetric transition kernel, two correlations must be performed and summed together with appropriate scaling. One correlation operates over code sign transitions and the other operates over non-transitions. An alternative is to move the mass M_4 to the lie within a transition chip and perform correlations only on transition chips. Recall the asymmetric nature of the correlation kernels necessitates the counting of code transitions and code non-transitions to achieve an unbiased phase estimate. Moreover, the GPS L1CA code has an unbalanced number of transitions to non-transitions as described in Appendix A. Therefore, any asymmetrical correlation kernel with the balancing mass located outside the current chip boundary must utilize the N_T and N_N found in Table A-2 of Appendix A.

4.2. *Simulation and Real Receiver Results*

For implementation purposes, it is convenient to express the Mass Balance (MB) kernel as the function of two parameters, MB(δ, δ_x-δ). Fig. 4-8 plots the MB kernels for various parameters δ and δ_x with γ=σ=1. The δ=1/16 EML kernel is also plotted for comparison. Note that a MB(1/32,1/32) kernel has the same ROS as the δ=1/16 EML kernel; therefore multipath performance comparisons will utilize MB(1/32,•). Plots of the corresponding normalized discriminants, without multipath, are shown in Fig. 4-9.

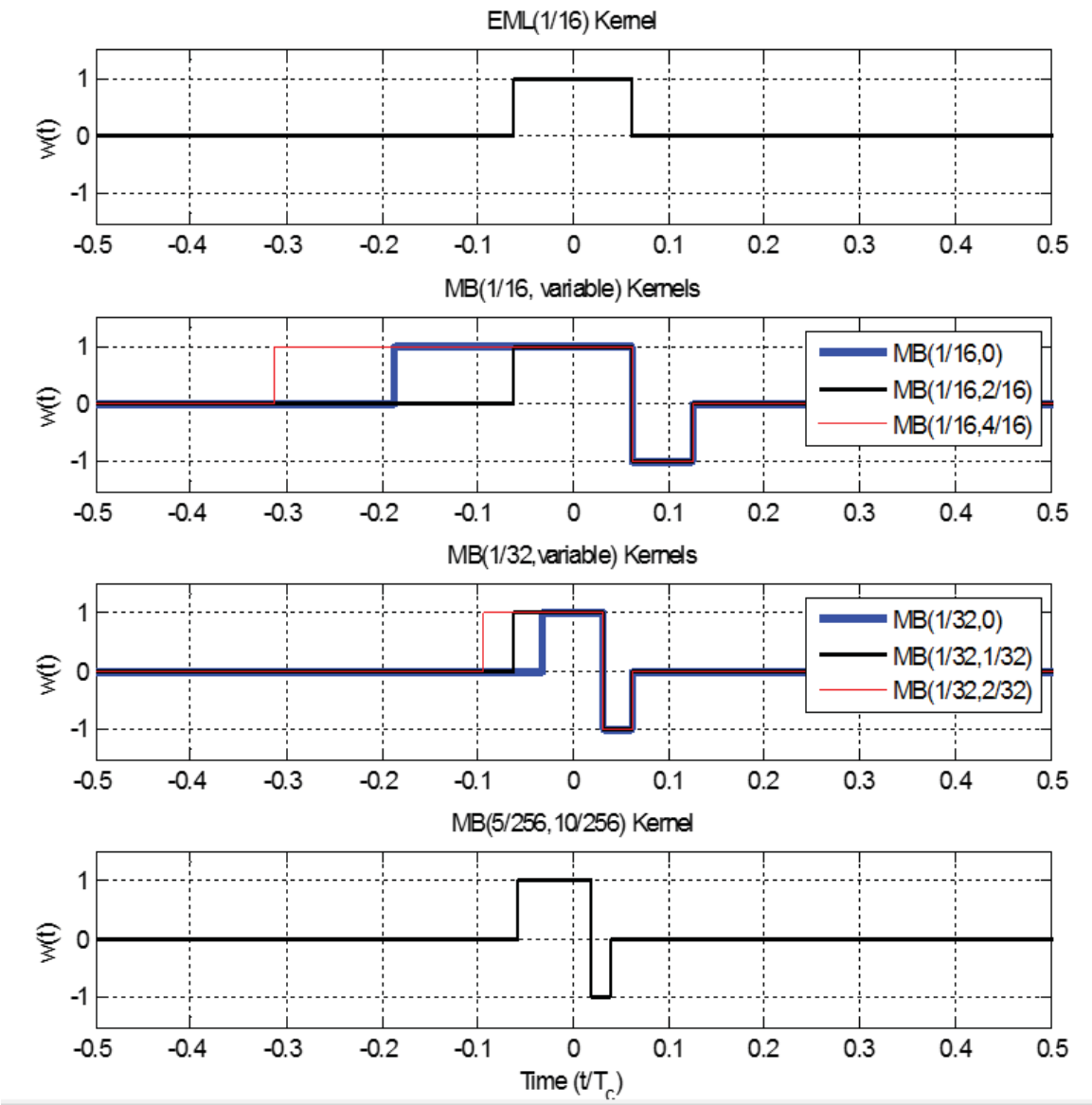

Fig. 4-8: Various Mass Balance Correlation Kernels.

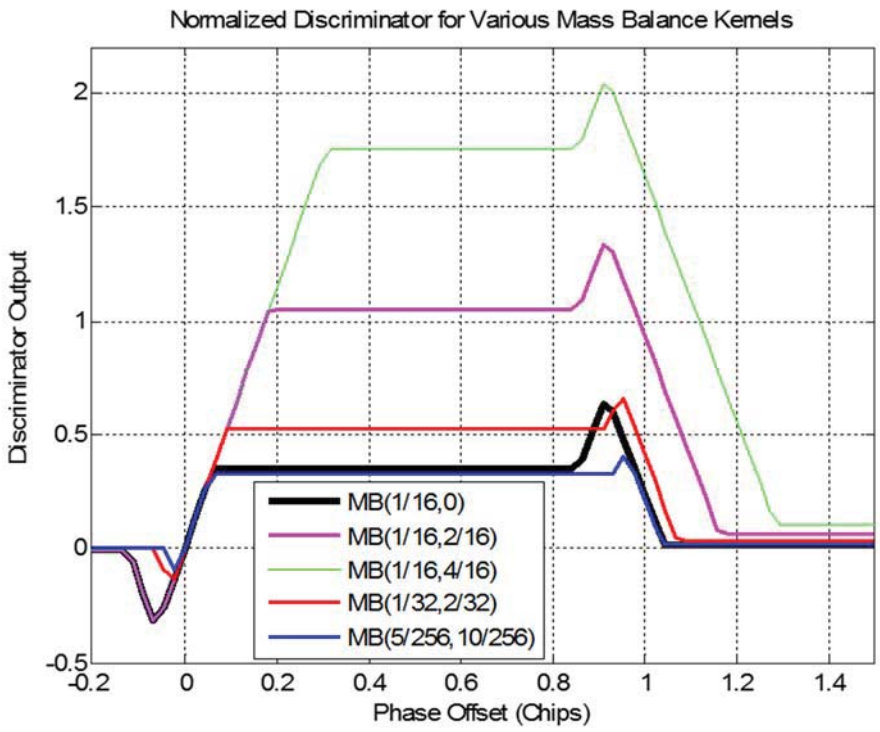

Fig. 4-9: Various Mass Balance Discriminators.

Comparing Fig. 2-10 to Fig. 4-7 and utilizing equivalent kernel ROS, the following observations can be made: (1) The MB Kernel has a lower discriminant gain than the EML kernel for $\varepsilon<0$; (2) the MB has a shorter tracking region when $\varepsilon<0$; however the MB tracking region can be extended [6]; (3) The MB and EML noise variances are equivalent when $N_T = N_N$ or when unbalanced α scaling is applied as in (4-10); (4) The MB kernel can be programmed to have a larger discriminant gain than the equivalent EML for $\varepsilon>0$.

Mass Balance MB(1/32,2/32) multipath simulation results are shown in Fig. 4-10 with zoomed in results around $\varepsilon=0$ shown in Fig. 4-11. Zoomed in results for the equivalent δ=1/16 EML kernel (Fig. 3-5) are shown in Fig. 4-12. Note that MB Kernels have better multipath performance given the same ROS. This is due to the odd-symmetric leading pulses of the MB kernel (M_1 and M_2 of Fig. 4-6).

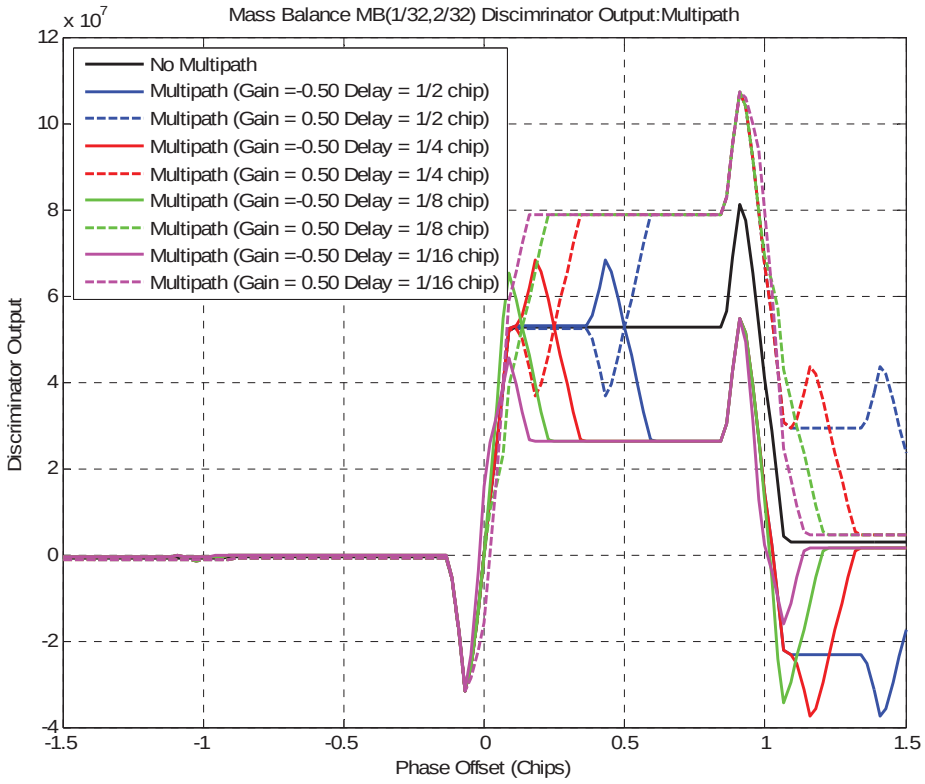

Fig. 4-10: MB(1/32,2/32) Discriminant Multipath Performance.

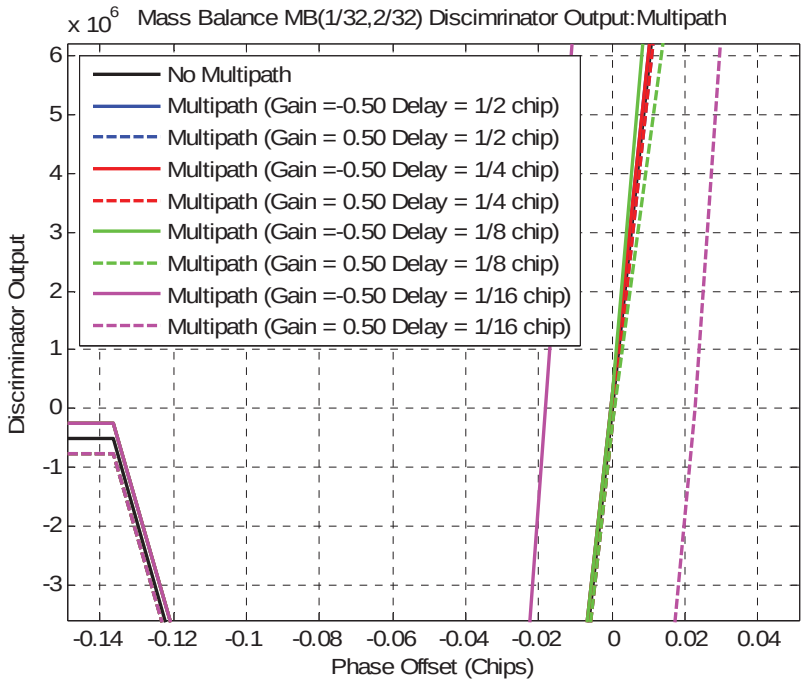

Fig. 4-11: MB(1/32,2/32) Discriminant Multipath Performance (Zoomed).

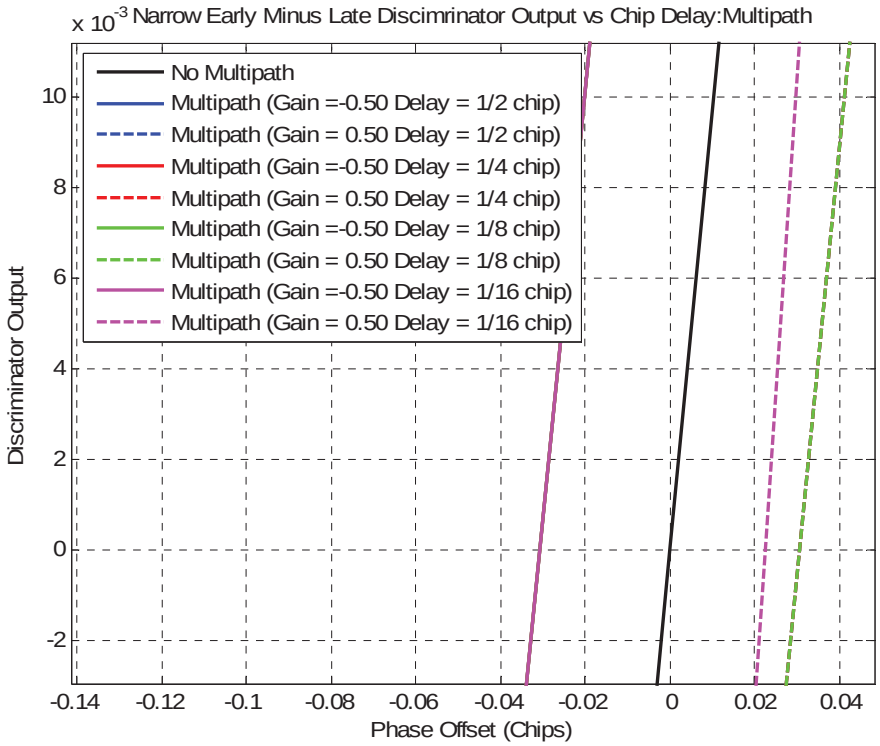

Fig. 4-12: EML $\delta=T_c/16$ Multipath Performance.

Fig. 4-13 demonstrates the multi-path mitigating performance for MB(5/256,10,256) and $\gamma=\sigma=1$. Note the reduced tracking range for delayed signals.

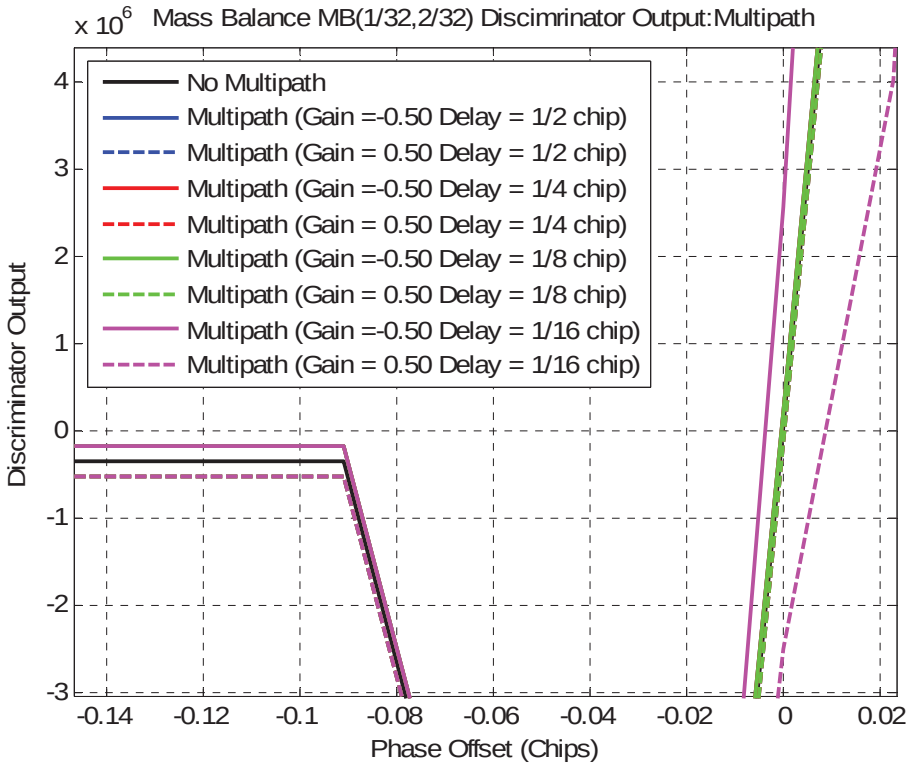

Fig. 4-13: MB(5/256,10/256) Discriminant Multipath Performance (Zoomed).

Real GPS L1CA receiver data taken at 33°36.8682'N and 111°55.0309'W at 4pm local time (UTC-7) is shown in Fig. 4-14 and 4-15. The data was simultaneously collected using the same antenna and splitting the RF signal into two receivers configured in GPS L1CA autonomous mode. The North and East error plots for a receiver programed for $\delta=0.5$ EML is shown in Fig. 4-14. The receiver utilizing a Mass Balance kernel with $\delta=T_c/8$ and $\delta_x=0.4T_c$ is shown in Fig. 4-15. The Mass Balance kernel provided significant performance improvement over the EML kernel processing actual GPS L1CA signals.

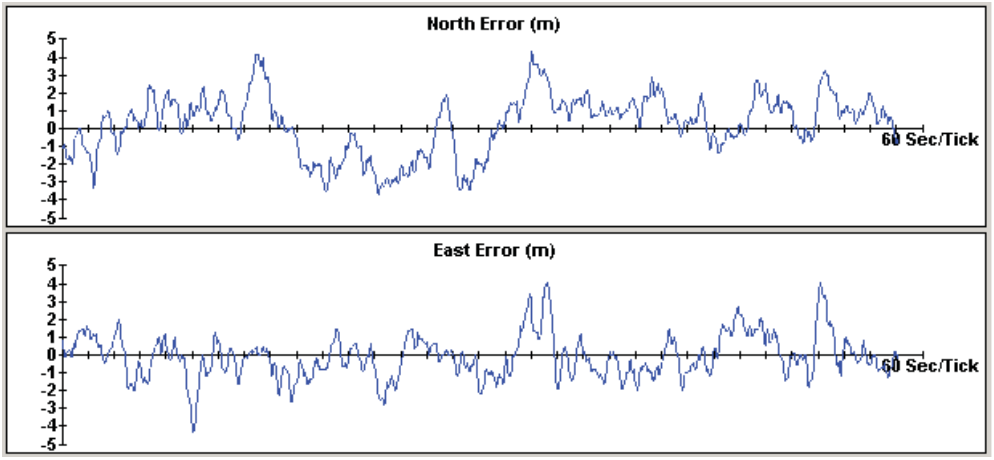

Fig. 4-14: North and East Errors (m) When Using δ=0.5 EML Discriminant.

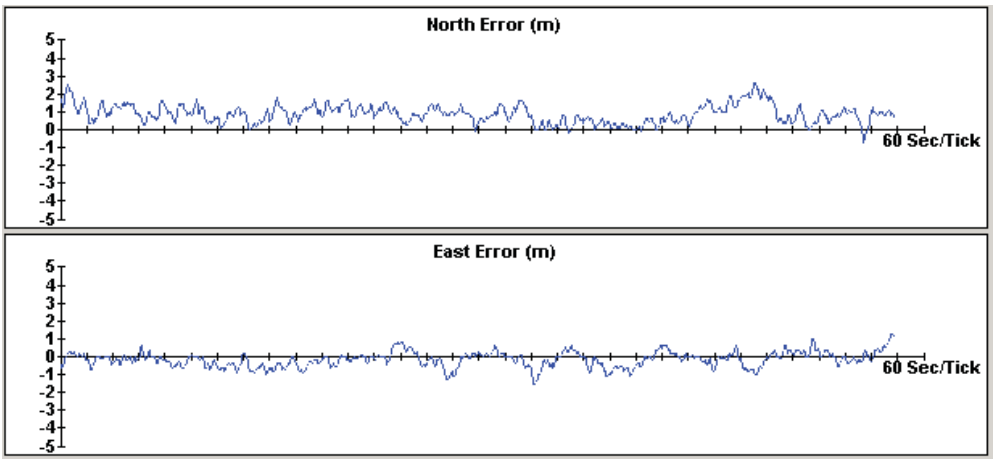

Fig. 4-15: North and East Errors (m) When Using MB Discriminant.

4.3. *Efficient Implementation Methods*

Fig. 4-16 illustrates a method to implement an arbitrary kernel, *w(t)*, utilizing a 2^N by M bit memory. The N bit address, $A_{N-1...}A_o$, consists of a 2 bit transition state and the upper N-2 bits of the code NCO. The M bit memory output, $D_{M-1...}D_o$, consists of the kernel weight associated with the fractional code NCO phase and transition state. The 2 bit transition state, $A_{N-1,}A_{N-2}$, is defined as 0 when not in a transition region and the model is low; 1 when not in a transition region and the model is high; 2 when in a code bit transition region going from high to low; and finally, 3 when in a code bit transition region going

from low to high. Other transition state encodings are possible including its elimination. The memory is referred to as the *Kernel Map*. This implementation enables the kernel to compensate for channel distortion, perform code pulse matched filtering or store the current kernel weights for a real-time adaption scheme.

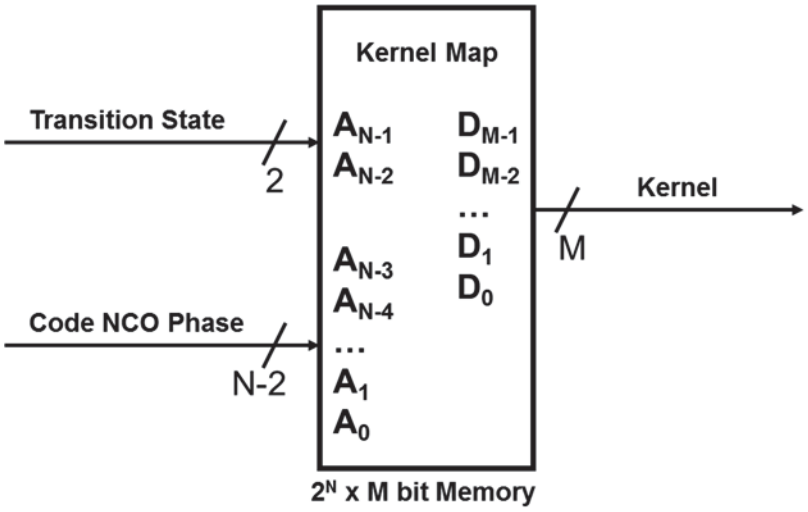

Fig. 4-16: Kernel Map Implementation.

The *XTionSum* and *NoXTionSum* metrics used to form the Mass Balance discriminator are obtained using the following steps:

1. Load the *Kernel Map* memory after a system reset.
2. Every sample clock update the Code NCO.
3. Form address bits $A_{N-1...}A_o$ by concatenating the 2 bit state transition value and the upper N-2 bits of the fractional code NCO phase.
4. Read the *Kernel Map* memory to obtain the kernel weight.
5. Calculate *Mass* = kernel weight times the baseband or near baseband sample.
6. With the transition state coding described above, the A_{N-1} address bit is set when the NCO phase is within a code bit transition region. When the code phase is within

a bit transition region, add *Mass* to the transition accumulator, *XTionSum*; otherwise add *Mass* to the *non*-transition accumulator, *NoXTionSum*.

7. At the end of the correlation accumulation period, usually at the end of each PRN epoch for GPS, copy the *XTionSum* and *NoXTionSum* values into registers *XTionReg* and *NoXTionReg* and then reset both *XTionSum* and *NoXTionSum* to zero. The *XTionReg* and *NoXTionReg* registers are used by the processor for subsequent processing of the Mass Balance discriminator.

The Mass-Balance transition and non-transition kernels as well as the EML kernel are simple kernels which can be efficiently implemented within combinational logic. The Mass-Balance transition and non-transition kernels were shown in Fig. 4-6 and are redrawn in Fig. 4-17 and Fig. 4-18 to show how they are a function of the total code phase. The kernel weights are determined by comparing the code phase against the phase boundaries P0 through P5. The transition kernel is initially 0 at zero code phase and becomes 1 at code phase P1. It remains 1 through code phase P2, P3, P0 and P4. The kernel transitions to -1 at P4 and remains -1 until P5. The kernel transitions back to 0 at P5.

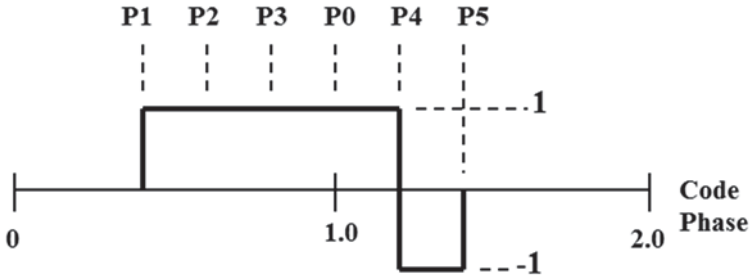

Fig. 4-17: MB Transition Correlation Kernel Mapped to Total Code Phase.

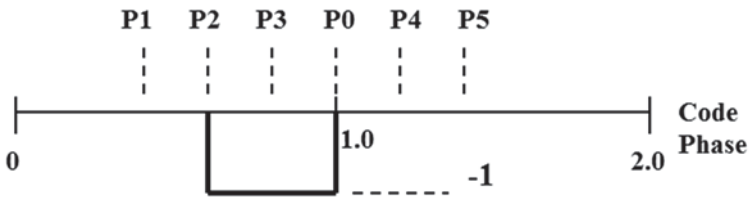

Fig. 4-18: MB Non-Transition Correlation Kernel Mapped to Total Code Phase.

Note that the transition and non-transition kernels have the potential to be used every new code bit; therefore, it is natural to use the fractional code NCO phase. The upper K bits of the fractional NCO code represent the phase from 0 (inclusive) to 1 (exclusive); where 1 represents the next code bit. Fig. 4-19 maps the Mass Balance transition kernel from the total code phase (Fig. 4-17) into the fractional NCO code phase between 0 (inclusive) and 1.0 (exclusive). Note that P0 is shown twice and represent the same fractional code NCO phase of zero. Also, note that the kernel region of support does not include phase 0.5, so P5 < 0.5 and P1 > 0.5.

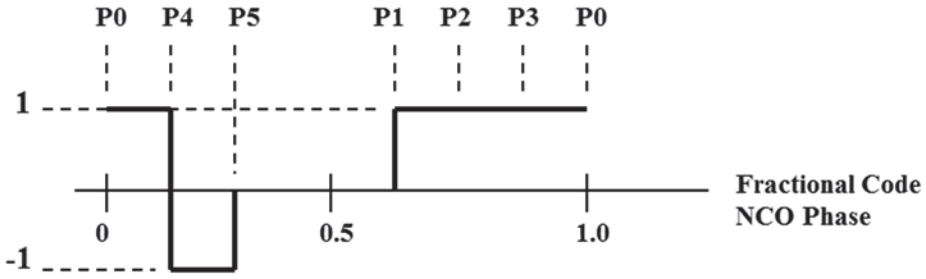

Fig. 4-19: MB Transition Correlation Kernel Mapped to Fractional Code Phase.

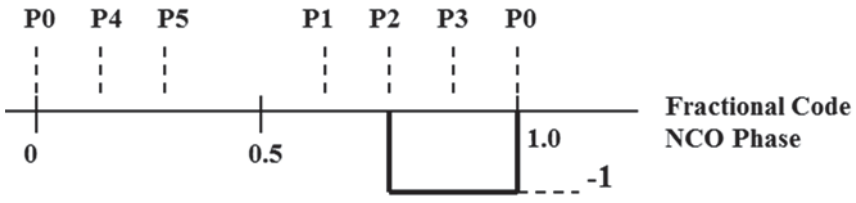

Fig. 4-20: MB Non-Transition Correlation Kernel Mapped to Fractional Code Phase.

Note that by defining the five code phase boundaries, P1 through P5, many different kernels can be implemented within the same algorithm including the Mass-Balance transition and non-transition kernels as well as the EML kernel with an arbitrary δ. A flow chart of the logic encoded algorithm is shown in Fig. 4-21. Note that the boundaries P0 through P5 are registers and can support arbitrary values corresponding to the required NCO phase. The initialization step occurs after every accumulation period and consists of reloading the phase boundaries, P1 through P5, the discriminator type (*DiscrimType*) and clearing the *XTionSum* and *NoXTionSum* correlation sums.

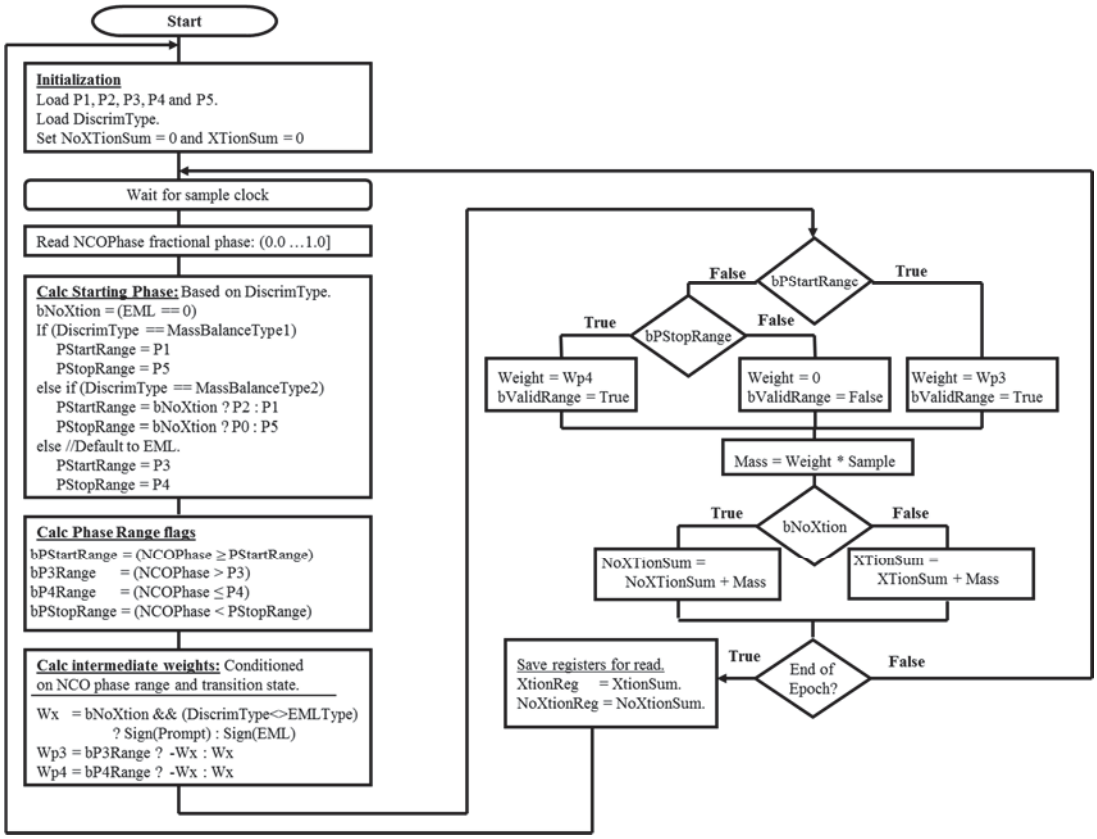

Fig. 4-21: Algorithm to Implement Kernel In Digital Logic.

Chapter 5

PRECORRLEATION FILTER EFFECTS

The precorrelation filter, *H(w)*, is shown in Fig. 2-7. The effects of a non-ideal filter on the statistic $R_{X\hat{X}}(\varepsilon)$ has been covered by [13]. However, this analysis has yet to be applied to the concept of a code tracking kernel, *w(t)*, and the impact on the statistic $R_{ZW}(\varepsilon)$. This chapter follows the analysis developed by [13] for $R_{X\hat{X}}(\varepsilon)$ and applies the analysis to $R_{ZW}(\varepsilon)$ to gain insight into the design constraints for the tracking kernel weights and its region of support. Specifically, we investigate the effects of bandwidth, passband ripple and phase distortion.

5.1. *Precorrelation Filter Distortion Model*

When *h(t)* is real, the magnitude response is even and the phase response is odd. Therefore, passband ripple can be represented via the even function,

$$|H(j\omega)| = 1 + a\cos(\alpha\omega) = 1 + \frac{a}{2}e^{-j\alpha\omega} + \frac{a}{2}e^{+j\alpha\omega}, \tag{5.1}$$

where a is the ripple amplitude and α the ripple frequency. Phase distortion can be represented via the odd function,

$$Arg[H(\omega)] = e^{-j\theta\sin\gamma\omega}. \tag{5.2}$$

Where θ is the phase ripple amplitude and γ is the phase ripple frequency. The combined filter response is given by (5.3) and (5.4) to yield,

$$H(j\omega) = |H(j\omega)| \cdot Arg[H(\omega)] = \left(1 + \frac{a}{2}e^{-j\alpha\omega} + \frac{a}{2}e^{+j\alpha\omega}\right)e^{-j\theta\sin\gamma\omega}. \tag{5.3}$$

For a band limited filter response, (5.3) holds for $-\frac{B}{2} \leq \omega \leq \frac{B}{2}$ and is 0 elsewhere. For the Mitel DW9255 IF SAW filter, a = -1.6dB, α = 0.8Tc (GPS L1CA), θ = 150ns, and γ = 0.8Tc and B=2.046MHz.

5.2. *Effect on* $R_{Z\hat{X}}(\varepsilon)$

The effects of the filter distortions defined above on $R_{Z\hat{X}}(\varepsilon)$ are considered. The statistic $R_{Z\hat{X}}(\varepsilon)$ is defined in (2.17) and repeated below for convenience.

$$R_{Z\hat{X}}(\varepsilon) = R_{X\hat{X}}(\varepsilon) * h(t), \tag{5.4}$$

and the corresponding time domain signal utilizing the Fourier Transform is,

$$R_{Z\hat{X}}(\varepsilon) = \frac{1}{2\pi}\int_{-\infty}^{\infty} S(\omega)\, H(\omega) e^{-j\omega\epsilon} d\omega. \tag{5.5}$$

Where $S(w)$ is the spreading code power spectrum as described in Chapter 2 and *H(ω)* is the frequency domain representation of the real filter *h(t)*. Four characteristics of *H(ω)* are considered:

1. Passband ripple distortion;
2. Phase ripple distortion;
3. Combined passband and phase distortion; and
4. Band limited.

Case 1: Passband Ripple Distortion

Substitute the *H(jω)* passband ripple model (5.1) into (5.5) to obtain,

$$R_{Z\hat{X}}(\varepsilon) = \frac{1}{2\pi}\int_{-\infty}^{\infty} S(\omega)\left\{1 + \frac{a}{2}e^{-j\alpha\omega} + \frac{a}{2}e^{+j\alpha\omega}\right\} e^{-j\omega\epsilon} d\omega. \tag{5.6}$$

Expanding terms yields,

$$\begin{aligned} R_{Z\hat{X}}(\varepsilon) = {} & \frac{1}{2\pi}\int_{-\infty}^{\infty} S(\omega) e^{-j\omega\epsilon} d\omega + \frac{1}{2\pi}\int_{-\infty}^{\infty} S(\omega)\frac{a}{2} e^{-j\alpha\omega} e^{-j\omega\epsilon} d\omega \\ & + \frac{1}{2\pi}\int_{-\infty}^{\infty} S(\omega)\frac{a}{2} e^{+j\alpha\omega} e^{-j\omega\epsilon} d\omega, \end{aligned} \tag{5.7}$$

and then taking the inverse Fourier transform,

$$R_{Z\hat{X}}(\varepsilon) = R_{X\hat{X}}(\varepsilon) + \frac{a}{2}R_{X\hat{X}}(\varepsilon - \alpha) + \frac{a}{2}R_{X\hat{X}}(\varepsilon + \alpha). \quad (5.8)$$

From (5.8) we see that passband ripple manifests as scaled and shifted replicas of the autocorrelation with the scale equal to the ripple amplitude, a, and the shifts equal to $\pm\alpha$. Note that (5.8) is even symmetric due to even symmetry $R_{Z\hat{X}}(\varepsilon)$.

Case 2: Phase Ripple Distortion

Substitute the *H(jω)* phase distortion model (5.2) into (5.5) to obtain,

$$R_{Z\hat{X}}(\varepsilon) = \frac{1}{2\pi}\int_{-\infty}^{\infty} S(\omega)\left\{e^{-j\theta sin\gamma\omega}\right\}e^{-j\omega\epsilon}d\omega. \quad (5.9)$$

Use the following Bessel function identities,

$$e^{-j\theta sin\gamma\omega} = \sum_{k=-\infty}^{\infty} J_k(\theta)e^{-jk\gamma\omega} \text{ and } J_{-k}(\theta) = (-1)^k J_k(\theta), \quad (5.10)$$

to obtain,

$$R_{Z\hat{X}}(\varepsilon) = \frac{1}{2\pi}\int_{-\infty}^{\infty} S(\omega)\left\{\sum_{k=-\infty}^{\infty} J_k(\theta)e^{-jk\gamma\omega}\right\}e^{-j\omega\epsilon}d\omega, \quad (5.11)$$

and moving the integral inside the summation,

$$R_{Z\hat{X}}(\varepsilon) = \sum_{k=0}^{\infty}(-1)^k J_k(\theta)\frac{1}{2\pi}\int_{-\infty}^{\infty} S(\omega)e^{-jk\gamma\omega}e^{-j\omega\epsilon}d\omega, \quad (5.12)$$

we obtain the final result,

$$R_{Z\hat{X}}(\varepsilon) = J_0(\theta)R_{X\hat{X}}(\varepsilon) + J_k(\theta)R_{X\hat{X}}(\varepsilon + k\gamma) - J_k(\theta)R_{X\hat{X}}(\varepsilon - k\gamma), \quad (5.13)$$

for k=1 to ∞. From (5.13) we see that phase distortion manifests as scaled and shifted replicas of the autocorrelation. However, the sign change within the scaling produces an odd symmetric $R_{Z\hat{X}}(\varepsilon)$. This odd symmetry will manifest as a code phase tracking bias as shown in below. The remainder of this chapter is new and extends [13].

Case 3: Passband Ripple and Phase Distortion

Substitute (5.3) into (5.5) and simplify using the steps outlined above to obtain the result found in Fig. 5-1. Note that the distortion cross-terms are also odd symmetric.

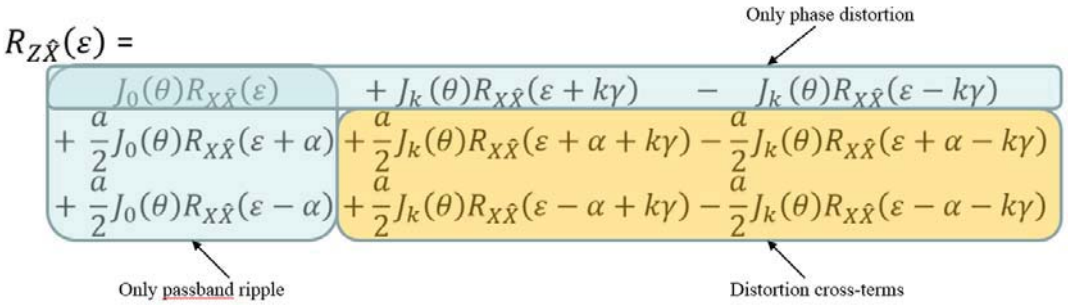

Fig. 5-1: *H(jω)* Amplitude and Phase Distortion (*k*=1 to ∞).

BOC(m,n) signals exaggerate the distortion over Rect(n) due to the multi-peak and sign changes within their autocorrelation function.

Case 4: *H(jω)* is Band-limited

When the incoming signal is band limited, either at the transmitter or the receiver, some signal power is removed at the higher frequencies. This frequency truncation manifests itself as ringing in the time domain. This time-domain distortion directly increases the phase estimation variance. Consequently, it is desirable to match the receiver bandwidth to the transmit bandwidth to the extent practical for the system. The effect of signal bandwidth on phase estimation and the size of δ is investigated below.

Fig. 5-2 plots the autocorrelation peak degradation as a function of an ideal band-limited filter with varying bandwidths for Rect(1), SinBOC(1,1) and CosBOC(1,1). A receiver must have a single sided input bandwidth of at least $4.5f_0$, $12f_0$, and $18f_0$ respectively, to limited the peak autocorrelation degradation by less than 0.1dB.

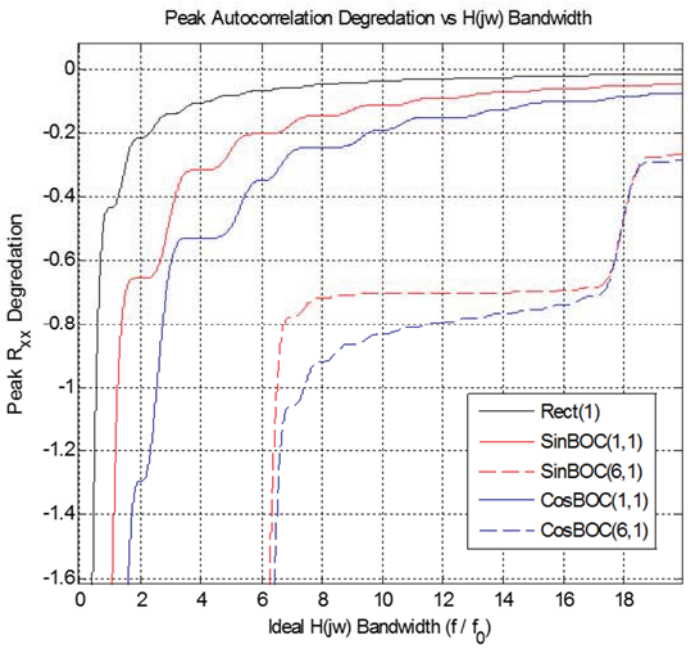

Fig. 5-2: Peak Autocorrelation Degradation for Ideal Band-limited *H(jω)*.

A multipath free code pulse is plotted in Fig. 5-3 for a variety of *H(jω)* bandwidths modeled as a 6th order Butterworth with specified 3dB cutoff. The time domain ringing creates distortion similar to multipath with a positive path gain and the slope of the chip edge boundary decreases with decreasing bandwidth.

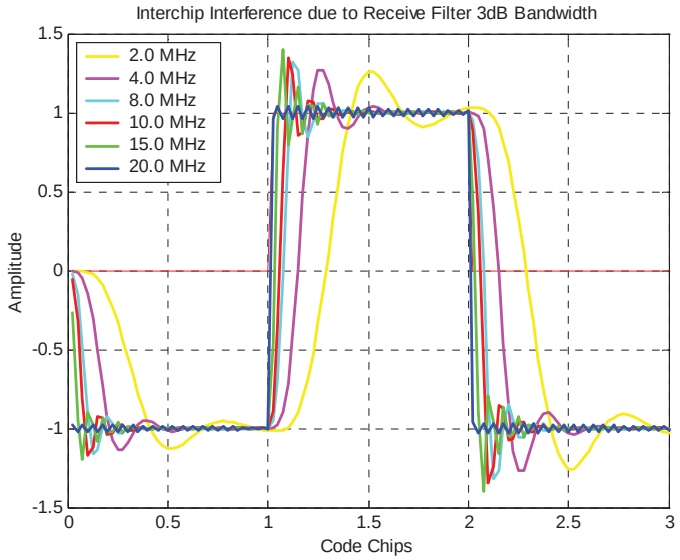

Fig. 5-3: Rect(1) Inter-Chip Interference Due to Ideal Band-limited *H(jω)*.

The cross correlation between the band limited received signal and a matched- chip for Rect(1) and SinBOC(1,1) are plotted in Fig. 5-4 and Fig. 5-5 respectively. The plot compensates for the varying analog group delays. Within the GNSS receiver, these fixed delays are projected onto the receiver clock estimate within (1.1) and do not cause a phase estimation bias. Note that the cross correlations for the lower 3dB bandwidths are wider and begin to flatten at zero lag compared to the 20.0 MHz single sided filter bandwidth. This produces a rapid rate of change in the first derivative of $R_{Z\hat{X}}(\varepsilon)$ which reduces the ML code phase estimator. This is undesirable and motivates maximizing the receiver bandwidth to capture more signal power. The disadvantage of a wider input bandwidth is an increase in noise power, susceptibility to narrow band interference, an increase in the required sample rate, and finally an increase in the required receiver power.

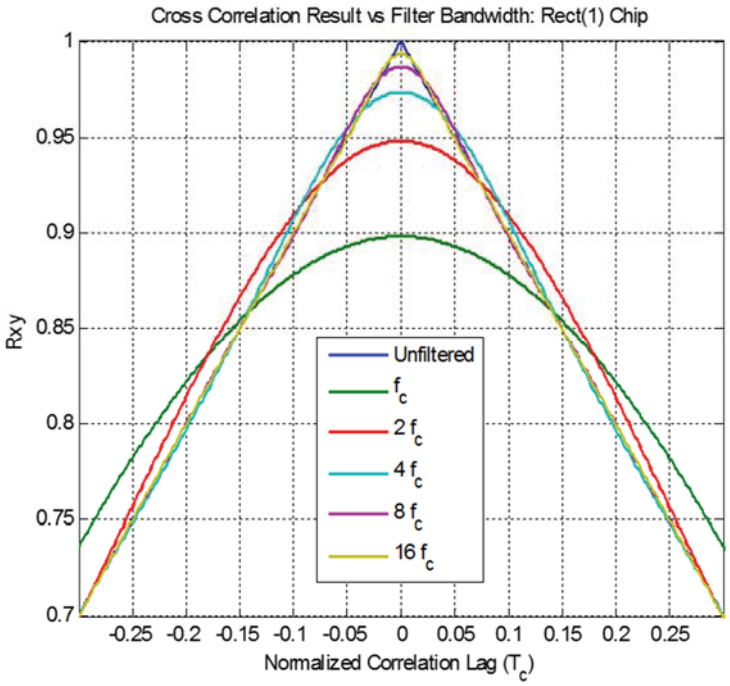

Fig. 5-4: Band Limited Received Signal Cross Correlated with Rect(1) Chip.

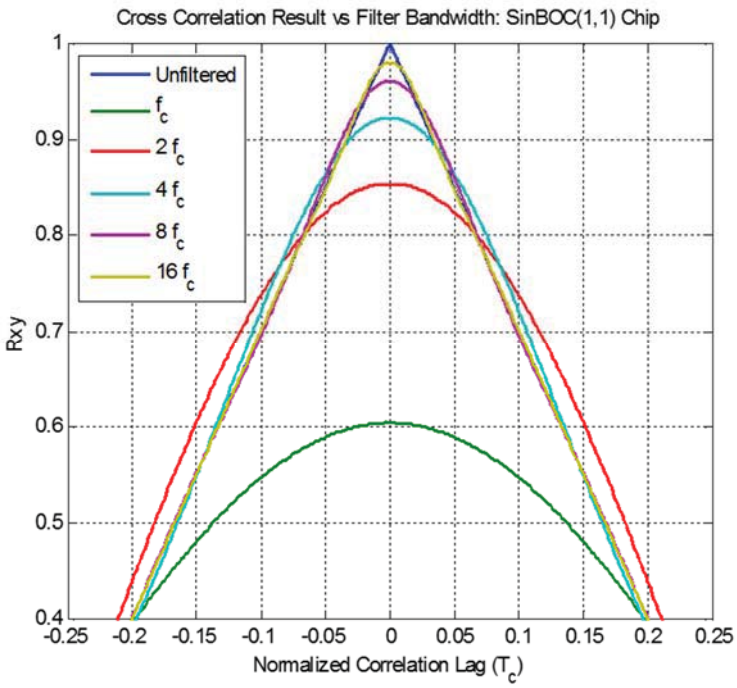

Fig. 5-5: Band Limited Received Signal Cross Correlated with SinBOC(1,1) Chip.

Plots of a severely band limited SinBOC(1,1) signal that includes multipath are presented in Fig. 5-6 and Fig. 5-7. Note that all possible combinations of two-chip transitions are represented. Note that when a signal has positive multipath gain, as in Fig. 5-6, the direct filtered path leads the composite signal filtered path. When the signal has negative multipath gain (180 degree phase shift) the direct filtered path lags the composite signal filtered path as shown in Fig. 5-7. Therefore, although multi-path signals are inherently delayed, they can produce either a positive or a negative phase bias.

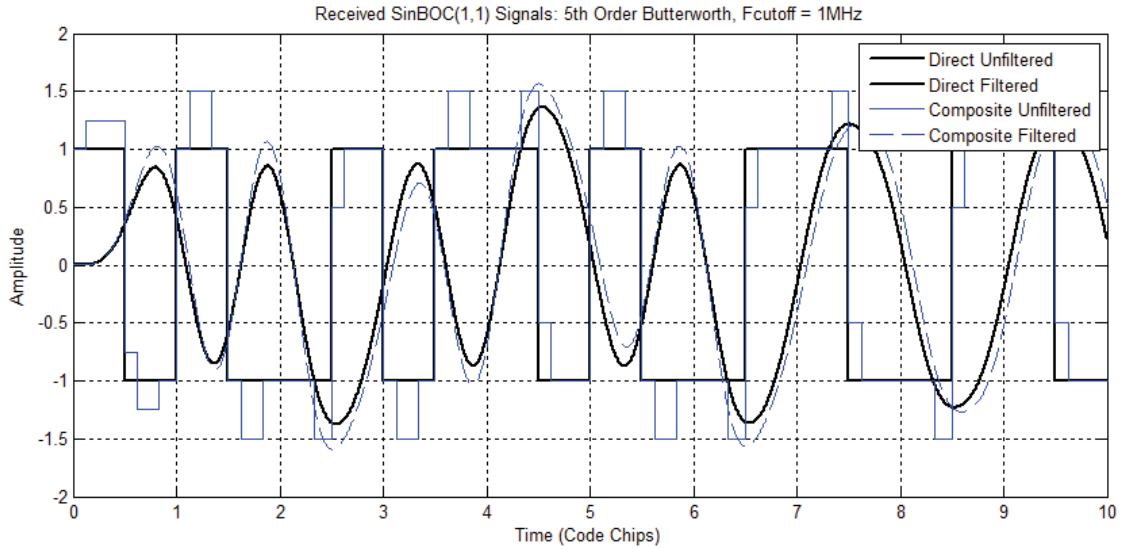

Fig. 5-6: SinBOC(1,1) with α=[0.25, 0.25] and τ=[Tc/8, Tc/1.2].

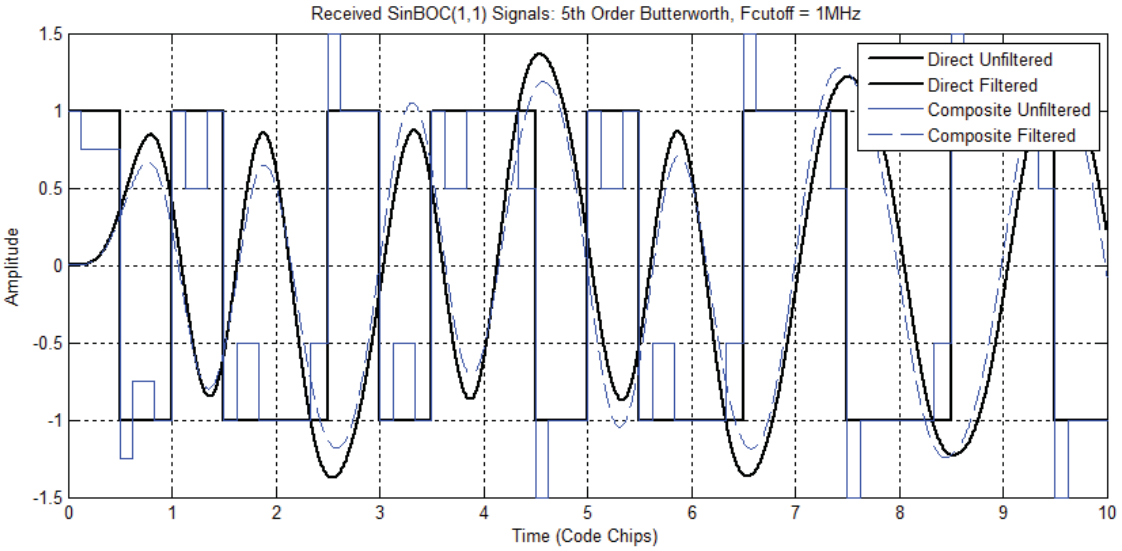

Fig. 5-7: SinBOC(1,1) with α=[-0.25, -0.25] and τ =[Tc/8, Tc/1.2].

For narrow channels, note that the zero crossing for the direct path and composite path align when the two prior chips had equal sign. This can be exploited by blanking the correlation kernel when the two previous chip changed sign. The disadvantage is that this will reduce the overall correlation gain. What is happening is that the composite path has higher frequency content that experiences different phase delay through the low pass filter.

5.3. *Effect on $R_{ZW}(\varepsilon)$ and Kernel Design Considerations*

This section applies the analysis from the previous section to the correlation kernel statistic, $R_{XW}(\varepsilon)$, to derive kernel design constraints. The statistic $R_{ZW}(\varepsilon)$ is defined in (2.21) and repeated below for convenience,

$$R_{ZW}(\varepsilon) = R_{XW}(\varepsilon) * h(t). \tag{5.13}$$

We can represent the correlation as a convolution,

$$R_{ZW}(\varepsilon) = x^*(t) * w^*(t-\varepsilon) * h(t), \tag{5.14}$$

and represent the time domain signal utilizing the Fourier Transform as,

$$R_{ZW}(\varepsilon) = \frac{1}{2\pi}\int_{-\infty}^{\infty} S_x(\omega)\Psi^*(\omega)\, H(\omega)e^{-j\omega\epsilon}d\omega. \tag{5.15}$$

Where $S_x(\omega)$ is the spreading code ensemble average power spectrum, $\Psi(\omega)$ is the power spectrum of the correlation kernel, *w(t)*, and *H(ω)* is the frequency domain representation of the real filter, *h(t)*. We consider the same four characteristics of *H(ω)* as the previous section.

Case 1: Passband Ripple Distortion

Recall that $R_{ZW}(\varepsilon)$ is a linear combination of $R_{X\hat{X}}(\varepsilon)$ for the ML discriminant as well as any discriminant based upon a linear combination of EML kernels. Therefore, taking results from (5.8), we obtain,

$$\begin{aligned}\check{R}_{ZW}(\varepsilon) = {} & R_{X\hat{X}}(\varepsilon-\delta) - R_{X\hat{X}}(\varepsilon+\delta) + \frac{a}{2}R_{X\hat{X}}(\varepsilon-\delta-\alpha) \\ & - \frac{a}{2}R_{X\hat{X}}(\varepsilon+\delta-\alpha) + \frac{a}{2}R_{X\hat{X}}(\varepsilon-\delta+\alpha) \\ & - \frac{a}{2}R_{X\hat{X}}(\varepsilon+\delta+\alpha).\end{aligned} \tag{5.16}$$

This is an even function since the EML kernel is even; therefore, a phase bias does not occur. However, the discriminant is still distorted as shown in (5.16). When a non-symmetric kernel is used,

$$\check{R}_{ZW}(\varepsilon) = R_{XW}(\varepsilon) + \frac{a}{2}R_{XW}(\varepsilon - \alpha) + \frac{a}{2}R_{XW}(\varepsilon + \alpha), \tag{5.17}$$

which results in a phase bias, $\check{R}_{ZW}(0)$. The bias can be reduced by limiting the kernel's region of support. For MB, the early replica cannot be mitigated and will remain as a bias. However, the late replica, which looks like multipath, can be removed by setting the region of support so $\delta < \alpha$.

Case 2: Phase Ripple Distortion

$R_{ZW}(\varepsilon)$ is a linear combination of $R_{X\hat{X}}(\varepsilon)$ for the ML discriminant. Making the appropriate EML substitutions into (5.13) yields,

$$\begin{aligned}\check{R}_{ZW}(\varepsilon) = {} & J_0(\theta)R_{X\hat{X}}(\varepsilon - \delta) - J_0(\theta)R_{X\hat{X}}(\varepsilon + \delta) \\ & + J_k(\theta)R_{X\hat{X}}(\varepsilon - \delta + k\gamma) - J_k(\theta)R_{X\hat{X}}(\varepsilon + \delta + k\gamma) \\ & - J_k(\theta)R_{X\hat{X}}(\varepsilon - \delta - k\gamma) + J_k(\theta)R_{X\hat{X}}(\varepsilon + \delta - k\gamma),\end{aligned} \tag{5.18}$$

Since $R_{X\hat{X}}$ is even, then $\check{R}_{ZW}(\varepsilon)$ is an odd function. $\check{R}_{ZW}(0)$ of (5.18) is the resulting phase bias. When a non-symmetric kernel is used,

$$\check{R}_{ZW}(\varepsilon) = J_0(\theta)R_{XW}(\varepsilon) + J_k(\theta)R_{XW}(\varepsilon + k\gamma) - J_k(\theta)R_{XW}(\varepsilon - k\gamma). \tag{5.13}$$

which results in a phase bias, $\check{R}_{ZW}(0)$. The bias can be reduced by limiting the kernel's region of support. For MB, the early replica cannot be mitigated and will remain as a bias. However, the late replica, which looks like multipath, can be removed by setting the region of support so $\delta < k\gamma$.

To mitigate phase bias for amplitude or phase ripple, a kernel can be developed such that the discriminant's ROS is less than $\min\{\alpha, k\gamma\}$.

Case 3: Passband Ripple and Phase Distortion

The cross-terms generate a code phase bias in-addition to the code phase bias due to the amplitude and phase ripple discussed above. To mitigate phase bias for amplitude and phase ripple, a kernel would need to be developed such that the discriminant's ROS is less than $\min\{\alpha, k\gamma, \pm\alpha - k\gamma, \pm\alpha + k\gamma\}$.

Case 4: *H(jω)* is Band-limited

A band-limited *H(jω)* imposes constraints on the *w(t)* kernel. For EML, the derivative of $R_{X\hat{X}}(\varepsilon)$ changes with lag with the rate of change determined by the filter 3dB bandwidth and filter order. As $dR_{X\hat{X}}(0)/d\varepsilon$ approaches zero as shown in Fig. 5-4 and Fig. 5.5, the required EML δ must increase since the discriminant can become zero even when the phase error does not equal zero. To maintain discriminant gain, δ should be selected such that $\frac{d\check{R}_{X\hat{X}}(\delta)}{d\varepsilon} = \frac{dR_{X\hat{X}}(\delta)}{d\varepsilon}$, where $\check{R}_{X\hat{X}}$ is the autocorrelation distorted by the bandlimited filter, h(t). This also applies to δ_x for the MB kernel.

The EML $\delta = 1/2$, EML $\delta = 1/16$, and MB(8/256,16/256) discriminators for various multipath scenarios with *H(jω) B=10f*$_c$ are shown in Fig. 5-8, Fig. 5-9 and Fig. 5-10 respectively. For a Rect(1) signal, the *B=10f*$_c$ filter passes greater than 97% of the available signal power which results in an undistorted linear region of the discriminators. Fig. 5-11 provides a zoomed in view of Fig. 5-10 around the negative lag region and demonstrates the increased sensitivity of the MB kernel to bandwidth over the EML kernel.

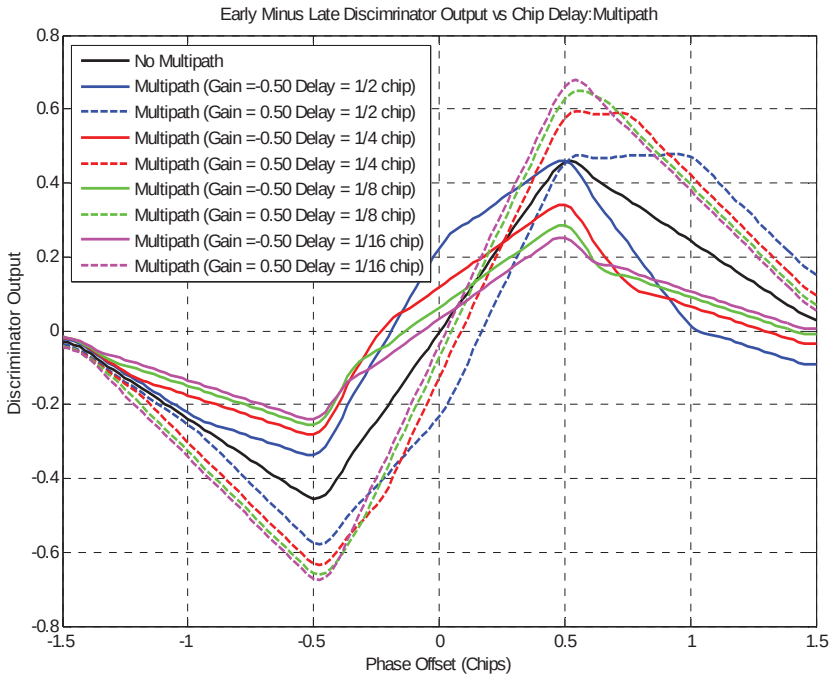

Fig. 5-8 Rect(1) EML δ=1/2 Discriminant with H(ω) B=10f_c.

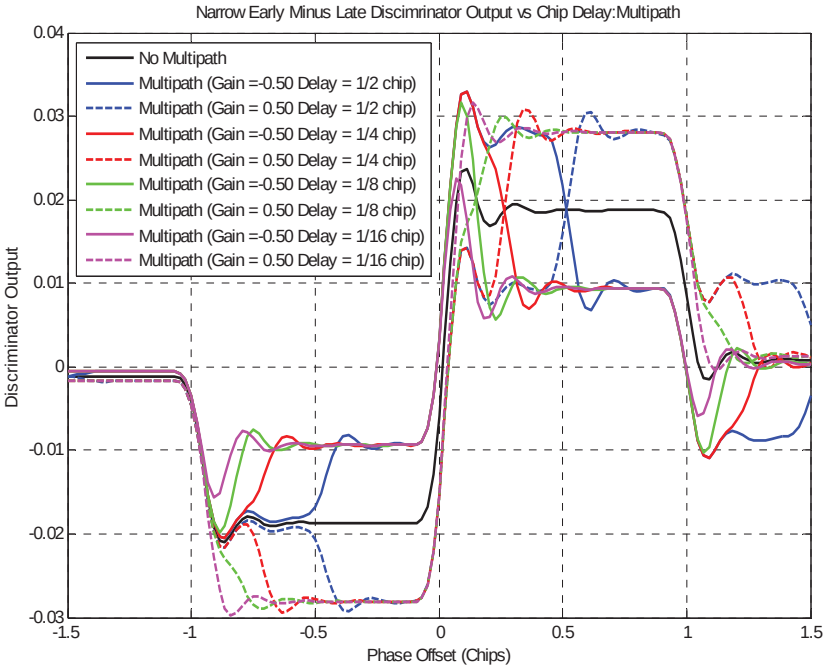

Fig. 5-9: Rect(1) EML δ=1/16 Discriminant with H(ω) B=10f_c.

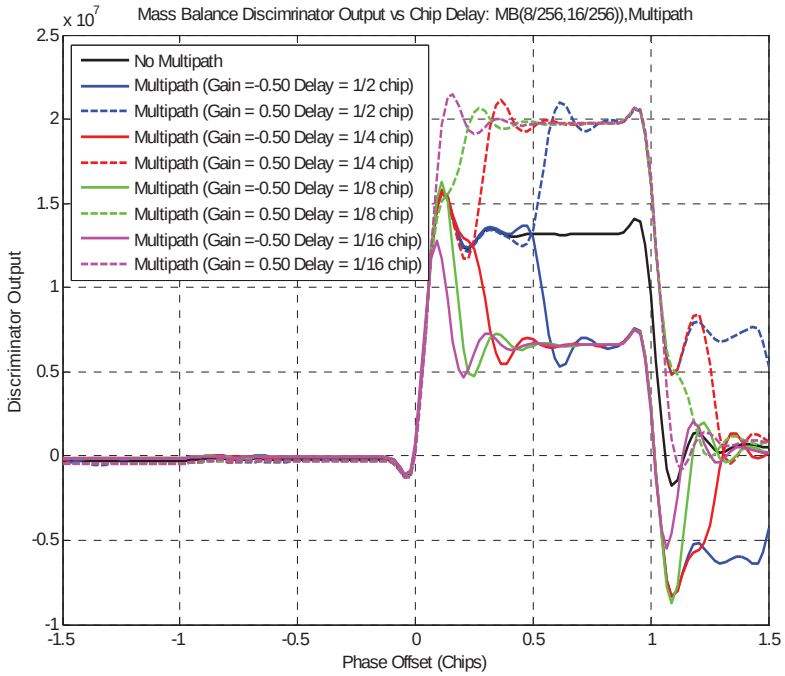

Fig. 5-10: Rect(1) MB(8/256,16/256) Discriminant with H(ω) B=10f_c.

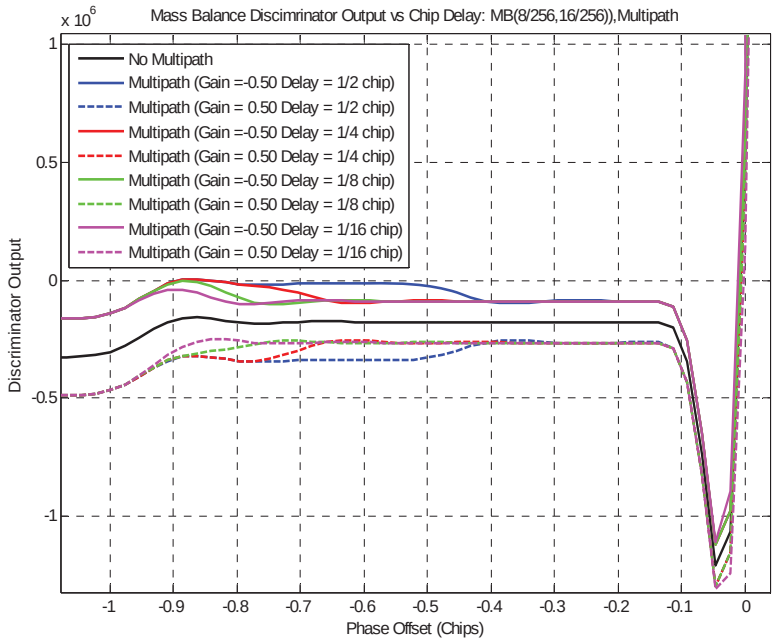

Fig. 5-11: Negative Lag Portion of MB(8/256,16/256) Discriminant with H(ω) B=10f_c.

Fig. 5-10 through Fig. 5-12 show the discriminators under the same multipath scenarios but with $H(j\omega)$ $B=4f_c$. The EML based discriminator gains have decreased and the multipath bias has increased. Note that the EML(1/16) kernel no longer mitigates multipath with a delay spread smaller than $T_c/8$. Further note the MB kernel is unable to function as a code phase discriminator with this narrow bandwidth.

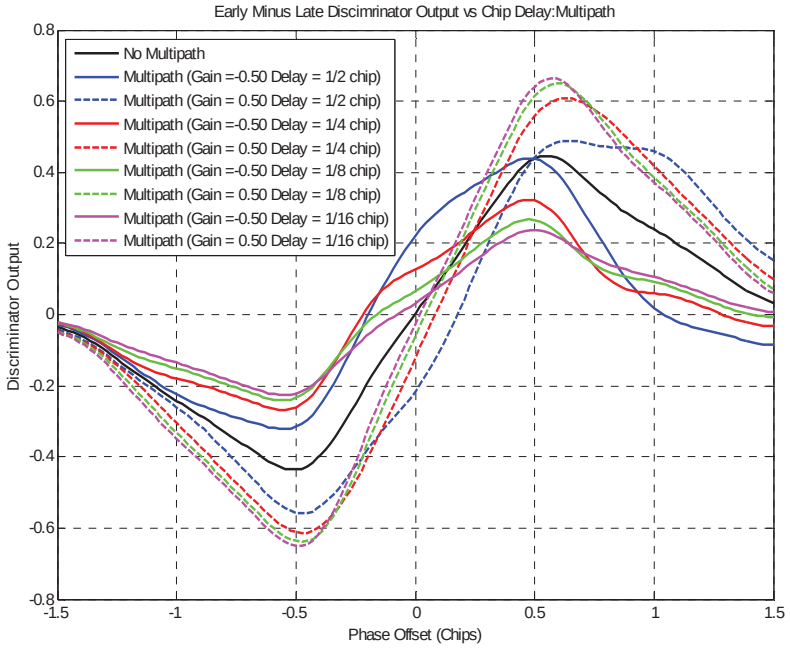

Fig. 5-12: Rect(1) EML $\delta=1/2$ Discriminant with $H(\omega)$ $B=4f_c$.

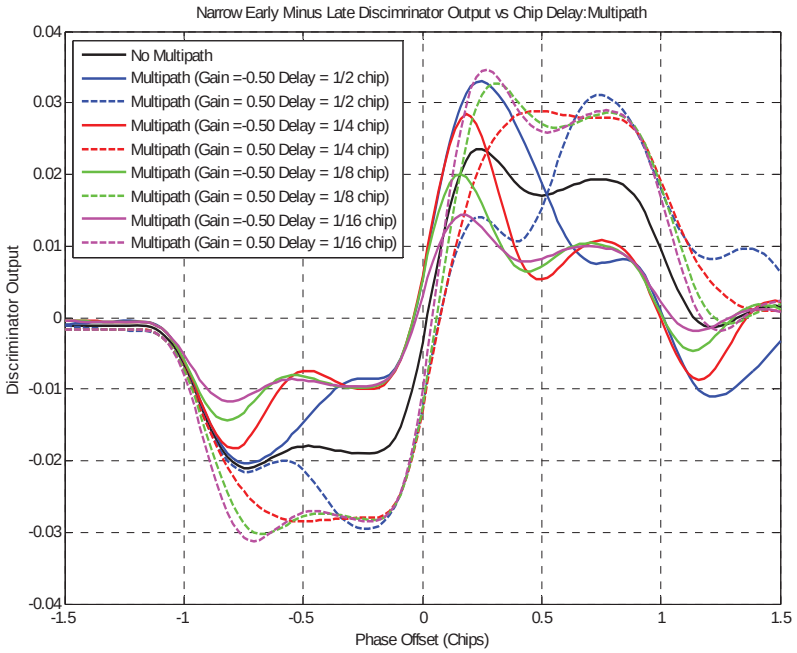

Fig. 5-13: Rect(1) EML δ=1/16 Discriminant with H(ω) B=4f_c.

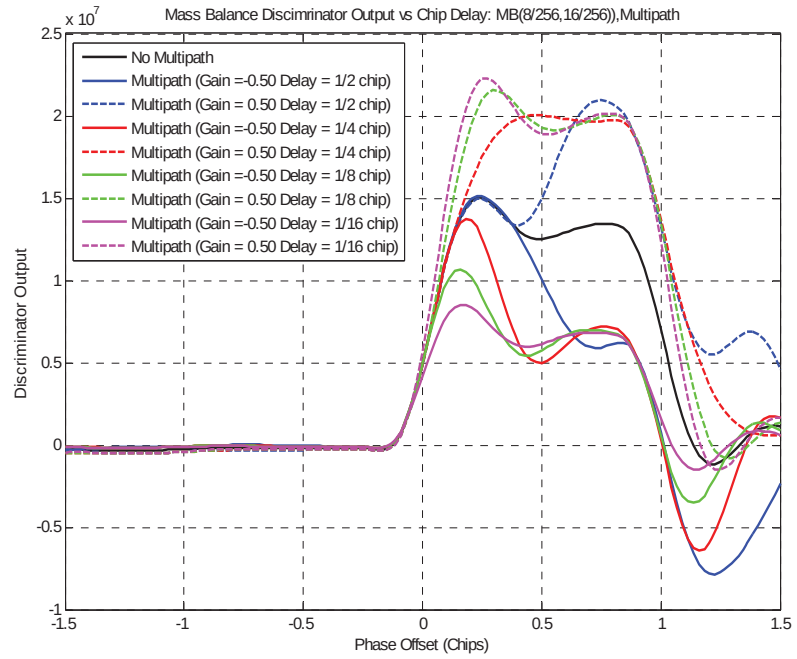

Fig. 5-14: Rect(1) MB(8/256,16/256) Discriminant with H(ω) B=4f_c.

Fig. 5-15 compares the normalized power spectrum of the EML(1/16) kernel with a series of MB(1/16, *x*/16) for *x*=0,2, and 4. The MB asymmetric kernel has a power spectrum similar to a non-return-to-zero (NRZ) signal indicated by the spectral null at zero frequency. The MB kernel mean increases with *x*, so spectral null becomes less pronounced as *x* increases. This is also illustrated by the corresponding power profile plotted in Fig. 5-17. Note that the spectral envelope is set by the first MB parameter and the ripple amplitude and frequency are set by the second MB parameter. MB parameters of MB(1/*x*,2/*x*) provide a good trade-off between power profile and R_{ZW} accumulation gain. Fig 5-16 plots the power spectrum of various MB kernels.

Recall that the kernel based multipath mitigation performance is set by the leading pulse width and its ROS. Recall that the MB multipath performance of MB(1/{2x},•) is equivalent to EML(1/x); which corresponds to 1:2 pulse width ratio. Therefore, as expected, the MB kernel requires a greater bandwidth than the EML kernel. The MB kernel provides independent control of the discriminator gain via the second MB parameter. The discriminator tracking range for delayed signals is set by the leading pulse width, so the EML(1/x) kernel has better tracking range than the equivalent performing MB kernel.

Fig. 5-18 plots the power profile for various MB parameters. The upper envelope of the power profile is set by the first parameter which sets the negative pulse width of the MB kernel. The lower envelope of the MB power profile occurs when the second MB parameter is zero.

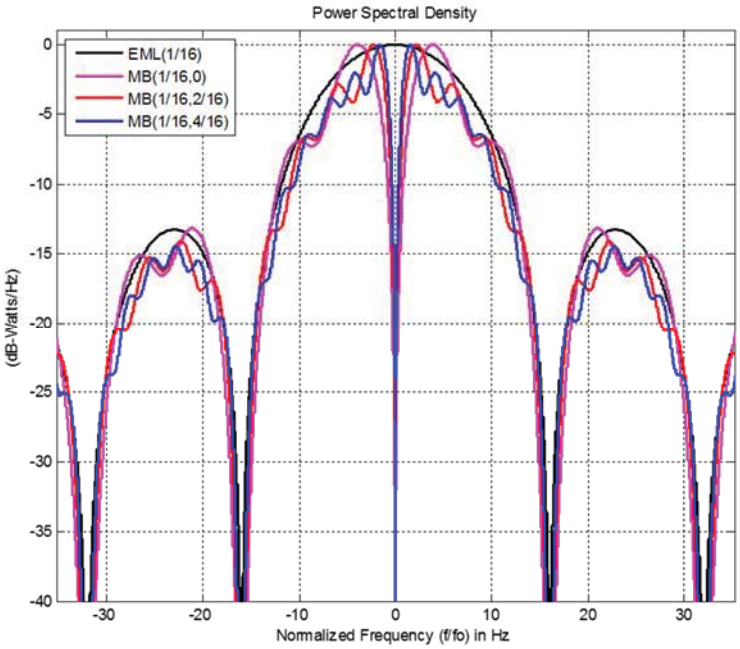

Fig. 5-15: Comparison of EML(1/16) and MB(1/16,•) Kernel Power Spectrums.

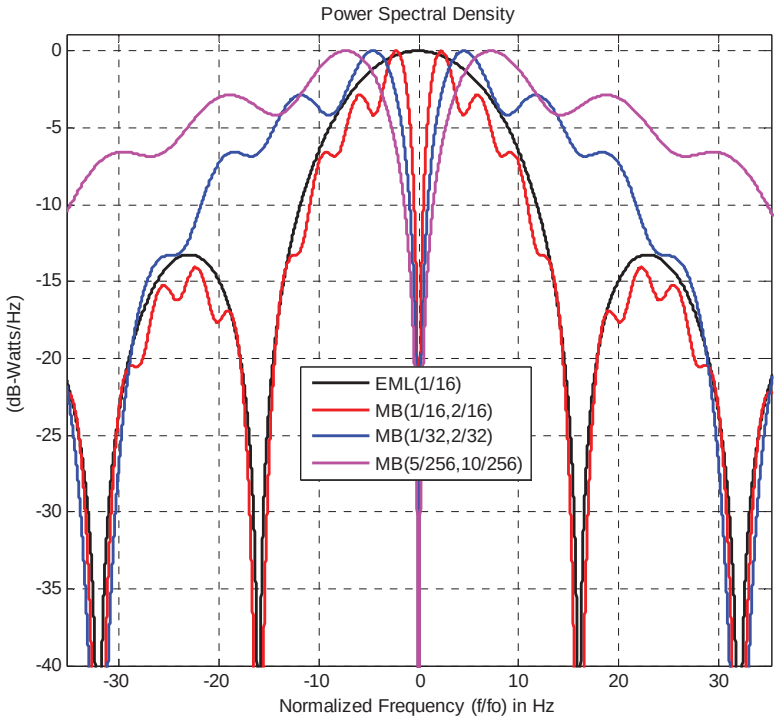

Fig. 5-16: Comparison of EML(1/16) and Various MB Kernel Power Spectrums.

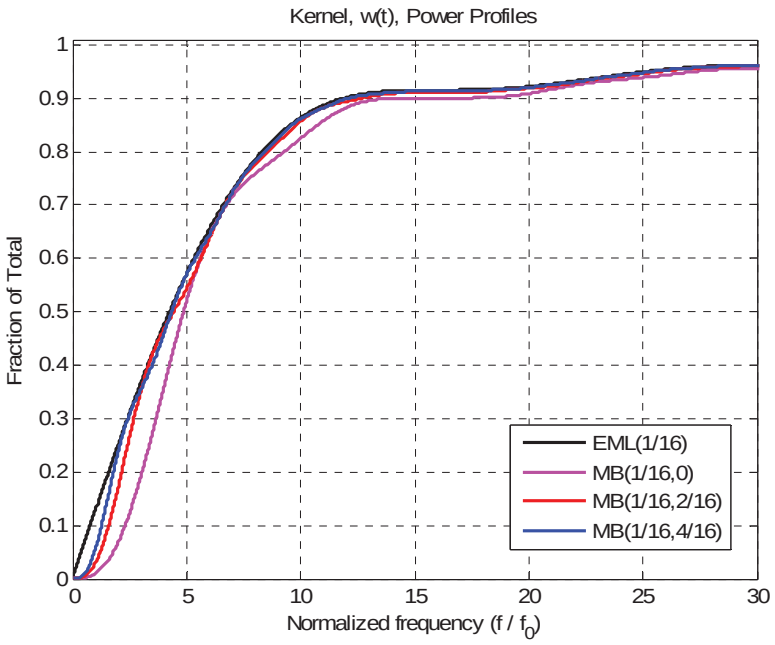

Fig. 5-17: Comparison of EML(1/16) and MB(1/16,•) Kernel Power Profiles.

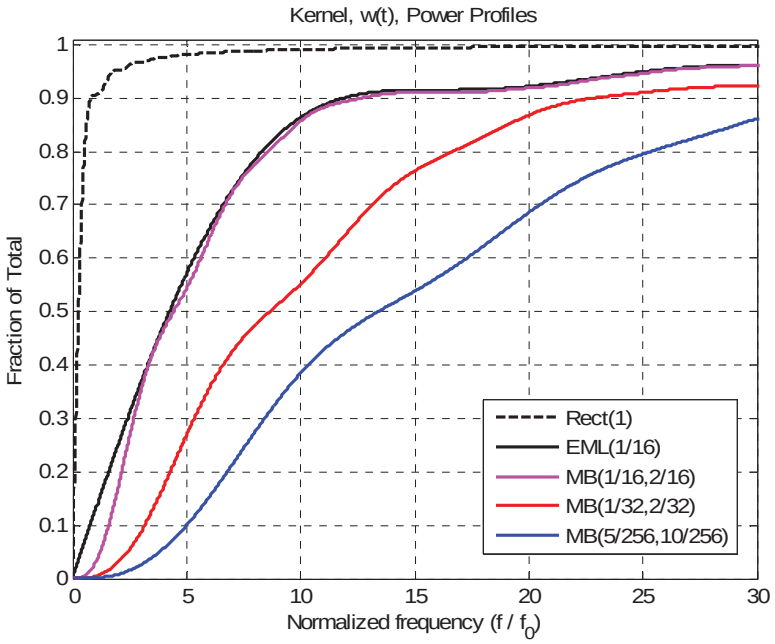

Fig. 5-18 Comparison of EML(1/16) and Various MB Kernel Power Profiles.

Chapter 6

CONCLUSION AND FUTURE WORK

6.1. *Concluding Remarks*

GNSS systems estimate position based upon time-of-arrival estimates obtained by tracking a satellite's code and carrier phase. Multipath is a dominant error source since it corrupts the signal phase estimates with a time-varying bias. This dissertation investigated multipath mitigating techniques utilizing correlation kernels.

A novel class of asymmetric correlation kernels, referred to as a Mass Balance (MB), was developed which resulted in US and international patents. It was also discovered that some GPS L1CA spreading codes have unequal transition and non-transition probabilities which may cause a code phase estimation bias for non-symmetric kernels. The proposed algorithm compensated for the imbalance and does not suffer a tracking bias due to unbalanced spreading codes. Efficient digital implementation schemes were also presented. One scheme is utilized within an application specific integrated circuit with over 300,000 units deployed.

The MB performance was compared against existing maximum likelihood EML techniques and assessed on signal tracking bias, variance and robustness. It was shown that the MB kernel has better multipath performance given the same ROS but requires a greater *minimum* precorrelation bandwidth than the equivalent EML kernel. Specifically, the precorrelation filter bandwidth should capture greater than 97% of the available signal power. A disadvantage of the MB kernel is its lower discriminator gain and shorter tracking region for $\varepsilon<0$. However the MB tracking region can be extended. The MB kernel

can be programmed to have a larger discriminator gain than the equivalent EML for $\varepsilon>0$. The MB and EML noise variances are equivalent.

The effects of precorrelation filter distortion on the correlation kernel discriminator was presented. A novel scheme was presented to mitigate some filter distortion by treating it like multipath and then selecting the correlation kernel parameters appropriately. It was shown that some distortion cannot be mitigated and minimizing phase ripple and amplitude ripple within the passband is recommended.

6.2. Future Work

Additional research into the benefits of a non-rational sample rate to code rate ratio is required to assess the potential phase estimation bias under very low Doppler conditions. Such a slowly varying bias may result when the signal is sampled at the same phase points of the spreading code. Intuitively, a better result will be achieved when the sample rate is not commensurate with the code rate and the position within the spreading code chip changes more rapidly than the multipath changes. Moreover, additional research into the relationship between the sample rate and the kernel ROS and weights is required.

Additional research is required to improve the code tracking range of multipath mitigating kernels. Future GNSS signals will utilize Alt-BOC(m,n) signals that are not constant modulus; therefore, additional research into maximum likelihood receivers that process this signal class would be beneficial.

Research into the use of space-time-frequency (STF) techniques within the field of GNSS multipath mitigation, rapid signal acquisition and robust signal tracking is required. A jointly adaptive STF scheme that exploits multiple frequencies, multiple antennas and multiple signals to develop tracking kernels is sketched in Fig. 6-1 and 6-2. Note that the

results in Chapter 5 constrain the minimum distance between kernel particles (weighted regions) as a function of the pre-correlation filter bandwidth. The research would concentrate on the error equation with applied constraints as well as particle spacing. Commercial multi-antenna systems used within the marine market to obtain attitude information may be well suited to exploit this technology. Moreover, stationary GNSS reference (base) stations would be able to characterize the multipath environment and mitigate the multipath effects resulting in improved differential phase data.

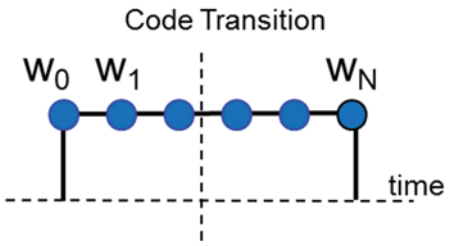

Fig. 6-1: Adaptive Kernel Weights (Shown with Equal Spacing).

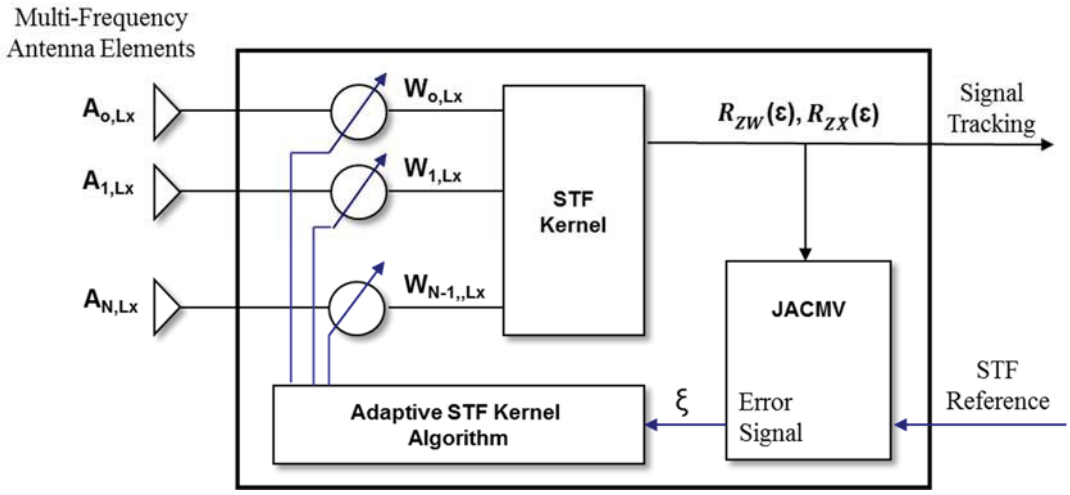

Fig. 6-2: Adaptive Space-Time-Frequency (STF) Kernel.

REFERENCES

[1] C. Francescani, "Commuter ferry crash in New York injures 57, one critical," 9 January 2013. [Online]. Available: http://www.reuters.com/article/2013/01/09/us-usa-newyork-ferry-idUSBRE9080O920130109.

[2] "NAVSTAR Global Positioning System Interface Specification IS-GPS-200 Revision F," 21 September 2011.

[3] E. D. Kaplan and C. J. Hegarty, *Understanding GPS: Principles and Applications, Second Edition ed.*, Norwood, MA: Artech House, 2006.

[4] E. D. Kaplan, *Understanding GPS: Principles and Applications, Artech House Telecommunication Library*, 1996.

[5] M. L. Whitehead and S. R. Miller, "Unbiased Code Phase Estimator for Mitigating Multipath in GPS," US Patent US 6,744,404 B1, 1st June 2004.

[6] M. L. Whitehead and S. R. Miller, "Unbiased Code Phase Discriminator," US Patent US 8,000,381 B2, 15th August 2011.

[7] "Global Navigation Satellite System GLONASS Interface Control Document," 1998 Version 4.0.

[8] "The European GNSS (Galileo) Open Service Signal In Space Interface Control Document Issue 1," September 2010.

[9] J. W. Betz, "Binary offset carrier modulatinos for radionavigation," *Navigation, Journal of the Institude of Navigation*, vol. 48, no. 4, pp. 227-246, 2002.

[10] T. S. Rappaport, *Wireless Communications Principles & Practice*, Upper Saddle River, New Jersey: Prentice Hall PTR, 1996.

[11] A. Steingass and A. Lehner, "Land mobile satellite navigation - characteristics of the multipath channel," in *Proceedings of the ION GNSS 2003*, Portland, Oregon, USA, 2003.

[12] J. Max, "Quantizing for Minimum Distortion," *IRE Trans. on Information Theory*, pp. 7-12, March 1960.

[13] J. K. Holmes, Coherent Spread Spectrum Systems, Malabar, FL,: Krieger Publishing Company, 1990.

[14] J. J. Jr. Spilker and D. T. Magill, "The Delay-Lock Discriminator-An Optimum Tracking Device," *Proc. IRE,* vol. 49, no. 9, pp. 1403-1416, 1961.

[15] W. J. Gill, "A comparison of binary delay-lock loop implementations," *IEEE Trans. Aerospace Electron. Syst.,* Vols. AES-2, pp. 415-424, July 1966.

[16] J. J. Jr. Spilker, "Delay-Lock Tracking of binary signals," *IEEE Trans. Space Electron. Telemetry,* Vols. SET-9, p. 18, March 1963.

[17] A. Polydoros and C. Weber, "Analysis and Optimization of Correlative Code-Tracking Loops in Spread-Spectrum Systems," *IEEE Trans. On Comm.,* vol. 33, no. 9, pp. 30-43, January 1985.

[18] W. J. Hurd and T. O. Anderson, "Digital Transition Tracking Symbol Synchronizer for LOW SNR Coded Systems," *IEEE Trans. On Comm. Tech.,* Vols. COM-18, no. 2, April 1970.

[19] J. K. Holmes and C. C. Chen, "Acquisition Time Performance of PN Spread-Spectrum Systems," *IEEE Trans. On Comm.,* Vols. COM-25, no. 8, pp. 778-784, August 1977.

[20] L. I. Basilio, J. T. Williams, D. R. Jackson and M. A. Khayat, "A Comparision Study of a New GPS Reduced-Surface-Wave Antenna," *Antennas and Wireless Propagation Letters,* vol. 4, pp. pp.233-236, 2005.

[21] R. D. Van Nee, J. Siereveld, P. Fenton and B. Townsend, "The Multipath Estimating Delay Lock Loop: Approaching Theoretical Accuracy Limits," in *Proc. IEEE Position, Location and Navigation Symposium,* Las Vegas, Nevada, USA, April 1994.

[22] J.-C. Juang, "Multi-Objective Approach in GNSS Code Discriminator Design," *IEEE Trans. on Aerospace and Electronic Systems,* vol. 44, no. 2, pp. 481-492, April 2008.

[23] F. D. Nunes, F. M. G. Sousa and J. M. N. Leitao, "Gating Functions for Multipath Mitigation in GNSS BOC Signals," *IEEE Transactions on Aerospace and Electronic Systems,* vol. 43, no. 3, pp. 951-964, July 2007.

[24] A. J. Van Dierendonck, P. Fenton and T. Ford, "Theory and Performance of Narrow Correlator Technology in GPS Receiver," *Navigation, Journal of The Institute of Navigation,* vol. 39, no. 3, pp. 265-283, 1992.

[25] V. A. Veitsel, A. Zhdanov and M. Zhodzishsky, "The Mitigation of Multipath Errors by Strobe Correlators in GPS/GLONASS Receivers," *GPS Solutions,* vol. 2, no. 2, pp. 38-45, 1998.

[26] H. Chen, W. Jia, J. Ren and M. Yao, "Unambiguous S-Curve Shaping Technique for Multipath Mitigation in Cosine-BOC Signals," *IEEE Communications Letters,* vol. 16, no. 11, pp. 1725-1728, November 2012.

[27] C. Counselman, "Multipath-Rejecting GPS Antennas," *Proceedings of the IEEE,* vol. 87, no. 1, January 1999.

[28] Simon, *IEEE Trans. On Comm. Tech.*, 1970.

[29] Miller, "Sample Clock to Chip Rate Ratio and its Effect on Code Estimation Bias," *Unpublished,* 2004.

[30] B. Townsend and P. Fenton, "A Practical Approach to the Reduction of Pseudorange Multipath Errors in a L1 GPS Receiver," in *Proceedings of ION GPS-94*, Salt Lake City, 1994.

[31] L. Garin and J. Rousseau, "Enhanced Strobe Correlator Multipath Rejection for Code & Carrier," in *Proceedings of ION GPS-97*, Kansas City, 1997.

[32] B. Schnaufer and G. McGraw, "A Peak Tracking/Measurement Compensation Multipath Mitigation Technique," in *Proceedings of the IEEE Position, Location and Navigation Symposium (PLANS 2000)*, San Diego, 2000.

[33] L. Weill, "A GPS Multipath Mitigation by Means of Correlator Reference Waveform Design," in *Proc. Institute of Navigation National Technical Meeting*, 1997.

[34] J. K. Ray, "Mitigation of GPS Code and Carrier Phase Multipath Effects using a Multi-Antenna System," University of Calgary, Calgary, Ph.D. Thesis UCGE Report No. 20136, 2000.

[35] M. S. Braasch, "Isolation of GPS Multipath and Receiver Tracking Errors," *Navigation: Journal of the Institute of Navigation,* vol. 41, no. 4, Winter 1994-1995.

[36] R. D. Van Nee, "Spread Spectrum Code and Carrier Synchronization Errors Caused by Multipath and Interference," *IEEE Trans. on Aerospace and Electronic Systems,* vol. 29, no. 4, pp. 1359-1365, October 1993.

[37] B. Townsend, P. Fenton, K. Van Dierendonck and R. D. Van Nee, "L1 Carrier Phase Multipath Error Reduction Using MEDLL Technology," in *Proceedings of ION GPS-95*, Palm Spring, 1995.

[38] F. Sousa, F. Nunes and J. Leitao, "Strobe Pulse Design for Multipath Mitigation in BOC GNSS Receivers," in *Position, Location, And Navigation Symposium, 2006 IEEE/ION*, 2006.

[39] A. Jovanovic, Y. Tawk, C. Botteon and P. A. Farine, "Multipath Mitigation Techniques for CBOC, TMBOC and AltBOC Signals using Advanced Correlator Architectures," in *Position Location and Navigation Symposium (PLANS, 2010 IEEE/ION*, Indian Wells, CA, USA, 2010.

[40] J. H. Wu and A. G. Dempster, "BOC-Gated-PRN', a multipath mitigation technique for BOC(n,n) waveforms," *IEEE Transaction on Aerospace and Electronic Systems*, vol. 46, no. 4, pp. 1136-1153, 2011.

[41] M. S. Braasch, "Performance Comparison of Multipath Mitigiating Receiver Archiectures," in *Proc. 2001 IEEE Aerospace Conference*, 2001.

[42] S. Yoo, S. H. Yoo, E. Lee, S. Y. Kim and S. Yoon, "A Novel Tracking Scheme for Band-Limited GNSS," *IEEE*, pp. 228-229, 2007.

[43] B. Krach, M. Lentmaier and P. Robertson, "Bayesian Detection and Tracking for Joint Positioning and Multipath Mitigation in GNSS," in *Proceedings of the 5th Workshop on Positioning, Navigation and Communications 2008 (WPNC'08)*, 2008.

[44] M. Sahmoudi and M. G. Amin, "Fast Iterative Maximum-Likelihood Algorithm," *IEEE Transactions on Wireless Communications*, vol. 7, no. 11, pp. 4362-4374, November 2008.

[45] W. Sun and M. G. Amin, "Interference Suppression and Multipath Mitigation for Global Navigation Satellite Systems," pp. 140-143, 2004.

[46] P. Closas, C. Fernandez-Prades, J. Fernandez-Rubio and A. Ramirez-Gonzales, "Multipath mitigation using particle filtering," in *Proceedings*, Fort Worth, Texas, USA, 2006.

[47] P. Closas, C. Fernandez-Prades, J. Fernandez-Rubio and A. Ramirez-Gonzales, "A Bayesian Approach to Multipath Mitigation in GNSS Receivers," *IEEE JOURNAL OF SELECTED TOPICS IN SIGNAL PROCESSING*, vol. 3, no. 4, pp. 695-706, August 2009.

[48] M. Sahmoudi, R. J. Landry and F. Gagnon, "A Constrained Maximum-Likelihood Approach for efficient Multipath Mitigation in GNSS Receivers," in *IEEE/SP 15th Workshop on Statistical Signal Processing*, 2009.

[49] X. Liu, P. Closas, J. Liu and X. Hu, "Particle Filtering and its Application for Multipath Mitigation with GNSS Receivers," 2010.

[50] N. Jardak, A. VervischPicois and S. N., "Multipath Insensitive Delay Lock Loop in GNSS Receivers," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 47, no. 4, pp. 2590-2609, October 2011.

[51] N. Blanco-Delgado and F. D. Nunes, "Multipath Estimation in Multicorrelator GNSS Receivers using the Maximum Likelihood Principle," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 48, no. 4, pp. 3222-3233, October 2012.

[52] T. Irsigler and B. Pany, "S-curve shaping: A new method for optimum discriminator based code multipath mitigation," in *Proceedings of the Satellite Division Technical Meeting (ION GNSS 2005)*, Long Beach, CA, Sept. 2005.

[53] J. Pany, H. G. Avila-Rodriguez, "Bounds on signal performance regarding multipath-estimating discriminators," in *Proceedings of the Satellite Division Technical Meeting (ION GNSS 2006)*, Fort Worth, TX, Sept. 2006.

[54] P. V. Hung, D. N. Chien and N. V. Khang, "A Novel Multipath Mitigation Technique for GNSS Software Receivers," in *The 2012 International Conference on Advanced Technologies for Communications (ATC 2012)*, 2012.

APPENDIX A

GPS L1 C/A SPREADING CODE CHARACTERISTICS

The GPSL1CA spreading code is generated from the sum of two Gold Code sequences, G1(x) and G2(x). The G1(x) sequence is the same for all space vehicles. Each satellite CA CDMA sequence is unique and is determined by a different G2(x) polynomial for each satellite. Gold codes have a near ideal autocorrelation function: Unity at zero lag and $1/N_{EPOCH}$ for non-zero lag. Where N_{EPOCH} is the sequence epoch length and is the number of chips before the sequence repeats. Since the GPS Gold codes utilize a 10 bit shift register and are maximum length sequences, $N_{EPOCH} = 2^{10}-1$. When N_{EPOCH} is large, the autocorrelation is nearly an impulse function, and therefore provides nearly orthogonal CDMA code sequences.

The GPS CA code is generated by the modulo two sum of the G1(x) and G2(x) gold code sequences as illustrated in Fig. A-1. The G1(x) polynomial and initial seed of 0x3FF is the same for all space vehicles. Each satellite CA code sequence is unique and determined by a different G2(x) polynomial for each satellite. Two distinct methods for generating the G2 code sequence exist: Phase tap method and the phase delay method.

The phase tap method is illustrated in Fig. A-1. This method utilizes the closure property of addition for a gold code sequence: Given the sequences {a} and {b} generated from the polynomial G2(x) using different initial conditions $x=x_A$ and $x=x_B$; then c={a}+{b} is also a sequence generated by G2(x) using the initial condition $x=x_C$[2]. The tap selection matrix picks and adds the two different shift taps, T1 and T2, to form the final G2 sequence. The total number of unique sequences equals the combination of 10 items taken 2 at a time (unordered, no replacement) and equals 45. Eight of the sequences have been removed due to their poor cross and autocorrelation functions.

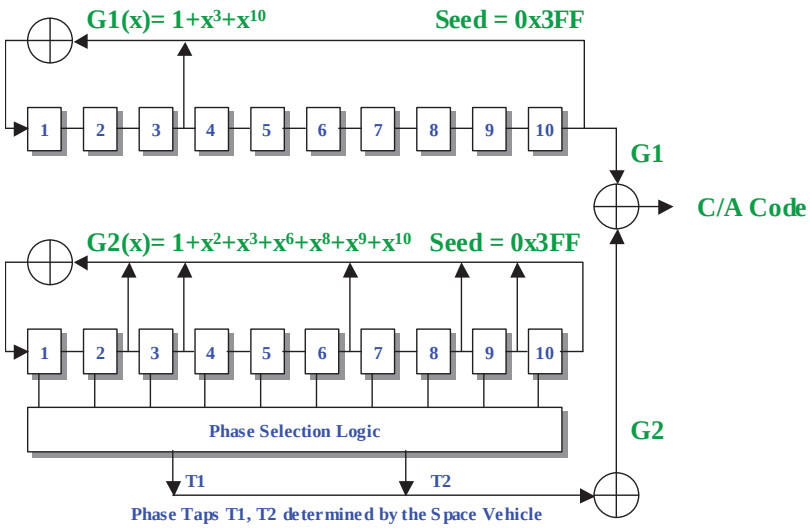

Fig. A-1: Phase Tap Method for Generating C/A Gold Code Sequence

The phase delay method is illustrated in Fig. A-2 and also utilizes the closure property of addition. This scheme simply loads the G2(x) polynomial with the correct initial seed of $x=x_C$ which removes the tap selection logic of Fig. A-1.

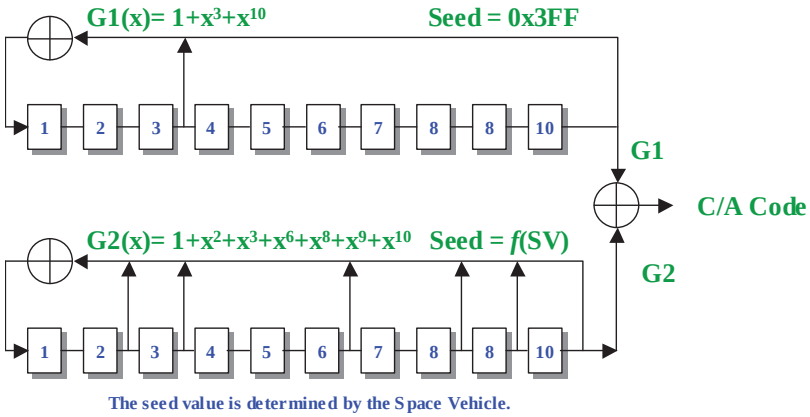

Fig. A-2: Phase Delay Method for Generating C/A Gold Code Sequence

Table A-1 lists the equivalent G2(x) sequence delay, phase tap selections of Fig. A-1, and the seed value for the phase delay method of Fig. A-2.

Table A-1: Space Vehicle G2 Gold Code Generator Phase Taps and Seed Values

SV	G2 Delay	G2 Phase Taps Fig. A-1	G2 Seed Fig. A-2		SV	G2 Delay	G2 Phase Taps Fig. A-1	G2 Seed Fig. A-2
1	5	2 ⊕ 6	0x03EC		17	469	1 ⊕ 4	0x0226
2	6	3 ⊕ 7	0x03D8		18	470	2 ⊕ 5	0x004C
3	7	4 ⊕ 8	0x03B0		19	471	3 ⊕ 6	0x0098
4	8	5 ⊕ 9	0x0360		20	472	4 ⊕ 7	0x0130
5	17	1 ⊕ 10	0x0096		21	473	5 ⊕ 8	0x0260
6	18	2 ⊕ 8	0x012C		22	474	6 ⊕ 9	0x00C0
7	139	1 ⊕ 9	0x0196		23	509	1 ⊕ 3	0x00CE
8	140	2 ⊕ 10	0x032C		24	512	4 ⊕ 6	0x0270
9	141	3 ⊕ 3	0x0258		25	513	5 ⊕ 7	0x00E0
10	251	2 ⊕ 4	0x0374		26	514	6 ⊕ 8	0x01C0
11	252	3 ⊕ 5	0x02E8		27	515	7 ⊕ 9	0x0380
12	254	5 ⊕ 6	0x03A0		28	516	8 ⊕ 10	0x0300
13	255	6 ⊕ 7	0x0340		29	859	1 ⊕ 6	0x0056
14	256	7 ⊕ 8	0x0280		30	861	2 ⊕ 7	0x00AC
15	257	8 ⊕ 9	0x0100		31	862	3 ⊕ 8	0x0158
16	258	9 ⊕ 10	0x0200		32	863	4 ⊕ 9	0x02B0

Although the CA code is the sum of two gold codes, it is *not* a gold code sequence. The CA cross correlations are not zero and the autocorrelation is not time shift orthogonal. This is evident from the CA autocorrelation sequence for space vehicle one illustrated in Fig. A-3. The maximum autocorrelation is unity and occurs at lag 0. The maximum side-lobe is 0.075 which only provides a coding gain of 10log(14) = 11.46 dB. This substantially differs from the 30 dB {$10\log(N_{EPOCH})$} coding gain for GPS gold codes. The cross correlation between any two space vehicles also exhibits the 11.46 dB coding gain. Fig. A-4 plots the cross correlation between space vehicle 1 and space vehicle 2.

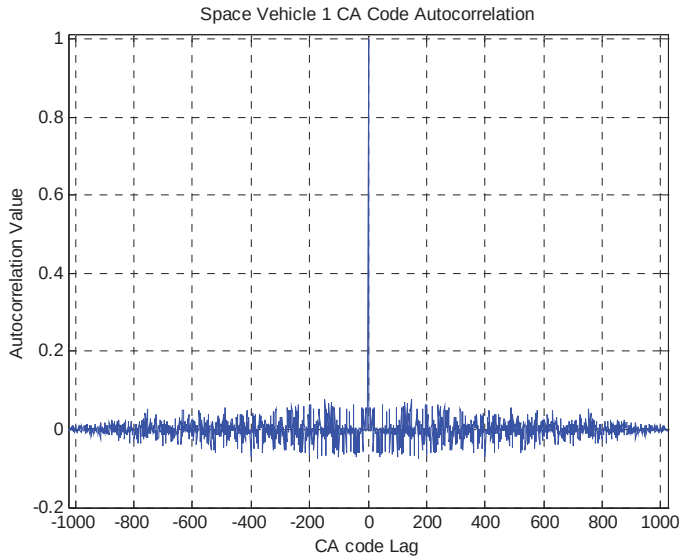

Fig. A-3: Space Vehicle 1 Autocorrelation

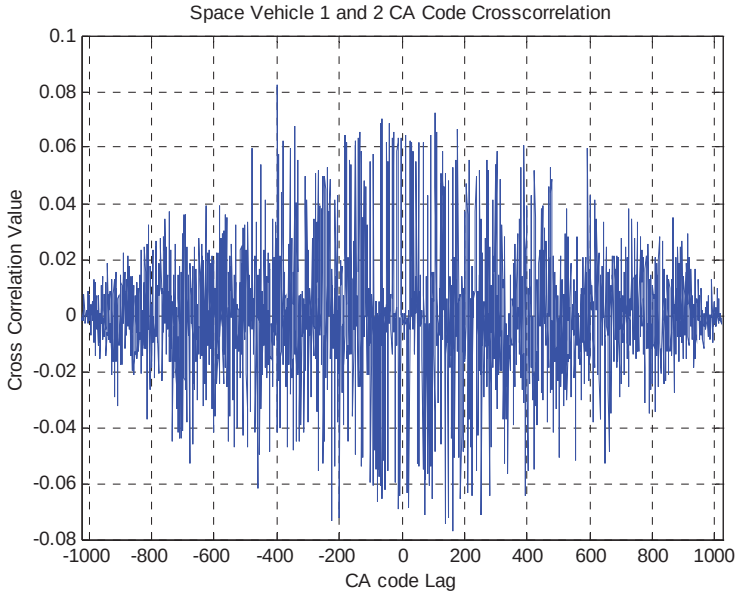

Fig. A-4: Space Vehicle 1 and 2 CA Cross Correlation

The entire CA code sequence was generated for the first 32 space vehicles. Logic 0 and 1 are mapped to +1 and -1 symbols respectively. A balanced random sequence has a sinx/x Power Spectral Density (PSD). Unbalanced, non-uniform transition densities distort the CA PSD from the ideal.

The code transition densities were analyzed and tabulated in Table A-2 for the four possible types: Low-to-low, high-to-high, low-to-high and high-to-low. Non-uniform transition counts were measured for space vehicles 7, 8, 15, 17, 21, 22 and 24 which can be divided into two groups with {8, 22} and {7, 15, 17, 21, 24}. The other space vehicles exhibited uniform transition counts with the high-to-high transition count one less than the others since the CA sequence contains one less +1 symbol than a -1. The transition densities are accumulated across the entire CA sequence. However, localized imbalances may occur.

Fig. A-5 plots the partial sum of CA symbols as a function of the CA symbol index for space vehicle 1. Recall that space vehicle 1 has a balanced transition density. Note that from chip index 230 to 280 the sequence gain 21 additional -1 symbols than +1symbols. This change in transition density (long string of -1's) enables the carrier to leak through which produces tones in the GPS PSD. Note that the partial sum is -1 for the last chip which indicates the sequence is balanced when measured across the entire epoch.

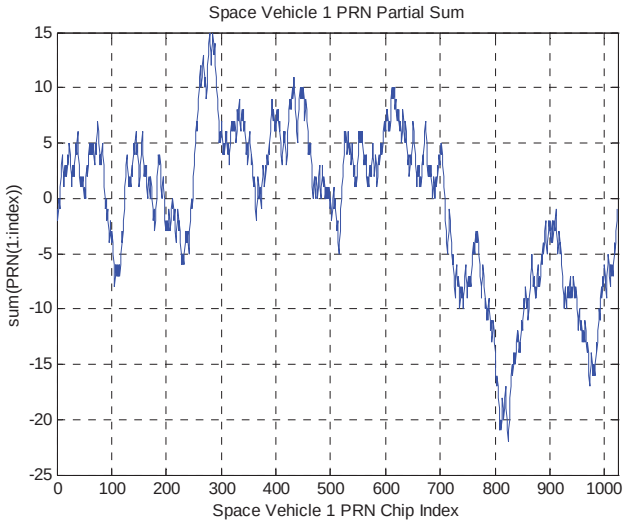

Fig. A-5: Space Vehicle 1 CA Partial Sum

Table A-2: GPS L1CA Code Transition Density

Clock-Edge Events within the 1023 chip-length C/A code of GPS							
PRN	Total Non-Transitions		Total Sign Transitions		Total L-H and H-L (Polarity Transitions)	Total Non Transitions	Difference (previous two columns)
	L-L	H-H	L-H	H-L			
1	256	255	256	256	512	511	1
2	256	255	256	256	512	511	1
3	256	255	256	256	512	511	1
4	256	255	256	256	512	511	1
5	256	255	256	256	512	511	1
6	256	255	256	256	512	511	1
7	272	271	240	240	480	543	-63
8	240	239	272	272	544	479	65
9	256	255	256	256	512	511	1
10	256	255	256	256	512	511	1
11	256	255	256	256	512	511	1
12	256	255	256	256	512	511	1
13	256	255	256	256	512	511	1
14	256	255	256	256	512	511	1
15	272	271	240	240	480	543	-63
16	256	255	256	256	512	511	1
17	272	271	240	240	480	543	-63
18	256	255	256	256	512	511	1
19	256	255	256	256	512	511	1
20	256	255	256	256	512	511	1
21	272	271	240	240	480	543	-63
22	240	239	272	272	544	479	65
23	256	255	256	256	512	511	1
24	272	271	240	240	480	543	-63
25	256	255	256	256	512	511	1
26	256	255	256	256	512	511	1
27	256	255	256	256	512	511	1
28	256	255	256	256	512	511	1
29	256	255	256	256	512	511	1
30	256	255	256	256	512	511	1
31	256	255	256	256	512	511	1
32	256	255	256	256	512	511	1

APPENDIX B

EARLY MINUS LATE DISCRIMINATOR DERIVATION

An Early minus Late (EML) code synchronization algorithm is developed and its performance with respect to data transition density, carrier frequency offsets, and symbol timing frequency offsets is analyzed. Furthermore, a symbol lock detector is proposed and its performance investigated. The symbol synchronization algorithm operates on complex, near base-band data which consists of multiple samples per symbol to determine the symbol timing phase. Fig. B-1 illustrates the early-late code phase tracking timing diagram which is used to calculate the early-late code discriminator.

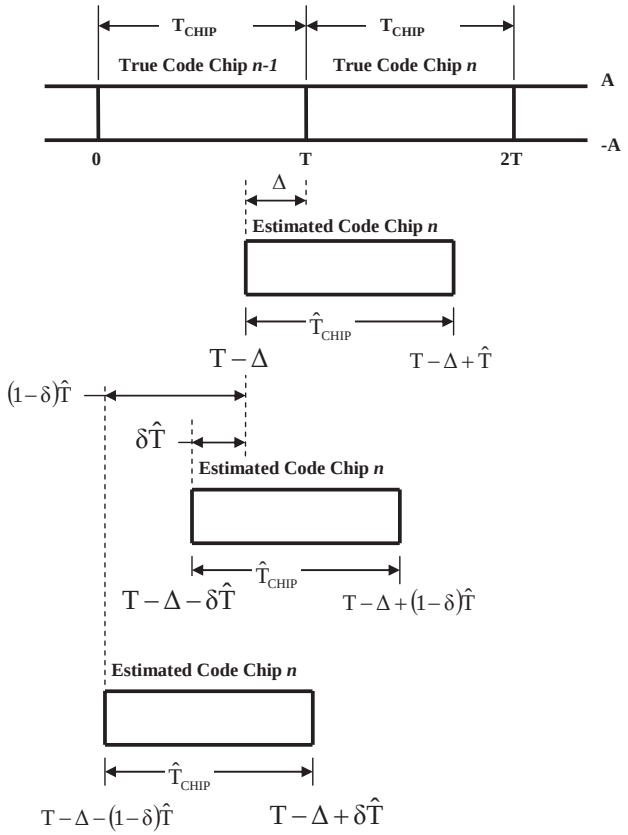

Fig. B-1: Early-Late Code Phase Tracking Timing Diagram

Where *A* is the code chip amplitude; *T* is the true code chip period; $\hat{T}$ is the current estimate of the code chip period and will deviate from truth due to doppler and self-noise; Δ is the timing error between the start of the next true code chip and the start of the next estimated

code chip; and finally, δ is the early and late code spacing expressed as a fraction of the estimated code chip period. Consider a code sequence with transition probability, *p*; then the probability that the next code chip has the same amplitude as the current code chip (non-transition probability) is *(1-p)*. A correlation metric is then calculated for the prompt, early and late channels considering both the chip transition and chip non-transition case.

When no transition occurs, the expected value of the prompt correlator is *A*. This occurs with probability, *(1-p)*. When a transition occurs, the expected value of the prompt correlator is given by,

$$E[\text{prompt}] = \frac{-1}{\hat{T}}\int_{T-\Delta}^{T} A\,dt + \frac{1}{\hat{T}}\int_{T}^{T-\Delta+\hat{T}} A\,dt\ , \tag{B-1}$$

$$E[\text{prompt}] = \frac{-A[T-(T-\Delta))]}{\hat{T}} + \frac{A[T-\Delta+\hat{T}]}{\hat{T}}, \tag{B-2}$$

$$E[\text{prompt}] = \frac{-A\Delta}{\hat{T}} + \frac{A[\hat{T}-\Delta]}{\hat{T}}, \tag{B-3}$$

$$E[\text{prompt}] = \frac{A[\hat{T}-2\Delta]}{\hat{T}}. \tag{B-4}$$

Finally, combining the transition and non-transition expectations yields,

$$E[\text{prompt}] = A(1-p) + \frac{A[\hat{T}-2\Delta]p}{\hat{T}}, \tag{B-5}$$

$$E[\text{prompt}] = A - Ap + Ap - \frac{2A\Delta p}{\hat{T}}, \tag{B-6}$$

$$E[\text{prompt}] = A - \frac{2A\Delta p}{\hat{T}}. \tag{B-7}$$

The expected value of the non-transition early channel is simply *A* and occurs with probability *(1-p)*. The expected value of the early channel for the transition case is,

$$E\left[\text{early}\right] = \frac{-1}{\hat{T}}\int_{T-\Delta-\delta\hat{T}}^{T} A dt + \frac{1}{\hat{T}}\int_{T}^{T-\Delta+(1-\delta)\hat{T}} A dt \ , \tag{B-8}$$

$$E[\text{early}] = \frac{-A\left[T-\left(T-\Delta-\delta\hat{T}\right)\right]}{\hat{T}} + \frac{A\left[T-\Delta+(1-\delta)\hat{T}-T\right]}{\hat{T}}, \tag{B-9}$$

$$E[\text{early}] = \frac{-A\left(\Delta+\delta\hat{T}\right)}{\hat{T}} + \frac{A\left[(1-\delta)\hat{T}-\Delta\right]}{\hat{T}}, \tag{B-10}$$

$$E[\text{early}] = \frac{A\hat{T}(1-2\delta)-2A\Delta}{\hat{T}}. \tag{B-11}$$

The expected value of the non-transition late channel is simply *A* and occurs with probability *(1-p)*. The expected value of the late channel for the transition case is,

$$E\left[\text{late}\right] = \frac{-1}{\hat{T}}\int_{T-\Delta-(1-\delta)\hat{T}}^{T} A dt + \frac{1}{\hat{T}}\int_{T}^{T-\Delta+\delta\hat{T}} A dt \ , \tag{B-12}$$

$$E[\text{late}] = \frac{-A\left[T-\left(T-\Delta-(1-\delta)\hat{T}\right)\right]}{\hat{T}} + \frac{A\left[T-\Delta+\delta\hat{T}-T\right]}{\hat{T}}, \tag{B-13}$$

$$E[\text{late}] = \frac{-A\left[\Delta+(1-\delta)\hat{T}\right]}{\hat{T}} + \frac{A\left(\delta\hat{T}-\Delta\right)}{\hat{T}}, \tag{B-14}$$

$$E[\text{late}] = \frac{-A\hat{T}\left[(1-2\delta)+2A\Delta\right]}{\hat{T}}. \tag{B-15}$$

The discriminator metric is,

$$J_{\Delta} = \left(E[\text{early}] - E[\text{late}]\right)\hat{x}, \tag{B-16}$$

where $\hat{x}$ is the model spreading code. This has the effect of wiping off the sign for the early and late statistic. In non-DSSS systems an absolute value is taken instead of multiplying by the code model.

When no transition occurs, the metric is,

$$J_{\Delta} = A(1-p) - A(1-p) = 0\,, \tag{B-17}$$

which makes intuitive sense since no timing information is contained in a non-transitioning sequence. When transitions occur, (B-11), (B-15) and (B-16) are combined to form the metric,

$$J_{\Delta} = \frac{p\left[A\hat{T}(1-2\delta) - 2A\Delta\right]}{\hat{T}} - \frac{p\left[A\hat{T}(1-2\delta) + 2A\Delta\right]}{\hat{T}}, \tag{B-18}$$

$$J_{\Delta} = \frac{p\left[\left(A\hat{T}(1-2\delta) - 2A\Delta\right) - \left(A\hat{T}(1-2\delta) + 2A\Delta\right)\right]}{\hat{T}}, \tag{B-19}$$

$$J_{\Delta} = \frac{p\left[\left(A\hat{T}(1-2\delta) - A\hat{T}(1-2\delta) - 2A\Delta - 2A\Delta\right)\right]}{\hat{T}}, \tag{B-20}$$

$$J_{\Delta} = \frac{-4Ap\Delta}{\hat{T}}, \tag{B-21}$$

Many factors affect the S-curve: Noise, carrier offset, signal amplitude and transition probability. All factors will reduce the phase detector gain, which directly reduces the open loop gain. The dependence on amplitude and transition density can be eliminated by employing an automatic gain controller (AGC) such as the Lloyd-Max quantizer.

The symbol tracking loop can function with a small carrier offset which is some fraction of the symbol rate. However, the carrier offset has the effect of reducing the average signal amplitude with reduces the phase detector gain.

APPENDIX C

CARRIER NCO ANALYSIS AND CORRELATION SELF NOISE

Coherent spread spectrum receives require knowledge of all signal phases to perform optimal detection. The required phase information includes the carrier phase and frequency as well as the spreading code sequence, phase and frequency. Knowledge of signal phase is obtained by comparing the incoming signal to a precisely controlled replica; and then adjusting the replica to minimize the error between it and the true signal. A numerically controlled oscillator (NCO) is used to generate and precisely control the replica. The GNSS carrier NCO is used within the carrier tracking loop to translate the signal to baseband, track out up to ±5 KHz of doppler frequency shift and precisely measure the carrier cycles and phase. A detailed general analysis of NCO properties is found in [Analog devices tech document]. The following discussion provides specific values that are found in some commercial GPS L1CA receivers.

Consider the $\cos(\theta)$ function. The phase at time, *t*, is given by $\theta = \theta_o + 2\pi ft$ radians. Where θ_o is the starting phase and *f* is the frequency. Now consider a digital representation with a sample period, T_{sample}, and a constant frequency over the sample period, denoted $\dot{\theta}$,

$$\theta = \theta_o + 2\pi\dot{\theta} \cdot NT_{sample}, \tag{C.1}$$

with $t = N \cdot T_{sample}$. A 1st order difference equation is developed from,

$$\begin{gathered} \theta = \theta_o + 2\pi\dot{\theta} \cdot (N-1)T_{sample} + 2\pi\dot{\theta} \cdot T_{sample}, \\ \theta = \theta_o + \left(2\pi\dot{\theta}T_{sample}\right) \cdot (N-1) + \left(2\pi\dot{\theta} \cdot T_{sample}\right), \\ \theta = \theta_o + \sum_0^N \left(2\pi\dot{\theta}T_{sample}\right). \end{gathered} \tag{C.2}$$

Let $M = 2\pi\dot{\theta} \cdot T_{sample}$ which defines a phase increment that is accumulated every sample period. This notion of a phase accumulator forms the basis for the NCO. The NCO structure is illustrated in Fig C-1 and consists of two components: (1) A phase accumulator and (2) a phase to amplitude converter.

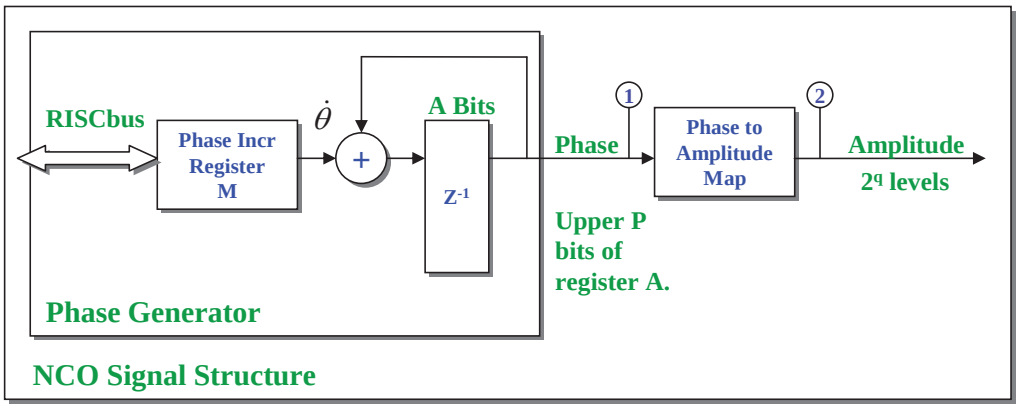

Fig. C-1: NCO Signal Structure and Local Oscillator Generation.

The phase increment register M, holds the phase accumulation that will occur every T_{sample} seconds. The accumulated phase is stored in a finite length accumulator (*A* bits) which provides a phase resolution of $2\pi / 2^A$ radians. The output frequency of the NCO is given by,

$$f_o = \frac{f_s}{2^A} \cdot M \ , \qquad \text{(C.3)}$$

where $f_s = 1/T_{sample}$. The smallest output frequency adjustment is $f_o = f_s / 2^A$ which corresponds to M=1.

The phase is then converted to the appropriate amplitude utilizing a phase to amplitude look up table or a CORDIC processor. Only a phase to amplitude look up table will be considered. A look up table scheme naturally quantizes both the *phase* (address) and the *amplitude* (data): Only the upper *P* bits of the phase accumulator, *A*, are used to

address the 2^P amplitude entries, where each entry is represented by Q bits. The quantization and truncation of the phase accumulator, phase address and amplitude data is a non-linear and noisy process. Consequently, the power spectrum of f_o is not a pure tone, but may include considerable phase noise.

Consider the following NCO common to GPS receivers: A=27 bits, P=3 bits, B=2 bits, M=0x01F7B1B9, and $f_s = 40.0/7 = 5.\overline{714285}$ MHz.

The phase accumulator value over time is plotted in Fig. C-2 for a constant M. A constant M produces a phase ramp with the slope equal to the NCO output frequency.

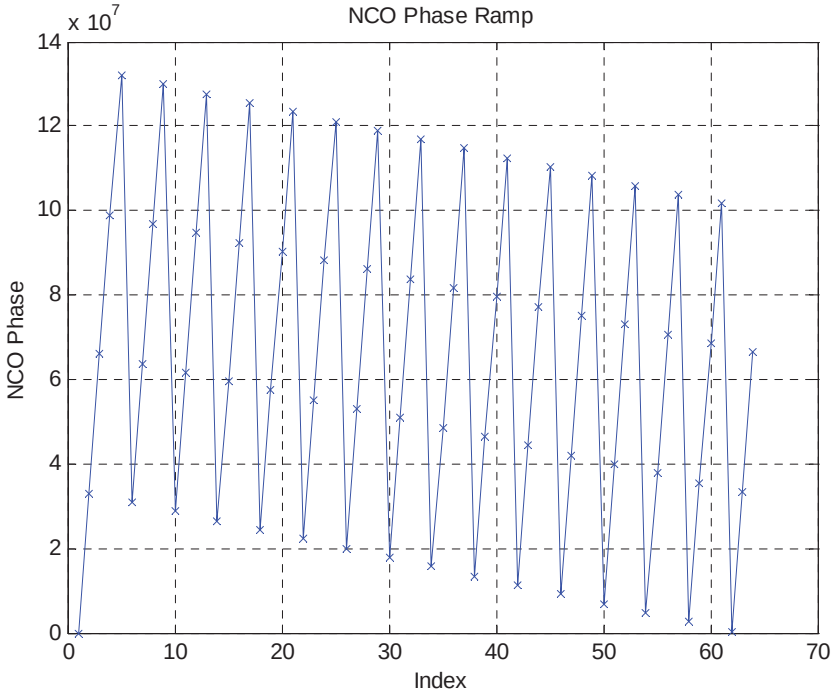

Fig. C-2: NCO Phase Accumulator Values.

Since the accumulator has finite length and will overflow, the accumulator phase is calculated modulo 2^A. Consequently, the phase may not start at identically zero ever cycle; but the zero-phase state will be periodic. The periodicity of the accumulator values creates a spectral tone at fs/GRR were GRR is the grand repetition rate. The GRR is given by,

$$GRR = 2^A / GCD(M_b, 2^A), \tag{C.4}$$

where M_b is the M register bit length (M_b=log2(M)), GCD is the Greatest Common Denominator of M_b and 2^A. The spectral tone, with the power given by (C.5) should either be positioned outside the SOI band or at a sufficiently low level so it does not degrade system performance. For this example the GRR is 62 samples.

The upper P bits of the phase word are utilized to generate the 2^P addresses into the amplitude table. This phase truncation can generate periodic table addresses as illustrated in Fig. C-3, which plots the $8=2^{P=3}$ table addresses as a function of the sample index (time). The periodicity of the phase address for the example is one-half the grand repetition rate and is clearly visible within Fig. C-3. Consequently, spectral lines are visible in the signal of interest power spectrum with a power level,

$$PTSM = -6.02P \text{ for } A - P > 4 \text{ dBc.} \tag{C.5}$$

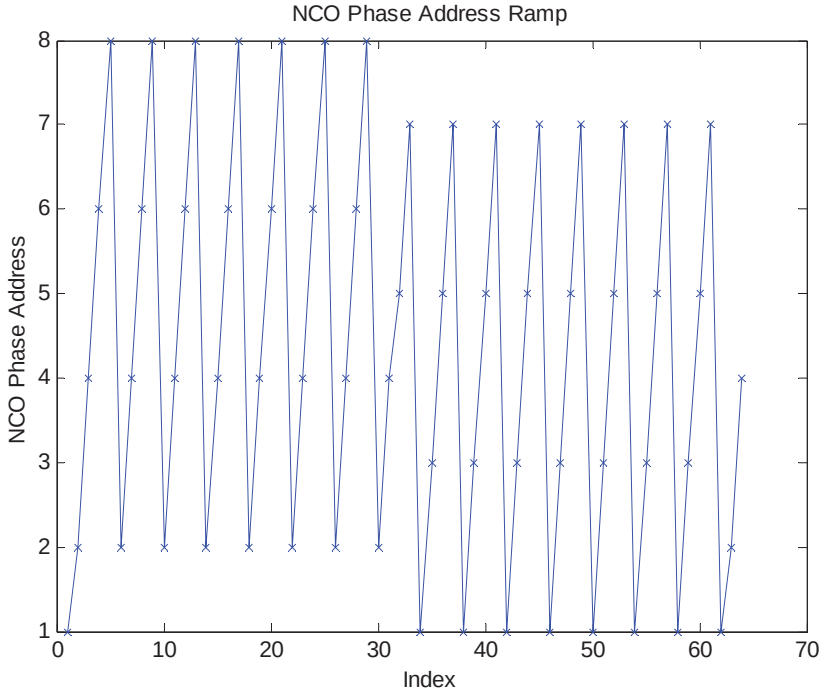

Fig. C-3: Periodicity of the Phase Address due to Phase Truncation to P Bits.

For the example with A=27 and P=3, phase truncation will result in spurs of no more than -18dBc regardless of the tuning word.

Moreover, the period will have the effect of modulating the NCO amplitude. The NCO phase to amplitude map is plotted in Fig. C-4 for both the sine and cosine function. The amplitude is quantized to 2 bits for 4 possible levels. The phase is mapped such that 30% of the phases are mapped to the maximum amplitude (+/-2 for the example) and the remaining 70% are +/-1. The time domain cosine sequence using the example NCO parameters is illustrated in Fig. C-5. Note the discontinuity at sample index 31. Prior to sample index 31 the amplitude exhibits “concave” distortion toward the left vertical axis. After sample index 31 the distortion converts to a “concave” distortion toward the right vertical axis.

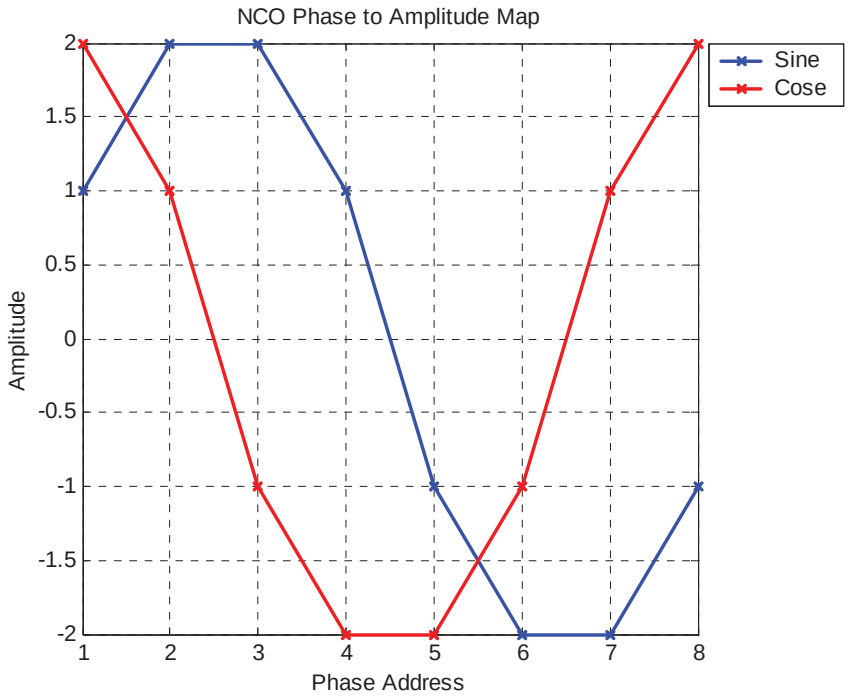

Fig. C-4: Phase to Amplitude Map: P=3, Q=2.

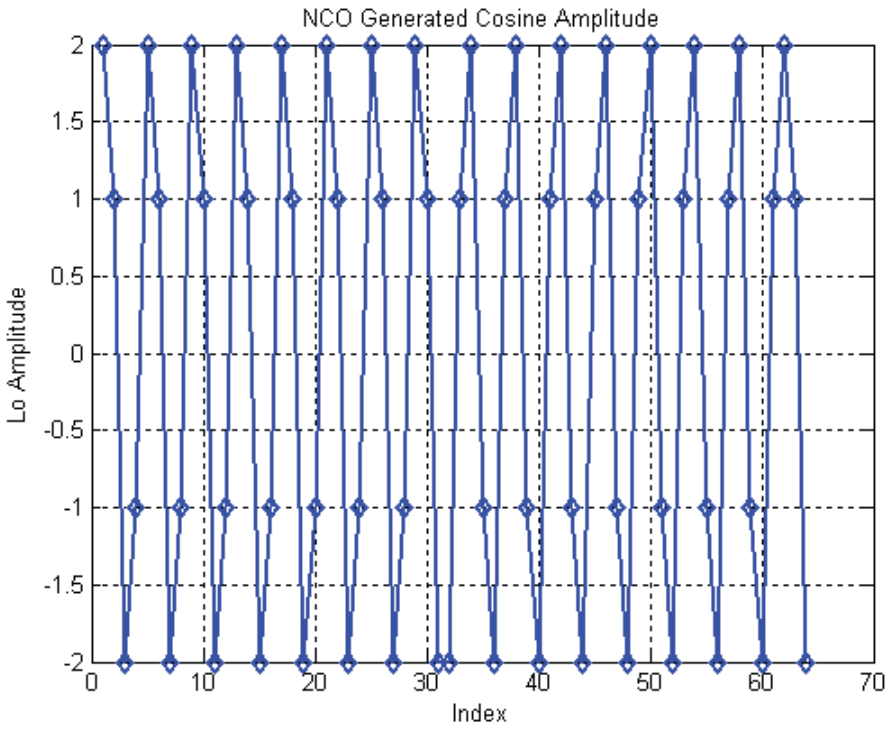

Fig. C-5: NCO Generated Cosine Amplitude: A=27, P=3, Q=2.

The NCO power spectral density is plotted in Fig. C-6 for the cosine amplitude map. The NCO has been tuned to 1.405 MHz by setting the phase increment, M, to 33,010,105 (0x01F7B1B9). Note that the largest spur is approximately -18 dBc as predicted by (C.5). Selection of the A, M, P and Q parameters are critical to obtain the desired NCO output frequency and tuning resolution while minimizing the required hardware (bits) and phase noise. It is desirable to select A and M to obtain the highest GRR to minimize spurs. The complex PSD is plotted in Fig. C-7 for the complex (cosine and sine) NCO output.

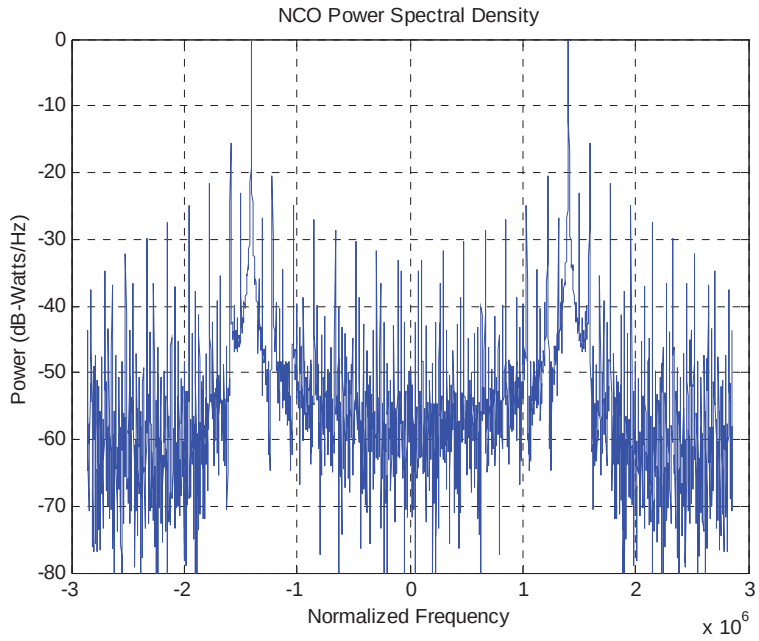

Fig. C-6: NCO Power Spectral Density: A=27, P=3, Q=2, M=0x01F7B1B9

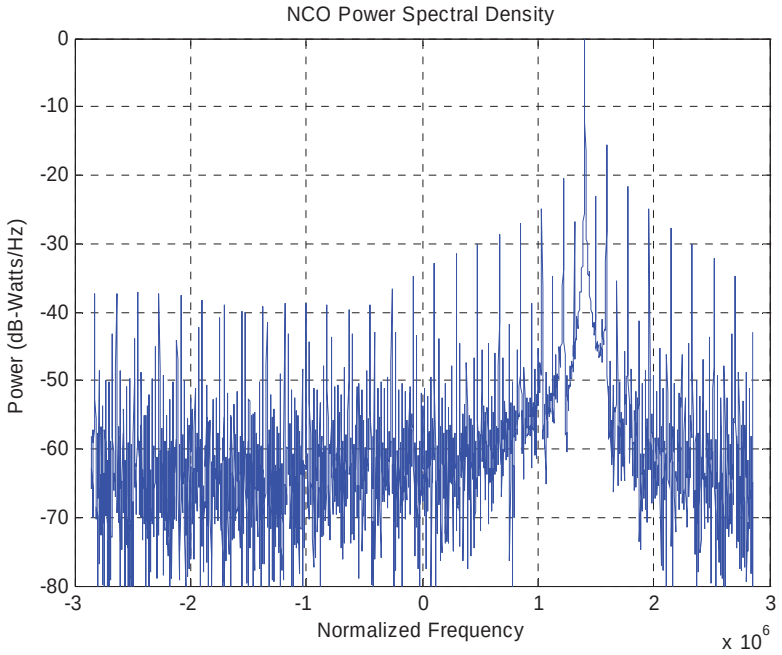

Fig. C-7: NCO Complex Power Spectral Density: A=27, P=3, Q=2, M=0x01F7B1B9

The carrier NCO contributes to the correlation self-noise which sets the 0 dB SNR noise floor and determines the carrier and code lock acquisition and tracking thresholds. With perfect code alignment, the correlation process has the form,

$$z = \sum_{i=0}^{N_{EPOCH}} x_i a_i \ . \tag{C.6}$$

where x_i is the input sample and a_i is the carrier NCO amplitude. The self-noise is a function of the sample and NCO probability distribution functions. The correlation process has zero mean self-noise, $\mu_z = 0$, which is guaranteed by the input AGC and the NCO design. The self-noise power, for a single correlator channel (I or Q), over one epoch is,

$$\sigma^2 = E\left[z \cdot z^*\right] - \underbrace{\mu_z}_{\equiv 0} \ . \tag{C.7}$$

Substituting in (C.6) into (C.7) yields

$$\sigma^2 = E\left[\left(\sum_{i=0}^{N_{EPOCH}} x_i a_i\right)\left(\sum_{j=0}^{N_{EPOCH}} x_j a_j\right)\right]. \tag{C.8}$$

Since the samples are modulated and therefore periodic, the expectations vanish when i≠j. The expectation when i=j is given by,

$$\sigma^2 = E\left[\left(\sum_{i=0}^{N_{EPOCH}} x_i^2 a_i^2\right)\right]. \tag{C.9}$$

Let P_x be the Input sample quantization level probability density function. It is set by the number of ADC bits, Q, and the automatic gain controller (AGC). The input samples are quantized to N_x =4 levels, {-3, -1, +1, + 3}. The AGC tracking profile sets the sample distribution of 70% for ±1 samples and 30% for ±3 samples based upon a 2 bit Lloyd-Max quantizer.

Let P_a be the Carrier NCO quantization level probability density function. The carrier NCO utilizes the upper three bits, P=3, to map eight phases, 23, into the associated amplitude. The amplitudes are quantized to 4 different levels. The phase to amplitude map

is given by {2, 1, -1, -2, -2, -1, 1, 2}. The M=0x01F7B1B9 tuning generates a complete carrier cycle in less than N_a=8 updates, therefore, not every phase amplitude is used each carrier cycle. However, the tuning word and NCO phase accumulator were designed such that every phase has equal probability of use over one code epoch. The expectation is removed by,

$$\sigma^2 = \sum_{m=1}^{N_x} P_x \sum_{k=1}^{N_a} P_a \left(\sum_{i=0}^{N_{EPOCH}} x_i^2 a_i^2 \right), \tag{C.10}$$

and,

$$\sigma^2 = \begin{array}{l} 0.35 \sum_{k=1}^{8} \frac{1}{8} \left(\sum_{i=0}^{N_{EPOCH}} 1^2 a_i^2 \right) + 0.35 \sum_{k=1}^{8} \frac{1}{8} \left(\sum_{i=0}^{N_{EPOCH}} (-1)^2 a_i^2 \right) + \\ 0.15 \sum_{k=1}^{8} \frac{1}{8} \left(\sum_{i=0}^{N_{EPOCH}} 3^2 a_i^2 \right) + 0.15 \sum_{k=1}^{8} \frac{1}{8} \left(\sum_{i=0}^{N_{EPOCH}} (-3)^2 a_i^2 \right) \end{array}. \tag{C.11}$$

Simplifying yields,

$$\sigma_z^2 = \left(2 \cdot (0.35 \cdot 1.25) + 2 \cdot (0.15 \cdot 11.25)\right) \cdot N_{EPOCH} = 48{,}571. \tag{C.12}$$

When considering both I and Q arms, the noise power is,

$$\left(I^2 + Q^2\right) = 2\sigma_z^2 = 97{,}142. \tag{C.13}$$