Adaptive High-Accuracy Timing Module: Algorithms and Performance Bounds

By

Hui Zhou

A thesis submitted to the Faculty of Graduate Studies and

Research in partial fulfillment of the requirements for the degree of

Master of Applied Science

Ottawa-Carleton Institute for Electrical and Computer Engineering

Faculty of Engineering

Department of Systems and Computer Engineering

Carleton University

Ottawa, Ontario, Canada

November 2009

The undersigned recommend to the Faculty of Graduate Studies and Research

acceptance of the thesis

Adaptive High-Accuracy Timing Module: Algorithms and Performance Bounds

Submitted by

Hui Zhou

in partial fulfillment of the requirements for

the degree of M.A.Sc. in Electrical Engineering

_____

Dr. Thomas Kunz, Thesis Supervisor

_____

Dr. Howard Schwartz, Thesis Supervisor

_____

Dr. Howard Schwartz,

Chair, Department of Systems and Computer Engineering

Carleton University

2009

Abstract

In this thesis, new adaptive OCXO frequency drift correction algorithms are proposed for the timing module on the base transceiver stations. The recursive system identification methods are used to replace the previous Batch Least Squares (BLS) method in the algorithm. Two different recursive system identification methods are evaluated and compared, the Recursive Least Squares (RLS) method and the Kalman Filter method.

New system models which include the digital control loop are created. Simulation results show that the new system model has better performance than the previous model.

The Cumulative Time Error (CTE) upperbound of the timing module is analyzed. This upperbound determines the performance bound of the timing module system. First, a simple model structure of the OCXO frequency stability is used to investigate the CTE upperbound. In this simple model, the temperature is linear related to the frequency stability. Then, a refined model structure is used to investigate the CTE upperbound. In this refined model, both temperature effect and ageing effect are considered. The control loop is included in both simple and refined model structures. The simulation results show that the CTE upperbound can be obtained analytically.

Acknowledgements

I would like to express my sincere gratitude to my supervisor Professor Thomas Kunz for his support in the last two years and the supervision of the thesis. His intellectual and continual guidance, encouragement throughout the whole research period are invaluable.

I would like also to express my deep gratitude to my co-supervisor Professor Howard Schwartz for his guidance throughout the research and the supervision of the thesis. His wide knowledge and challenging questions have been of a great value to me. His passion towards the work and dedication to the students are truly remarkable.

I would like to thank my parents and my wife for their tremendous support, inspiration, understanding and patience.

Table of Contents

List of Tables

List of Figures

List of Acronyms

ARMAX	AutoRegressive Moving Average with eXogenous inputs
BLS	Batch Least Squares
CDMA	Code Division Multiple Access
CTE	Cumulative Time Error
DAC	Digital to Analog Converter
DOCXO	Double Oven Controlled Crystal Oscillator
DF	Degree of Freedom
FPGA	Field Programmable Gate Array
GPS	Global Positioning System
OCXO	Oven Controlled Crystal Oscillator
PPB	Parts Per Billion
PPS	Pulse Per Second
RLS	Recursive Least Squares
RMS	Root Mean Square
RPEM	Recursive Prediction Error Method
RST	Radio System Time
SPXO	Simple Packaged Crystal Oscillator

TCXO	Temperature Compensation Crystal Oscillator
WiMAX	Worldwide Interoperability for Microwave Access

Chapter 1: Introduction

1.1 Overview

Time is very important not only for the daily schedules of human beings, but also for processing a sequence of events that happens in computers and for time-tagging information that flows through communication systems. Clock sources are essential for almost all electronic equipment and communication systems. Clock sources (another name is frequency control devices) can provide precise time and frequency information on which modern electronic equipment depends.

Quartz crystal oscillators are used as clock sources in the synchronization of distributed systems. One such system is a cellular network in which base station transceivers are operated within a specified time or frequency accuracy. Normal clocks such as clocks at home usually drift compared to the actual time. That is why one must regulate the time occasionally. The clocks in the base stations are much more accurate than clocks at home, but they drift too. The accuracy of clocks depends on their quality, the ambient temperature, and other environment variables. For example, a typical crystal oscillator such as MtronPTI's XO5120 SC-cut oscillator drifts within 173 microseconds when running for one day and the ambient temperature is within 0 to 70 celsius degrees [1].

The accuracy of oscillator is crucial to the normal operation of the cellular network.

Because GPS satellites are equipped with ultra-high accurate atomic clocks, the oscillators on the base stations are usually locked by the GPS signals which serve as timing reference signals and we call that the oscillators working in locked mode. This process is just like regulating a watch by a more accurate clock. In the event the GPS signal is lost, we call that the oscillators working in holdover mode. The clock accuracy of the oscillator on the base station is a function of the local environmental stimuli in holdover mode. Generally, a DOCXO (Double Oven Controlled Crystal Oscillator) is used in the timing module of the cellular network. The DOCXO is more accurate and therefore more expensive than the OCXO (Oven Controlled Crystal Oscillator) and the OCXO is not sufficient for the normal operation of the cellular network in holdover mode. However, the accuracy of the OCXO can be enhanced by an adaptive control module which guarantees that the OCXO can replace the DOCXO in the timing module. In order to improve the accuracy of the OCXO after losing the GPS signal, a system identification algorithm is trained by the adaptive control module when the timing reference signal is available. Accordingly, the relation between the time accuracy of the oscillator and environmental variables can be obtained and the effects of environment variables can be compensated. The accuracy of the oscillator can thus be enhanced to meet the needs of the cellular network.

Figure 1.1 shows a simple block diagram of the timing module system on the base station. First, GPS reference signals are received by the GPS signal receiver. Second, the correction signal generating module uses the GPS signals to generate the correction

signal. Third, the correction signal is used to improve the accuracy of the oscillator. Fourth, at the same time, the correction signal is used to feed the adaptive control module which trains the system identification algorithm. Last, when GPS reference signals are lost, the adaptive control module generates the correction signal to correct the oscillator. The detailed block diagram of the timing module system is much more complicated than this simple one and is shown in Chapter 3.

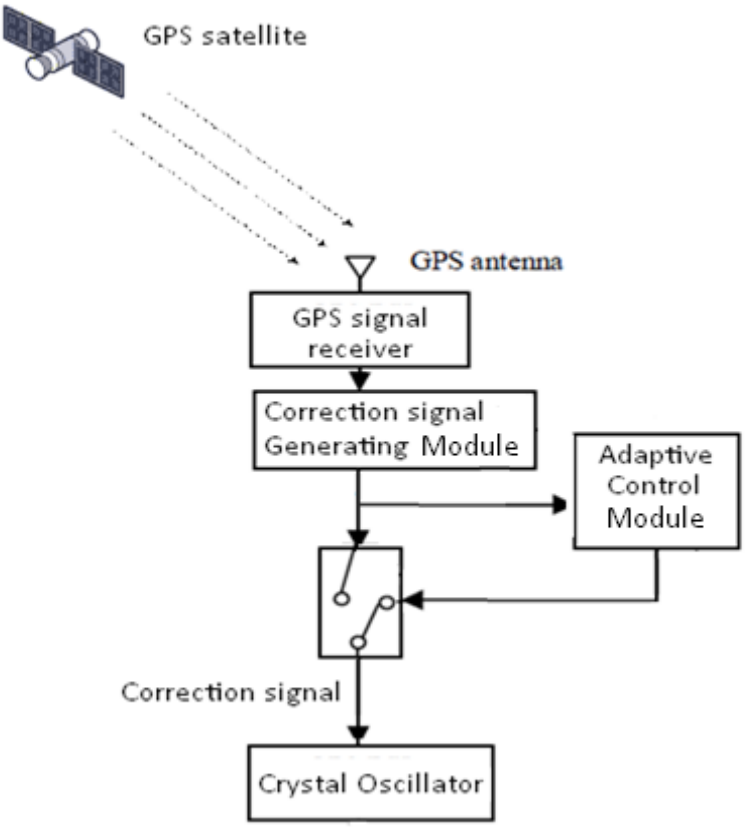

Figure 1.1 Simple Block Diagram of the Timing Module System

The following example can make the whole timing module system easy to

understand. A wrist watch is regulated through a more accurate clock. One discovers that the wrist watch drifts one minute everyday through the regulation process. Even if the clock is lost, one still can regulate the watch and increase its accuracy. In this example, the wrist watch is like the oscillator in the base station. The more accurate clock is like the timing reference signal from GPS satellites. Regulating the watch through the clock is like locking the oscillator by GPS signals. The algorithm training process is like getting the information that the watch drifts one minute every day.

The first stage in the creation of an accurate clock model is the identification and quantification of all significant frequency perturbing stimuli in terms of analytical expressions. The fundamental understanding of the parameters affecting the clock drift is paramount to determining the overall oscillator accuracy achievable by the system.

The second stage is the rational utilization of the proper system identification algorithms to identify the parameters which affect the accuracy of the oscillator. System identification is a mathematical term which describes the mathematical algorithms that build the mathematical models from measured input and output data. The mathematical model in this context is a mathematical description of the dynamic behavior of a process or a system in the time domain or the frequency domain [2]. The mathematical relation between the accuracy of the oscillator and the parameters which affect the frequency drift is an example for such physical system mathematical models. For instance, the frequency accuracy of quartz crystal oscillators over ambient temperature can be approximated by a quadratic function. System identification algorithms can determine the parameters of this

function through the environment temperature and the frequency accuracy measured.

System identification algorithms include linear, nonlinear, and hybrid identification, according to the characteristics of the models to be estimated. The accuracy and stability model of the clock source investigated in this thesis is a linear mathematical model. However, because of the limitations of the instrumental resolution in the control circuit, the model has some nonlinear characteristics. An accurate clock model can be created through using the correct accuracy and stability model and the suitable system identification.

Another research task is to estimate the accuracy of the enhanced oscillator in holdover mode. For example, in locked mode, the oscillators in CDMA base transceiver stations are required not to have a time error relative to Radio System Time (RST) greater than +/- 1μs. RST can be considered as the actual time. This target generally can be reached because the oscillators are locked by GPS signals. When the GPS timing reference signal is lost, the adaptive control module starts to create the correction signal to correct the oscillator. The oscillator still drifts and the drift will get worse over time because the adaptive control module cannot compensate the oscillator one hundred percent accurately. There is a cumulative time error (CTE) that exists in the oscillator relative to RST. In holdover mode, the oscillators in CDMA base transceiver stations are required not to exceed +/- 10 μs cumulative time error over an 8 hour time period relative to RST [3]. The accuracy of the oscillator needs to be estimated in holdover mode, which means the upperbound of the CTE needs to be estimated over a period of time in

holdover mode, given a specific system identification algorithm trained by the adaptive control module. The upperbound of the CTE determines how much the oscillator can drift relative to RST. This then determines whether or not the oscillator or the system identification algorithm is suitable for the base station timing module. For instance, if the upperbound of CTE over an 8 hour period in holdover mode is less than +/- $10\mu s$, the oscillator and the system identification algorithm are suitable for the CDMA base station timing module. If the upperbound of CTE over an 8 hour period in holdover mode is larger than +/- $10\mu s$, which means that sometimes the CTE cannot be tolerated, the oscillator or the system identification algorithm is not suitable for the CDMA base station timing module. In this case, either a better system identification algorithm or a more accurate and therefore expensive oscillator is needed.

1.2 Outline

The chapters in this thesis are organized as follows: Chapter 2 introduces background information about the elements which affect the frequency accuracy and stability of crystal oscillators. Crystal oscillators are major clock sources in modern electronic systems. The main system identification algorithms are introduced in this chapter as well, such as the Recursive Least Squares algorithm and the Kalman Filter algorithm. Chapter 3 reviews the state-of-the-art of Adaptive Oscillator drift correction algorithms. The timing module system in a base station is reviewed as well. The problem

statement is described in this chapter in detail. In Chapter 4, different system identification algorithms for the creation of an accurate clock source are evaluated. Chapter 5 addresses the CTE upperbound of clock sources in the system. A simple clock model is created. This clock model only includes the linear temperature effect on the oscillator accuracy. The parameter distribution of the clock model is investigated. In Chapter 6, a more detailed clock model which combines the effect of temperature and ageing are studied. The system identification algorithm and the CTE upperbound are investigated for this more detailed model. Chapter 7 presents the conclusions and proposes future work.

1.3 Contributions

The contributions of this thesis include:

- A new adaptive OCXO frequency drift correction algorithm is proposed. A recursive system identification method is used to develop the adaptive correction algorithm. A previous adaptive control algorithm for oscillators uses the Batch Least Squares (BLS) method. The BLS method needs a large memory and lots of computation. It also requires a matrix inversion computation which is complex to conduct. The recursive system identification method needs low memory and relatively less computation. The matrix inversion computation is not necessary. Therefore, the recursive system

identification method is more suitable for developing the adaptive correction algorithm.

- Two main recursive system identification methods are evaluated. They are the Recursive Least Squares (RLS) method and the Kalman Filter method. The characteristics of these methods are investigated and the more suitable one, the RLS method, is chosen.
- A new system model is created. The simulation results show that the new system model has better performance than the previous used model. The maximum cumulative time error of the new system model in simulations is lower than the old model.
- The CTE upperbound of the oscillator enhanced by the adaptive correction algorithm is investigated. There are no previous works to investigate the CTE upperbound of oscillators when a specific system identification algorithm is used to enhance oscillators. The CTE upperbound can determine the range of applications of the enhanced oscillator. It also determines whether or not the enhanced oscillator can replace a more expensive and more accurate oscillator.

Chapter 2: Background Information

In this chapter, the frequency accuracy and stability characteristics of crystal oscillators are reviewed first. Some key factors which impact the frequency accuracy and stability of oscillators such as temperature and ageing are reviewed. The reasons why these factors are critical are explained. In the second section of this chapter, the main system identification algorithms used in the thesis are introduced.

2.1 Frequency Accuracy and Stability Dependencies of Crystal Oscillators

In the modern world, a vibrating quartz crystal is the heart of nearly all frequency control devices. Quartz crystal oscillators provide relatively accurate time and are the sources of relatively precise frequency. Quartz crystal oscillators are electronic circuits which use the mechanical resonance of vibrating crystals of piezoelectric materials to create periodically varying electrical signals. The frequency stability, low cost and small size of quartz crystal oscillators have resulted in their ubiquitous usage as a frequency reference in electronic equipment. Crystal oscillators as frequency sources and frequency control components are most widely used in the time and frequency research and production fields, such as IT Industry, Communications, Electronic Instruments, Applied

Electronic Techniques, Measurements, Aerospace Systems, Military Industry, etc [4]. Table 2.1 shows the major applications of Quartz Crystal Oscillators.

Military and Aerospace	Research and Metrology	Industrial	Consumer	Automotive
Communications	Atomic clocks	Communications	Watches and clocks	Engine control, stereo, clock
Navigation	Instruments	Telecommunications	Cellular and cordless	Trip computer
IFF	Astronomy and	Mobile/cellular/portable	phones, pagers	
Radar	geodesy	radio, telephone and	Radio and hi-fi	
Sensors	Space tracking	pager	equipment	
Guidance systems	Celestial navigation	Aviation	Color TV	
Fuzes		Marine	Cable TV systems	
Electronic warfare		Navigation	Home computers	
Sonobuoys		Instrumentation	VCR and video camera	
		Computers	CB and amateur radio	
		Digital systems	Toys and games	
		CRT displays	Pacemakers	
		Disk drives		
		Modems		
		Tagging/identification		
		Utilities		

Table 2.1 Major Applications of Quartz Crystals [4]

The crystal resonator is the most important component of a crystal oscillator and the quartz crystal is the “heart” of it. Although some other materials like ceramic resonators have been developed, their frequency stability and accuracy cannot compare with quartz crystals. According to different accuracy, stability and cost requirements, different types of crystal oscillators are employed. The temperature dependence of the crystal resonance is generally recognized as a first-order perturbation of the frequency accuracy of the crystal oscillator. Compensation of the temperature dependence has resulted in a classification of crystal oscillators based on the different temperature control methods, such as SPXO (Simple Packaged Crystal Oscillator) which has no temperature

compensation, TCXO (Temperature Compensation Crystal Oscillator) which uses analog or digital temperature compensation circuits, OCXO (Oven Controlled Crystal Oscillator) which uses an oven to control crystal temperature and DOCXO (Double Oven Controlled Crystal Oscillator) which uses two temperature control ovens, one inside the other, to further improve the stabilization of the crystal temperature relative to variations in the ambient temperature.

2.1.1 Physical and Electrical Factors Affecting Crystal Oscillator Frequency Stability and Accuracy

The frequency accuracy of a crystal oscillator is the offset from the specified target frequency. The frequency stability of the oscillator is the spread of the measured oscillator frequency around its operational frequency in a period of time. Figure 2.1 shows accuracy and stability examples for a frequency source. Factors such as temperature, crystal ageing and retrace establish the frequency accuracy of the oscillator, whereas reference signal noise (if the oscillator is locked to a reference), tuning port noise, supply rail noise, and vibration establish the stability of the oscillator. With respect to applications reliant on synchronization, random frequency perturbations with zero mean are less significant compared to the frequency accuracy of the oscillator. The dependence of synchronization on oscillator frequency accuracy is because time error is

the integral of the frequency error.

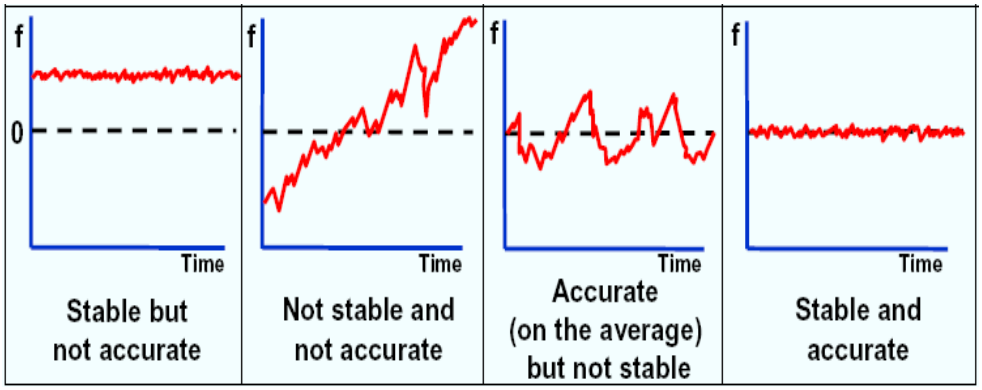

Figure 2.1 Accuracy and Stability Examples for a Frequency Source [5]

2.1.1.1 Temperature

Temperature is a significant factor which affects the frequency of resonators. Different crystal cuts have a different frequency-temperature characteristic. Figure 2.2 shows the frequency-temperature property of a typical AT-cut crystal resonator (here, AT, SC, or GT represents different crystal cut methods). The term φ represents the cut angle. One can see that crystals with different cut angles have different frequency-temperature curves. Some crystal resonator temperature characteristics are listed as follows:

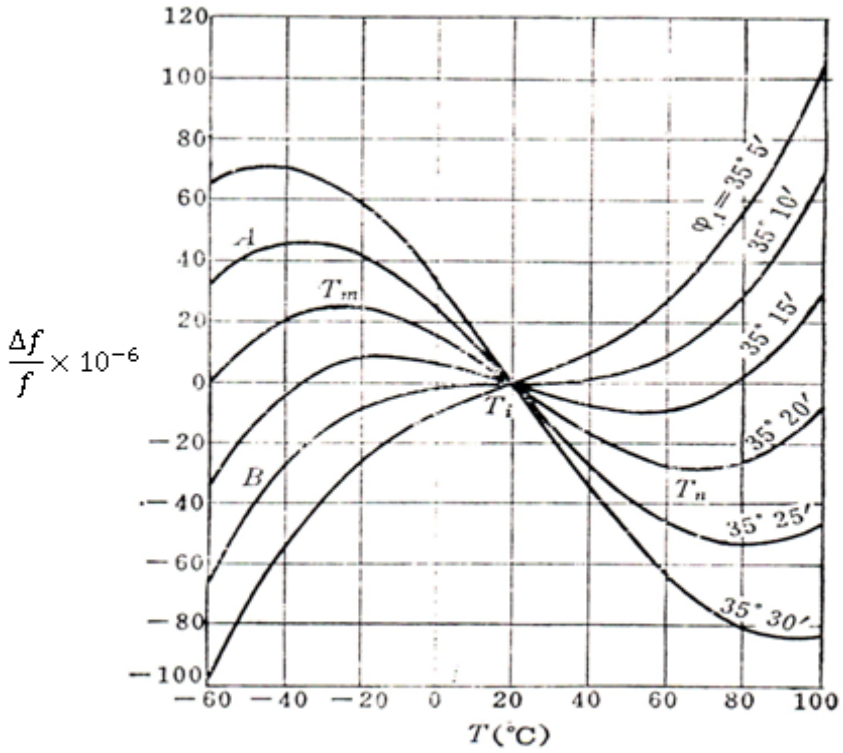

Figure 2.2 AT-cut Crystal Resonator Frequency-Temperature Properties [6]

1) The crystal cuts, in general, exhibit a cubic dependence on temperature [5].

2) In most situations, the zero temperature coefficient point can be changed through changing the angle between crystal wafer and crystal axis.

3) In a wide temperature range, like$-55\sim+105℃$, the relative frequency change of AT and GT cut crystals can be limited to $\pm 2\times 10^{-5}$ with suitable angle processing.

2.1.1.2 Ageing

The crystal resonator frequency changes according to the operational time and this

physical phenomenon is termed ageing. A representative ageing plot is shown in Figure 2.3. In this figure, the X-axis represents time and the unit is day. The Y-axis represents frequency accuracy and the unit is ppm (part per million). One can see that the ageing is generally not linear. However, when the ageing effect is considered as a period of only several hours, such as 24 hours, ageing can be considered linear approximately.

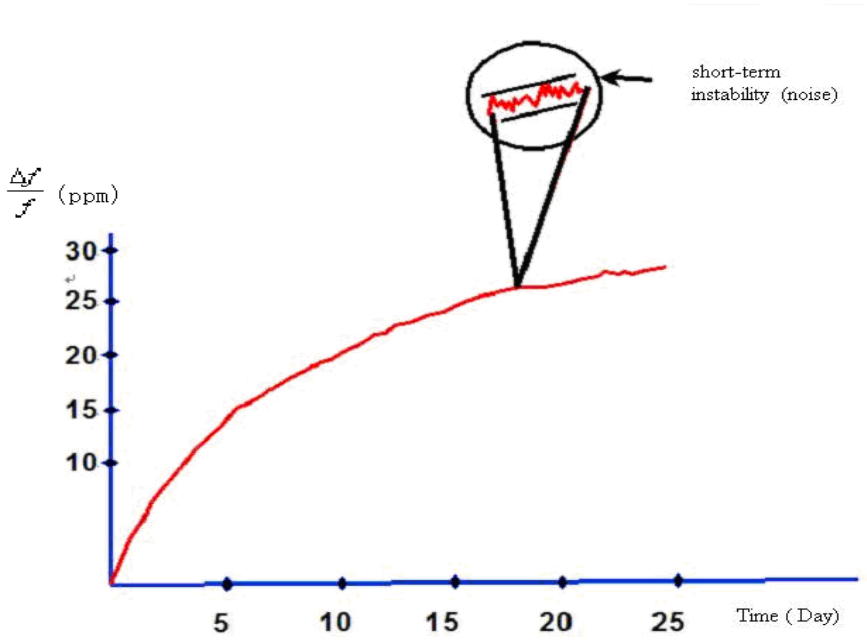

Figure 2.3 Ageing of Crystal Resonator [5]

It should be noted that although the plot is monotonic, this is not always the case and the ageing rate can reverse sign over time. When the vibration mode of a crystal wafer is Thickness-Shear, as in AT cut and SC cut crystals, ageing mostly results from:

1) Thermal gradient effect. This effect continues several minutes to several hours after thermal equilibrium [6]. Figure 2.4 shows the temperature gradient effects and

warm-up characteristics of two OCXOs, each containing an oven which reaches the thermal equilibrium in six minutes (Chapter “Warm-Up” in Reference [4] provides more information about the warm-up property of oscillators). One oven contains an AT-cut oscillator and the other oven contains an SC-cut oscillator. The frequency variation after six minutes comes from thermal gradient effects in Figure 2.4. One can see that an SC-cut OCXO has much better performance than an AT-cut OCXO. One does not need to consider the ageing rate before thermal equilibrium, because it only takes 3 to 10 minutes for an OCXO and a few seconds for other oscillators.

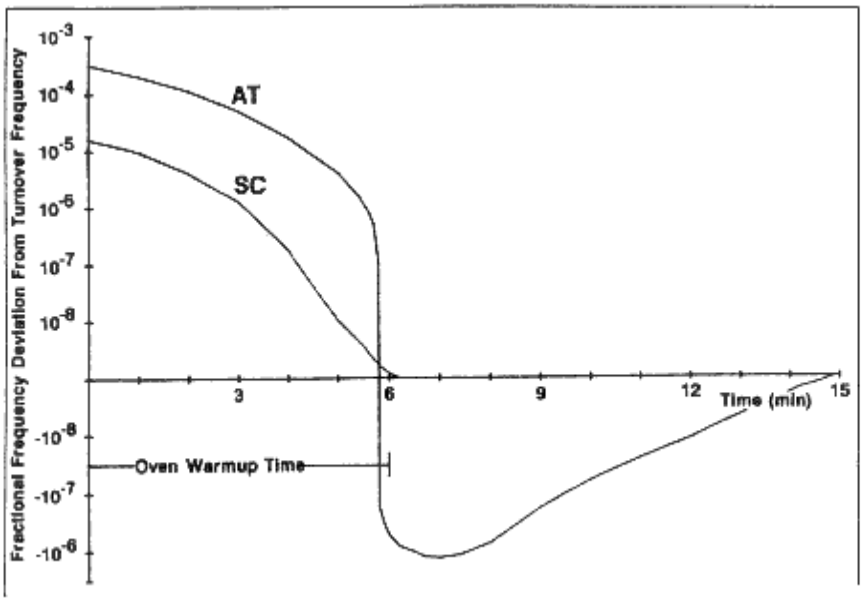

Figure 2.4 Warm-up Characteristics and Thermal Gradient Effects of AT-cut and SC-cut Crystal Oscillators (OCXOs) [7]

2) Pressure release effect. This effect is a function of the heat process above, and continues from 3 days to 3 months [6].

3) The increase or decrease of the crystal polar plates mass, which is caused by gas absorption or decomposition, continues for several weeks to several years [6].

4) Crystal structure change caused by a defective crystal lattice, which is a long-term effect.

In low-frequency quartz crystal resonators, when the vibration mode is face-shear, the ageing rate is the lowest. The ageing rate is higher in the case of bending vibration and extension vibration results in the highest ageing rate. When the vibration mode is the same, a lower frequency and a bigger polar plate crystal experience a lower ageing rate. Ageing effects can be divided into two time periods, the prior period and the later period. The prior period ageing (for 1 to 2 months) has a higher ageing rate and this ageing rate can reach up to 1×10^{-7} /month (i.e., the frequency accuracy changes by 1×10^{-7} per month) to 1×10^{-8} /month. As for the later period, when a crystal has been operational for 1~2 months, the ageing rate reduces to $(1\sim3) \times 10^{-9}$ /month to $(1\sim3) \times 10^{-10}$/month.

2.1.1.3 Retrace

When power is removed from an oscillator for several hours, and then re-applied again, the frequency of this oscillator stabilizes at a slightly different value. This frequency variation error is called retrace error. It usually occurs for twenty-four or more hours off-time followed by a warm-up time which is enough to reach thermal equilibrium.

Retrace errors reduce after warming. The shape of the error curve is as follows: the crystal walks back down its ageing curve when cold and then moves toward the prior drift curve when activated. If the resonator is in its prior period, the retrace error is added to the ageing drift, while with later period resonators the frequency looks for a new level characteristic for alternating operation. Usually, retrace errors show less spread with SC cut than with AT cut resonators. By careful selection of crystals, oscillators can decrease the influence from the retrace effect which is as close as a few parts in 10^{10} [8]. Retrace is one of the factors that affect the frequency accuracy of OCXO. As for TCXO or other oscillators, retrace is usually not considered a factor that significantly affects frequency accuracy [4].

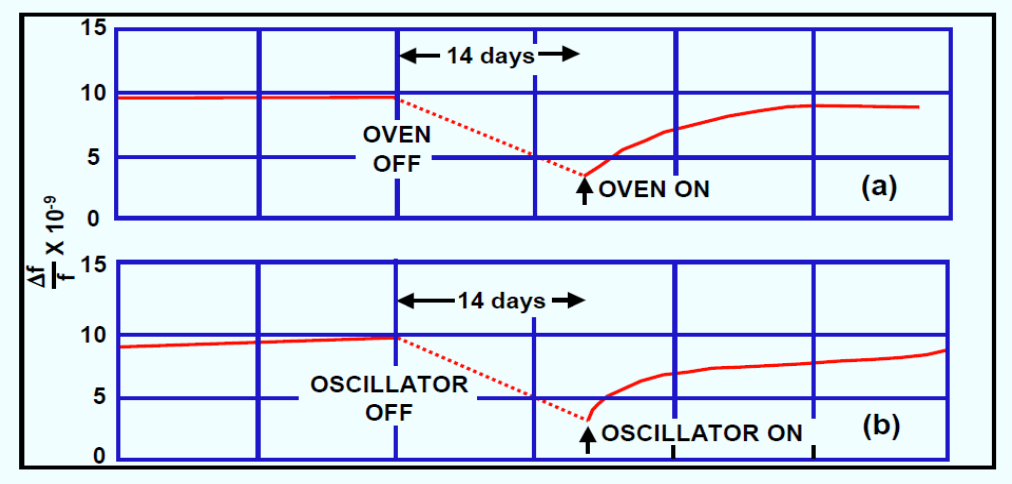

Figure 2.5 OCXO Retrace [4]

Figure 2.5 shows how OCXO retrace influences oscillator frequency accuracy. The X axis represents time and the Y axis represents frequency accuracy. In (a), the oscillator was kept on continuously while the oven was cycled off and on. In (b), the oven was kept

on continuously while the oscillator was cycled off and on.

2.1.1.4 Other Factors

Besides the factors mentioned above, other factors which affect the frequency accuracy of oscillators include Drive Level, Thermal Hysteresis, Frequency Pushing and Pulling, Tuning port reference voltage drift, etc. Other factors which affect the frequency stability of oscillators include Tuning port noise, Reference noise, Power supply noise, Vibration-induced noise, etc [9][10].

2.1.2 Factors Comparison

Different factors affecting accuracy and stability of crystal oscillators have different weights depending on the operating conditions of the oscillator. Temperature and ageing drift are the most important factors which affect the accuracy of oscillators. In the case that the thermal environment is stable, the ageing-induced frequency error may dominate the frequency behavior of the oscillator. Alternatively if the thermal environment is undergoing variations in a time frame that is short in comparison with the time required for the oscillator to drift significantly with respect to the ageing rate of the crystal, then the temperature-dependent frequency accuracy of the oscillator dominates the temporal accuracy of the clock. Ranking of other factors is highly dependent on the working environment of the oscillator and as such must be done on a case-by-case basis.

Various kinds of noises are factors which affect the stability of oscillators. There are no documents to compare which noise is the dominant factor, and this is still based on the specific application.

For example, one can consider an OCXO used in a wireless base station which references the GPS signal to keep its frequency accuracy. When the GPS reference signal is lost, the OCXO enters the “holdover” state and its accuracy drifts. Under normal operation of the base transceiver station, the maximal cumulative time error of the OCXO in a period of time (such as 24 hours) needs be limited and the OCXO accuracy needs to be enhanced by voltage control circuitry. In this situation, one should focus on the factors which affect the frequency accuracy of OCXO because the short term frequency stability does not contribute to the cumulative time error. Temperature and ageing are the two dominant factors. If the OCXO does not need to be frequently turned on and off, the retrace effect is minor.

2.1.3 Parameters of Quartz Crystal Resonators

As the most important component of the crystal oscillator, quartz crystal resonators have many technical parameters. Table 2.2 gives the characteristic parameters of a typical 5 MHz precise quartz crystal resonator.

nominal frequency	5MHz
crystal frequency difference	$-5\sim-13$Hz
zero temperature coefficient point	55℃ ± 5℃
frequency temperature coefficient	$(1\sim5)\times10^{-8}$/℃
quality factor (Q-factor)	$\geqslant 2.2\times10^{6}$
dynamic capacity C_q	1×10^{-4}pF
dynamic inductance L_q	8.5 H
dynamic resistance R_q	110～130Ω
static capacity C_0	～4pF
overtone order	**5 order overtone**
size	$\phi19\times45$ mm^2

Table 2.2 Parameters of a 5 MHz Crystal Resonator [6]

Here, the nominal frequency is the frequency that the quartz crystal resonator is designed to work at. The zero temperature coefficient point (also called the turnover point) is the temperature at which the frequency-temperature coefficient reaches zero. The crystal frequency difference means the difference between working frequency and crystal series resonance frequency when the resonator is working around the zero temperature coefficient point. The frequency temperature coefficient is the resonator frequency accuracy for each Celsius degree variation, when the crystal works around the zero temperature coefficient point (e.g. 50 degree to 60 degree here). The quality factor (Q-factor) is the ratio of the frequency at which the resonator works and the rate at which it dissipates its energy. A higher Q-factor indicates a lower rate of energy dissipation relative to the oscillation frequency, so the oscillations die out more slowly. Dynamic

capacity, dynamic inductance, and dynamic resistance are equivalent capacity, equivalent inductance, and equivalent resistance of the resonators when resonators are working, which are useful for circuit analysis. Static capacity is the capacity value of the resonator when the resonator is not working, which is used in energy storage analysis of resonators. The overtone order means the resonator works in overtone mode and the times of fundamental mode frequency.

Most highly stable crystal oscillators use a thermostatic control oven to guarantee high stability and accuracy. In a thermostatic control oven, the temperature is tuned to the zero temperature coefficient point. Table 2.3 shows the typical data for an MtronPTI's XO5120 as an example to show the frequency accuracy over temperature achievable by using OCXO technology. Figure 2.6 shows a block diagram of OCXO circuit.

Optional Temperature Ranges and Frequency Stabilities (F/T)		
OTR °C	SC-Cut	AT-Cut
0 to +50	$\pm 2 \times 10^{-9}$	$\pm 2 \times 10^{-8}$
0 to +70	$\pm 2 \times 10^{-9}$	$\pm 2 \times 10^{-8}$
-10 to +70	$\pm 3 \times 10^{-9}$	$\pm 2 \times 10^{-8}$
-30 to +70	$\pm 3 \times 10^{-9}$	$\pm 3 \times 10^{-8}$
-40 to +70	$\pm 3 \times 10^{-9}$	$\pm 3 \times 10^{-8}$
-40 to +85	$\pm 3 \times 10^{-9}$	$\pm 4 \times 10^{-8}$

Table 2.3 Ranges and Their Frequency Stability of MtronPTI's XO5120 [1]

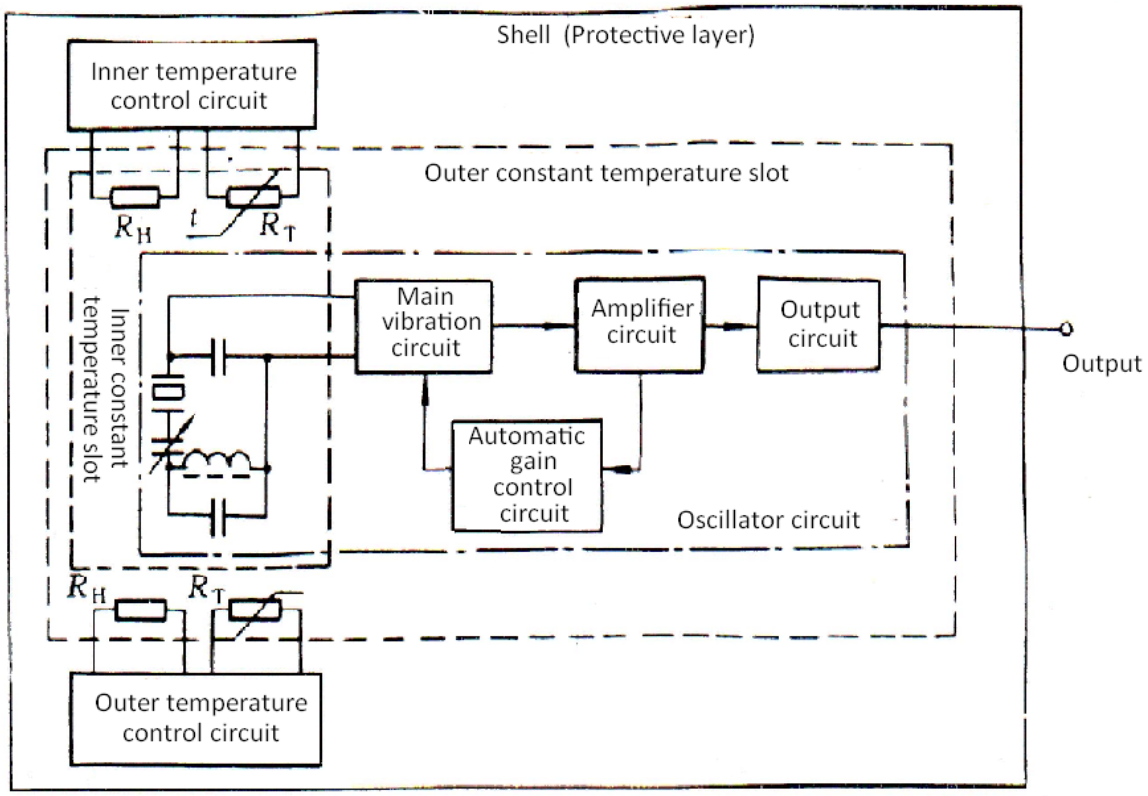

Figure 2.6 Block Diagram of OCXO Circuit [6]

2.2 Main System Identification Algorithms and Potter's Square Root Algorithm

Building mathematic models from observed input and output data is a basic factor in science. Many mathematical algorithms have been developed in the control field for different application areas. In this chapter, some classic system identification algorithms used in the thesis are reviewed. They include the Recursive Least Squares method, the Kalman Filter method, and the Recursive Prediction Error method. Potter's Square Root algorithm is also introduced. It serves as the auxiliary algorithm which can guarantee the

normal operation of the system identification algorithm.

2.2.1 The Recursive Least Squares (RLS) Method

Consider a linear difference equation model below:

$$y(t) + a_1 y(t-1) + \cdots + a_n y(t-n) = b_1 u(t-1) + \cdots + b_m u(t-m) + v(t). \quad (2.1)$$

Here, $\{y(t)\}$ is the output signal, and $\{u(t)\}$ is the input signal. $v(t)$ is the disturbance signal, and usually it follows Gaussian white noise distribution. The model (2.1) expresses the dynamic relation between input and output signals. It can be simply rewritten as

$$y(t) = \theta^T \varphi(t) + v(t). \quad (2.2)$$

Here,

$$\theta^T = (a_1 \ \ldots a_n \ \ b_1 \ldots b_m). \quad (2.3)$$

$$\varphi(t)^T = \big(-y(t-1) \ \ldots - y(t-n) \ \ u(t-1) \ \ldots \ \ u(t-m)\big). \quad (2.4)$$

The observed variable $y(t)$ can be expressed as an unknown linear combination of the components of the observed vector $\varphi(t)$ plus noise. The objective of the system identification algorithms is to identify the parameter vector θ.

For estimating the parameter vector θ, the ordinary method is to choose the estimate by minimizing the difference $y(t) - \theta^T \varphi(t)$. One can write the cost function in Equation (2.5):

$$V_N(\theta) = \frac{1}{N} \sum_{t=1}^{N} [y(t) - \theta^T \varphi(t)]^2 \tag{2.5}$$

Then one can minimize $V_N(\theta)$ with respect to θ. $V_N(\theta)$ is quadratic in θ, so it can be minimized analytically by setting $\frac{\partial V_N(\theta)}{\partial \theta} = 0$. The derived equation of the estimate of θ is written in Equation (2.6):

$$\frac{\partial V_N(\theta)}{\partial \theta} = \frac{1}{N} \sum_{t=1}^{N} (-2)\big(y(t) - \theta^T \varphi(t)\big)\varphi^T(t) = 0$$

$$\Rightarrow \quad \sum_{t=1}^{N} \varphi(t)\varphi^T(t)\theta = \sum_{t=1}^{N} \varphi(t) y(t)$$

$$\Rightarrow \quad \hat{\theta} = \left[\sum_{t=1}^{N} \varphi(t)\, \varphi^T(t)\right]^{-1} \sum_{t=1}^{N} \varphi(t) y(t) \tag{2.6}$$

Equation (2.6) is the Batch Least Squares (BLS) method. The BLS method uses all past input and output signals to estimate the parameter vector θ. Therefore, it needs a large memory and a significant amount of computation. From Equation (2.6), one can see that a matrix inversion computation is necessary. The BLS method is often used in offline system parameter estimation.

Equation (2.6) can be written in a recursive way for online system identification

purposes. According to [7], the term $\hat{\theta}(t)$ denotes the estimate for θ at time *t*. Then one can write the derived equation of $\hat{\theta}(t)$ recursively.

$$\hat{\theta}(t) = \hat{\theta}(t-1) + L(t)\left[y(t) - \hat{\theta}^T(t-1)\varphi(t)\right], \tag{2.7a}$$

$$L(t) = \frac{P(t-1)\varphi(t)}{1+\varphi^T(t)P(t-1)\varphi(t)}. \tag{2.7b}$$

$$P(t) = P(t-1) - \frac{P(t-1)\varphi(t)\varphi^T(t)P(t-1)}{1+\varphi^T(t)P(t-1)\varphi(t)} \tag{2.7c}$$

Equation (2.7) is the Recursive Least Squares estimate. The term $L(t)$ is the gain vector. The term $P(t)$ represents the covariance matrix of $\hat{\theta}(t)$. The expected value of $\hat{\theta}(t)$ is θ. According to [7], $\hat{\theta}(t)$ converges to θ when the training time approaches infinity. Generally, the initial value of $P(t)$ is $P(0) = C \cdot I$, where C is a large constant and I is the identity matrix. The initial value of $\hat{\theta}(t)$ is $\hat{\theta}(0) = 0$.

In the recursive calculations, for storing only finite information of $\hat{\theta}(k)$ for $k < t$, the forgetting factor λ is introduced, which is a number less than 1 but close to 1 such as 0.99999. According to [11], Equation (2.7) can be rewritten as:

$$\hat{\theta}(t) = \hat{\theta}(t-1) + L(t)\left[y(t) - \hat{\theta}^T(t-1)\varphi(t)\right], \tag{2.8a}$$

$$L(t) = \frac{P(t-1)\varphi(t)}{\lambda+\varphi^T(t)P(t-1)\varphi(t)} \tag{2.8b}$$

$$P(t) = \frac{1}{\lambda}\left[P(t-1) - \frac{P(t-1)\varphi(t)\varphi^T(t)P(t-1)}{\lambda+\varphi^T(t)P(t-1)\varphi(t)}\right] \tag{2.8c}$$

Equation (2.8) is the basic equation used in the thesis.

2.2.2 Potter's Square Root Algorithm

In Equation (2.8), $P(t)$ is an important term, which determines the distribution of $\hat{\theta}(t)$ at time t. For correctly updating $P(t)$ and $\hat{\theta}(t)$, $P(t)$ should always be positive definite symmetric. However, due to numerical limitations, this property of $P(t)$ sometimes cannot be guaranteed. Therefore, some algorithms have been developed to solve this problem. Here, Potter's Square Root algorithm is introduced. This algorithm guarantees that the $P(t)$ matrix remains positive definite symmetric [7].

When $P(t)$ is positive definite symmetric, it can be decomposed as

$$P(t) = Q(t)Q^T(t) \tag{2.9}$$

Here $Q(t)$ is a nonsingular matrix. Potter's Square Root Algorithm is based on the factorization Equation (2.9). The matrix $Q(t)$is calculated in the following algorithm [7].

At the initial time $t = 0$, $Q(0)Q^T(0) = P(0)$

At time t, update $Q(t-1)$by performing steps 1-5.

1. $f(t) = Q^T(t-1)\varphi(t)$.
2. $\beta(t) = \lambda + f^T(t)f(t)$.
3. $\alpha(t) = 1/[\beta(t) + \sqrt{\beta(t)\lambda(t)}]$
4. $\bar{L}(t) = Q(t-1)f(t)$.

5. $Q(t) = [Q(t-1) - \alpha(t)\bar{L}(t)f^T(t)]/\sqrt{\lambda(t)}$ (2.10)

The vector $\bar{L}(t)$ is the normalized form of the gain vector $L(t)$. The relation between them is:

$$L(t) = \frac{\bar{L}(t)}{\beta(t)}. \quad (2.11)$$

$\hat{\theta}(t)$ can be updated by Equation (2.12).

$$\hat{\theta}(t) = \hat{\theta}(t-1) + \bar{L}(t)\left[\frac{\varepsilon(t)}{\beta(t)}\right] \quad (2.12)$$

Here, $\varepsilon(t)$ is the update residual, and it is computed in Equation (2.13).

$$\varepsilon(t) = y(t) - \hat{\theta}^T(t-1)\varphi(t) \quad (2.13)$$

Then $P(t)$ can be calculated by Equation (2.9).

2.2.3 The Kalman Filter Method

The Kalman Filter method is widely used in state estimation. It can also be used in

parameter estimation problems. The general Kalman Filter equations for estimating the system state are:

$$x(t+1) = F(t)x(t) + w(t) \tag{2.14a}$$

$$y(t) = H(t)x(t) + v(t) \tag{2.14b}$$

The term $x(t)$ is the system state, and $y(t)$ is the system output. The term $F(t)$ is the state transition matrix, and $H(t)$ relates the system state to the system output. The term $w(t)$ is the process noise and $v(t)$ is the measurement noise [12]. From Equation (2.2), one can write the linear regression equation:

$$\hat{y}(t|\theta) = \varphi^T(t)\theta \tag{2.15}$$

For casting Equation (2.15) into the Kalman Filter method Equation (2.14), one can modify Equation (2.14) to Equation (2.16):

$$\theta(t+1) = \theta(t) + w(t) \tag{2.16a}$$

$$y(t) = \varphi^T(t)\theta(t) + v(t) \tag{2.16b}$$

One can assume:

$$R_1(t) = Ew(t)w^T(t) \tag{2.17}$$

$$R_2(t) = Ev(t)v^T(t) \tag{2.18}$$

Here, both $w(t)$ and $v(t)$ should be independent Gaussian white noise.

One can get Equation (2.16) by setting $F(t) = I, H(t) = \varphi^T(t)$ and $x(t) = \theta(t)$ in Equation (2.14).

Then the recursive parameter estimation equation of the Kalman Filter form can be written as:

$$\hat{\theta}(t) = \hat{\theta}(t-1) + L(t)\left[y(t) - \hat{\theta}^T(t-1)\varphi(t)\right], \tag{2.19a}$$

$$L(t) = \frac{P(t-1)\varphi(t)}{\lambda + R_2(t) + \varphi^T(t)P(t-1)\varphi(t)} \tag{2.19b}$$

$$P(t) = \frac{1}{\lambda}\left[P(t-1) - \frac{P(t-1)\varphi(t)\varphi^T(t)P(t-1)}{\lambda + R_2(t) + \varphi^T(t)P(t-1)\varphi(t)} + R_1(t)\right] \tag{2.19c}$$

When $R_1(t) \neq 0$ in (2.19c), the algorithm is tracking a time-varying parameter system. When $R_1(t) = 0$, the algorithm is tracking a time-invariant system.

2.2.4 The ARMAX Model and the Recursive Prediction Error Method

One can modify the basic linear difference Equation (2.1) to Equation (2.20).

$$A(q^{-1})y(t) = B(q^{-1})u(t) + C(q^{-1})e(t) \tag{2.20}$$

Here, q^{-1} is the backward shift operator:

$$q^{-1}y(t) = y(t-1) \tag{2.21}$$

Therefore,

$$A(q^{-1}) = 1 + a_1 q^{-1} + \cdots + a_n q^{-n} \tag{2.22}$$

$$B(q^{-1}) = b_1 q^{-1} + b_2 q^{-2} + \cdots + b_m q^{-m} \tag{2.23}$$

$$C(q^{-1}) = 1 + c_1 q^{-1} + \cdots + c_r q^{-r} \tag{2.24}$$

In Equation (2.20), the disturbance term $e(t)$ is also modeled. This is known as the

ARMAX model. One should notice that the general RLS method cannot be simply applied to the ARMAX model if the precise covariance matrix $P(t)$ is needed, because the disturbance term $C(q^{-1})e(t)$ is not independent on the last disturbance term $C(q^{-1})e(t-1)$.

For solving the ARMAX model, the Recursive Prediction Error Method (RPEM) should be applied. A simple first-order ARMAX model can be used to demonstrate how RPEM works.

Consider the ARMAX model in Equation (2.25):

$$y(t) = au(t) + e(t) + be(t-1). \tag{2.25}$$

Here, $e(t)$ is a sequence of independent white noise. The parameter is $\theta^T = (a \quad b)$. RLS method or Kalman Filter method cannot be used directly here because $e(\mathrm{t}-1)$ is not known.

The natural prediction of $y(t)$ is:

$$\hat{y}(t|\theta) = au(t) + b\hat{e}(t-1) \tag{2.26}$$

$\hat{e}(t-1)$ is calculated recursively with:

$$\hat{e}(s) = y(s) - au(s) - b\hat{e}(s-1) \tag{2.27}$$

One can evaluate the prediction error

$$\varepsilon(t|\theta) = y(t) - \hat{y}(t|\theta) \tag{2.28}$$

$\psi(t,\theta)$ is introduced here, which is the gradient of $-\varepsilon(t|\theta)$ with respect to θ.

From [7], one can get:

$$\psi(t,\theta) + b\psi(t-1,\theta) = (u(t-1) \;\; \varepsilon(t-1,\theta)) \tag{2.29}$$

Then the vector $\varphi(t)$ is introduced in Equation (2.30).

$$\varphi(t) = (u(t-1) \quad \varepsilon(t-1))^T \tag{2.30}$$

Rewrite (2.28) for $\varepsilon(t|\theta)$ as:

$$\varepsilon(t) = y(t) - \hat{\theta}^T(t-1)\varphi(t). \tag{2.31}$$

Rewrite (2.29) as:

$$\psi(t) = -\hat{b}(t-1)\psi(t-1) + \varphi(t) \tag{2.32}$$

Equation (2.32) is a typical approximation of the gradient.

The estimated parameter vector is:

$$\hat{\theta}(t-1) = (\hat{a}(t-1) \quad \hat{b}(t-1))^T \tag{2.33}$$

The Recursive Prediction Error Method (RPEM) can be given as

$$\hat{\theta}(t) = \hat{\theta}(t-1) + P(t)\psi(t)\varepsilon(t), \tag{2.34a}$$

$$P(t) = P(t-1) - \frac{P(t-1)\psi(t)\psi^T(t)P(t-1)}{1+\psi^T(t)P(t-1)\psi(t)} \tag{2.34b}$$

From Equation (2.34) and the definition of $\varepsilon(t)$ and $\psi(t)$, the algorithm for an ARMAX model of arbitrary order can be constructed.

The detailed RPEM deduction process is complicated. Anyone who is interested in this algorithm can refer to [7].

Chapter 3: Problem Statement

Highly accurate and stabilized timing modules are important in many electrical systems, such as wireless communication base transceiver stations. Such timing modules are expensive, have high power consumption and a large size. Building a low cost, low power consumption and small size timing module with the same high accuracy and stability is an important research objective. Some researchers have developed adaptive control algorithms for oscillators to meet the objective. [13] developed an algorithm for performing adaptive temperature and ageing compensation of oscillators. This algorithm is based on a Recursive Least Squares method. The performance of the algorithm is presented by employing it to a TCXO, an OCXO, and a Rubidium oscillator. The algorithm improves the performance of all of these oscillators in holdover mode. However, [13] did not consider the instability problem of GPS signal and did not analyze the characteristics of correction signal created through GPS signal. [14] and [15] used the Kalman Filter method to develop algorithms for enhancing the oscillator stability in holdover mode. These algorithms compensate the ageing effect of oscillators in a long period of time. They did not compensate the temperature effect. They also did not analyze the characteristics of correction signal. [16] used the Kalman Filter method to develop an algorithm to compensate the ageing and temperature effect. [17] used the Batch Least Squares method to compensate the temperature effect. Both [16] and [17] used the same

correction signal creation model and they can meet the CDMA base station timing requirements. However, algorithms developed from [16] and [17] assume the linear stability dependencies of oscillators which may not be true in most situations. This thesis is based on [17]. The system concept and hardware structure of [17] are described in Section 3.1 in detail.

The performance bound of oscillator when the adaptive control algorithm is used is another objective. The performance bound can determine whether the adaptive algorithm can meet system requirements. Generally, the performance bound of the adaptive control algorithm is represented by the cumulative time error of the oscillator. So far, there are no papers to investigate the performance bound of oscillators using adaptive control algorithms. This thesis will try to investigate this problem.

In this chapter, a useful adaptive OCXO drift correction algorithm and related system structure based on [17] are reviewed. Then the deficiencies of this algorithm are demonstrated. In order to meet the actual application requirements, a more comprehensive adaptive method is needed.

3.1 Review of an Adaptive OCXO Drift Correction Algorithm

WiMAX and CDMA base stations all need to be time synchronous with respect to RST (radio system time). GPS (global positioning system) satellites which are equipped

with ultra-high accuracy atomic clocks generally serve as these time servers. They provide a 1 pulse per second (1 pps) signal to which the wireless base stations are time synchronous through phase lock. This state is termed locked mode. In contrast to the locked mode, when the base stations lose their external frequency reference, they enter holdover mode. In the holdover mode, the frequency accuracy of the base station is totally dependent on its timing module.

According to [3], in the locked mode, the WiMAX and CDMA base stations must have a time offset relative to RST lesser than $\pm 1\ \mu s$. In the holdover mode, relative to RST, the CDMA base station 1pps signal must not exceed $\pm 10\ \mu s$ time error over an 8 hour period. A WiMAX base station must not exceed $\pm 25\ \mu s$ time error over an 8 hour period. The system design must be based on the specification above. The time error Δt and the time duration T for which the frequency stability error is preserved are related to the stability of the oscillator $\Delta f / f_0$ through Equation (3.1) [17].

$$\frac{\Delta t}{T} = \frac{\Delta f}{f_0} \tag{3.1}$$

Applying Equation (3.1) to the CDMA time error of $\pm 10\ \mu s$ over an 8 hour period in holdover mode, one can get the maximum allowable frequency error of the oscillator of ± 0.35 ppb (parts per billion). With the purpose of meeting this accuracy requirement over a $75\ ^{\circ}C$ operation temperature, a double oven crystal oscillator (DOCXO) is applied. Due to the cost, size and power consumption limitation of the DOCXO, it is feasible to

apply an adaptive modeling of the base station timing module during the locked mode and use the resulting model to correct the oscillator frequency drift during holdover mode.

In [16] and [17], a single oven crystal oscillator (OCXO) is used and the Batch Least Squares (BLS) fit algorithm is applied to compensate for the OCXO deficiencies to meet the CDMA and WiMAX base station timing module specification in holdover mode. Table 3.1 shows the difference between DOCXO and algorithm-enabled OCXO in the base station. Some advantages of using OCXO are apparent from this table.

Component parameter	***Incumbent DOCXO***	***Algorithm Enabled OCXO***
DC supply requirement	+12V	+5V
DC power consumption warm up	9.6W	3.5W
Peak to peak frequency stability over operational temperature ppb/75^{o}C	0.4ppb	4ppb
Ageing ppb/24hours	+/-0.05ppb	+/-1ppb
Dimension (L x W x H)/mm	50 x 50 x 38	25.4 x 25.4 x 12.7
Cost in volume 10K / USD	~$250	~$50

Table 3.1 Incumbent DOCXO and Algorithm Enabled OCXO [17]

Besides the low cost of the OCXO, the smaller power consumption and small size allow the timing module to be integrated onto the base station modem card. Using a modem card results in large cost saving as the DOCXO has to be built on a standalone module.

3.1.1 Timing Module System

The detailed system structure block diagram of the base station timing module is shown in Figure 3.1.

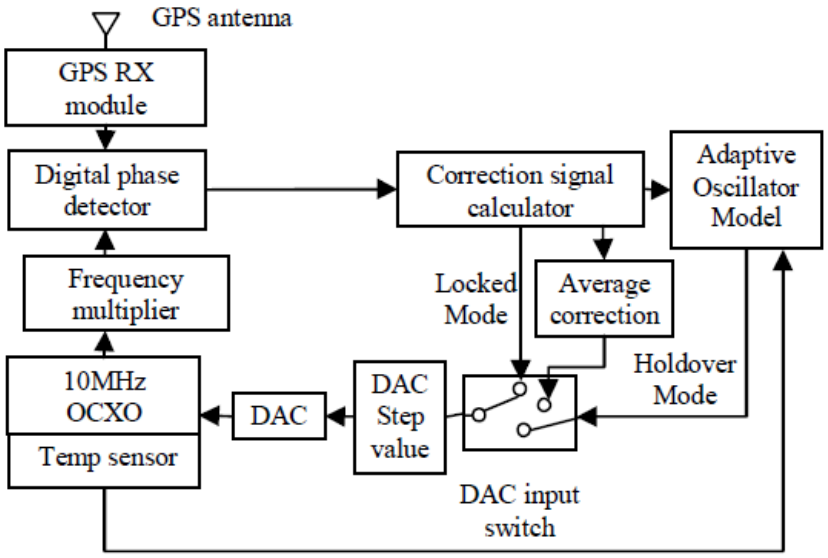

Figure 3.1 Detailed Block Diagram of the Timing Module System [17]

The GPS receiver module offers a 1pps reference signal, which is coming from GPS satellites. Because all GPS satellites are equipped with ultra-high accurate rubidium atomic clocks, this 1pps reference signal is very precise. The stability of the GPS 1pps

signal is not that high compared to its accuracy. Typically, the GPS receivers can bring in a GPS noise from 20ns to 30 ns rms (root mean square) jitter on the 1pps edge.

The whole digital control loop includes the digital phase detector, correction signal calculator, and the adaptive oscillator model. All of these functional models are resident on a Field Programmable Gate Array (FPGA), which includes a processor. A frequency source which is generated from the frequency multiplier is used to count the time interval between the rising edges of the 1pps reference signal from the GPS receiver module. A 10MHz OCXO is used to feed this frequency multiplier. The digital phase detector counts the numbers of periods of the frequency source. According to the count value, the correction signal is computed by the correction signal calculator. This correction signal then is applied to a Digital to Analog Converter (DAC) to control the 10 MHz OCXO and it is also used to feed the adaptive oscillator model which can be used when the system loses the GPS signal. A temperature sensor is used to collect the ambient temperature.

The 10MHz OCXO is the key component of the timing module, which is locked to the GPS reference signal through the control loop in the locked mode. In the holdover mode, the adaptive oscillator model creates the correct signal to the OCXO.

3.1.2 Digital Control Loop

Referring to Chapter 2, the accuracy of the OCXO is dependent on many factors. The correction signal generated by the control loop compensates for the effect of these

factors on the accuracy of the OCXO. In this section, the control loop is introduced in detail.

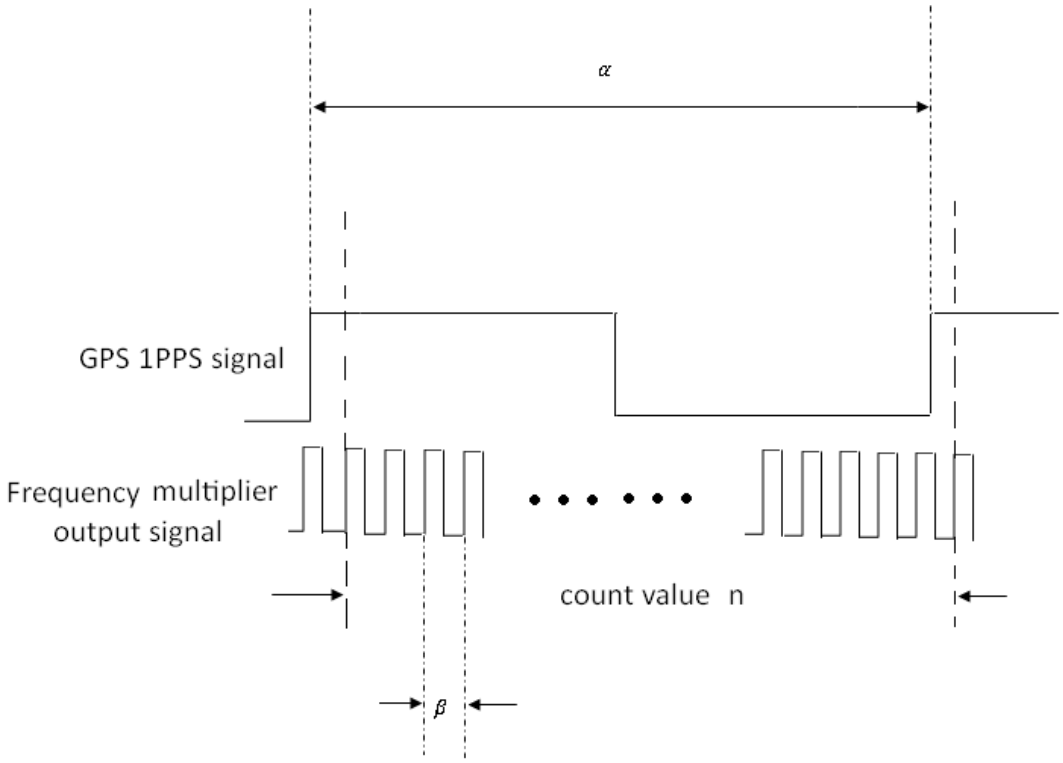

Figure 3.2 Frequency Multiplier Output Counting GPS 1pps Signal

In order to introduce the control loop, the first step is to introduce error counts. Figure 3.2 indicates how the frequency signal generated by the frequency multiplier is used to count between the rising edges of the GPS 1pps signal. For example, we assume the frequency of the frequency multiplier output is 160 MHz. Then the period of the

output is 6.25 ns. If there is no GPS noise and the oscillator does not have frequency drift, the length between two adjacent rising edges of the GPS 1pps signal should be just 1 second. The number of 160 MHz clock cycles between two adjacent rising edges of the GPS 1pps signal should be just 160,000,000. The number 160,000,000 here is the count value. However, the count value may not be 160,000,000 because there are always GPS noise and frequency drift. When the OCXO has no frequency drift, the count value is equal to the frequency of the frequency multiplier output +/- the error counts. These error counts are generated by the GPS noise. Generally, the GPS noise has a standard deviation from 20 ns to 30 ns in 1pps signal.

In Equation (3.2), the term α represents the actual time interval between the rising edges of the 1pps signal. The term β represents the digital phase detector resolution. The count value is represented by n. For example, β equals to 6.25 ns and the frequency multiplier output frequency is 160 MHz. The time interval α is 1 second plus 25 ns. The actual count value is:

$$n = \frac{\alpha}{\beta} + v = \frac{1+25\times10^{-9}}{6.25\times10^{-9}} + w = 160 \times 10^{6} + 4 + w \qquad (3.2)$$

The term w represents the inherent random error in the counting and it cannot be avoided. w is chosen from 1, -1, or 0 randomly according to different phases. Therefore, the actual error count in this equation is 3, 4, or 5. The term w does not affect the mean value of the error counts when there is no frequency drift and the mean count value

should be zero.

When the OCXO has frequency drift, this drift appears as a bias on the mean count value. A moving average filter is used by the control loop to divide the OCXO frequency drift from the GPS noise of the receiver. The error counts are multiplied by the digital phase detector resolution to produce the time error between the OCXO and the received GPS 1pps signal. All time errors are integrated to create the cumulative time error (CTE). CTE can be recursively calculated through Equation (3.3). The digital phase detector resolution is still denoted by β. The term ε is used to denote error counts. k represents how many seconds the timing module has been running for. Therefore, CTE_k represents the cumulative time error when the timing module has been running for k seconds. The term ε_k represents error counts at the *k-th* second. According to Equation (3.3), CTE_k must be multiples of β.

$$CTE_k = CTE_{k-1} + \beta * \varepsilon_k \tag{3.3}$$

The correction signal is created by combining the CTE and a moving average of the former correction signals.

$$correct_{ref} = \frac{1}{N}\sum_{t=k-N}^{k-1} C_t \tag{3.4}$$

$$C_k = correct_{ref} - CTE_k/damp \tag{3.5}$$

In Equations (3.4) and (3.5), C_t denotes the correction signal at time t and C_k denotes the correction signal at time k. The term $correct_{ref}$ which is the average value of the last N correction signals provides an equilibrium point about which the CTE acts. The term N is a large constant such as 2000. The term *damp* is a constant such as 150. It is used for suppressing the GPS receiver noise. A digital to analog converter (DAC) is used to convert the digital correction signal into an analog tuning voltage. The whole process of determining the tuning voltage is: first, the correction signal which is digital and expressed in ppb (parts per billion) is divided by the OCXO tuning sensitivity (Kvco), which is expressed in ppb/volt. Therefore, the voltage which is applied to the tuning port of the OCXO is obtained. Second, the tuning voltage is divided by the DAC resolution which is the ratio of the control voltage range to the total number of DAC steps. Then the actual number of DAC steps is obtained, which is a binary word. The calculation of the DAC steps is:

$$DAC_{steps} = fix\{\frac{correction\ \ signal}{Kvco * DAC_{resolution}}\} \tag{3.6}$$

The operator $fix(\cdot)$ truncates the arguments in the brackets toward zero. For example, $fix(2.1) = 2$ and $fix(-1.6) = -1$. $fix(\cdot)$ is used in Equation (3.6) because DAC steps must be an integer. This DAC steps value is fed into the DAC to get the real control voltage.

Figure 3.3 shows the block diagram of the digital control loop.

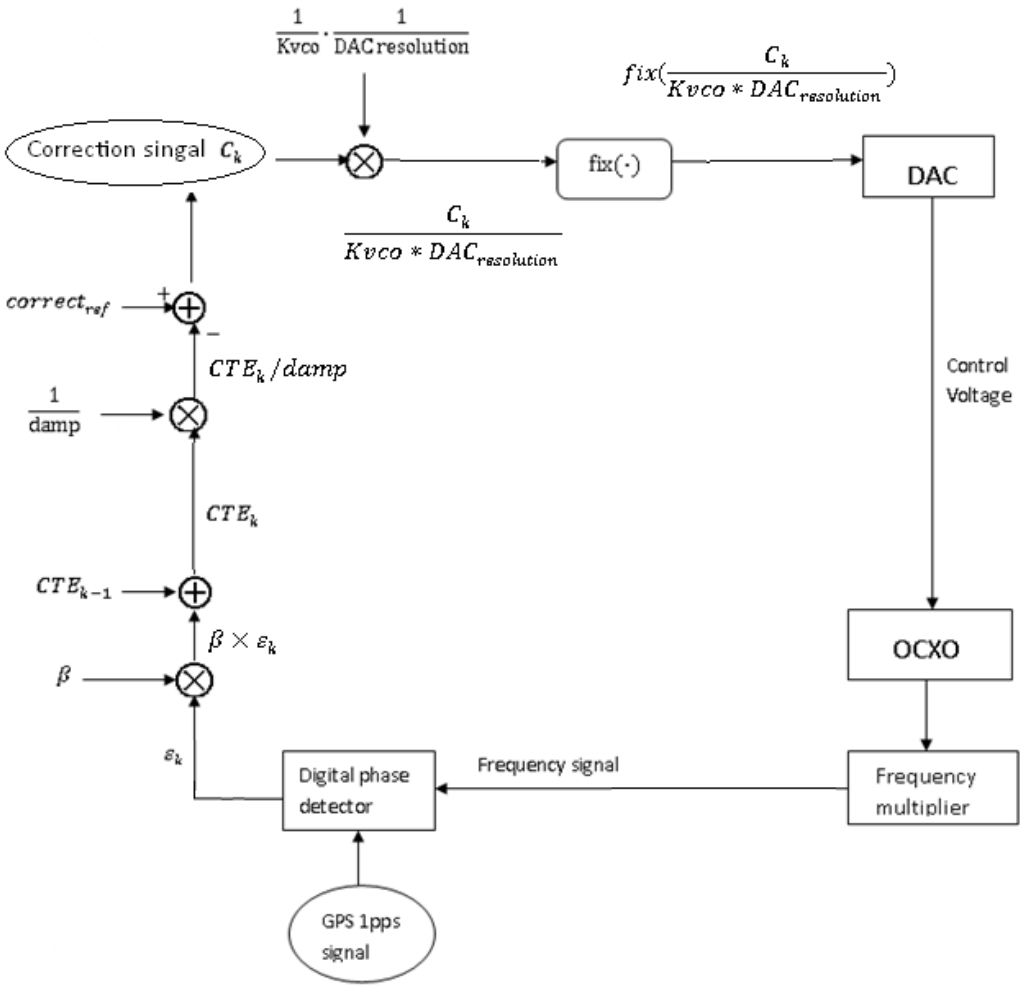

Figure 3.3 Block Diagram of the Digital Control Loop

3.1.3 The Adaptive Control Algorithm

Although there are many factors affecting the frequency accuracy of the OCXO, the two most important factors are temperature and ageing. When the OCXO is locked to the GPS satellite reference signal, we refer to this as the OCXO be in a training mode. The

temperature in the crystal oven of OCXO is limited to a small range around the turnover point. The turnover point refers to the temperature at which the frequency-temperature ratio is zero. The temperature is linear related to the frequency stability of the oscillator at the turnover point, so it can be assumed that a straight line fit is applied to the temperature and the correction signal. A Batch Least Squares (BLS) method is used to determine the coefficients of the straight line fit [17] [18] [19]. The ageing effect is not considered, so the OCXO model equation is:

$$y_i = a_2 \cdot x_i + a_1 + v_i \tag{3.7}$$

Here, y_i represents the i_{th} frequency stability reading of the OCXO.

$$y_i = \frac{\Delta f_i}{f_0} \tag{3.8}$$

The term x_i represents the i_{th} termperature sensor reading. The term a_1 represents the initial offset of the frequency stability and a_2 represents the thermal sensitivity of the crystal resonator frequency stability. The term v_i represents the GPS receiver noise.

The term r_i is used to represent the residual between the frequency stability reading and the frequency stability prediction. The term σ_i is used to represent the difference between the mean value of the y_i data set and the i_{th} data point.

$$\sigma_i = \bar{y} - y_i \tag{3.9}$$

The summation of the square of the residuals is expressed as:

$$S = \sum_i \left(\frac{r_i}{\sigma_i}\right)^2 \tag{3.10}$$

The algorithm minimizes S to obtain the optimal estimation of the coefficients a_2 and a_1.

σ_i is used here to decrease the impact of outliers on the line fit by decreasing the weighting on large residuals.

From Equation (3.7), it follows that the frequency stability prediction is:

$$\hat{y} = a_2 \cdot x_i + a_1 \tag{3.11}$$

Substituting Equation (3.11) into Equation (3.10):

$$S = \sum_i \left(\frac{y_i - (a_2 \cdot x_i + a_1)}{\sigma_i}\right)^2 \tag{3.12}$$

Equations (3.13) and (3.14) are obtained by setting the partial derivatives of S with respect to the coefficients a_1 and a_2 to zero.

$$\frac{\partial S}{\partial a_1} = \sum_i -2\left(\frac{y_i-(a_2\cdot x_i+a_1)}{\sigma_i^2}\right) = 0 \quad (3.13)$$

$$\frac{\partial S}{\partial a_2} = \sum_i -2x_i\left(\frac{y_i-(a_2\cdot x_i+a_1)}{\sigma_i^2}\right) = 0 \quad (3.14)$$

Set:

$$S_1 = \sum_i \frac{1}{\sigma_i^2} \quad (3.15)$$

$$S_x = \sum_i \frac{x_i}{\sigma_i^2} \quad (3.16)$$

$$S_y = \sum_i \frac{y_i}{\sigma_i^2} \quad (3.17)$$

$$S_{xx} = \sum_i \frac{x_i^2}{\sigma_i^2} \quad (3.18)$$

$$S_{xy} = \sum_i \frac{x_i\cdot y_i}{\sigma_i^2} \quad (3.19)$$

$$\Delta = S_1 \cdot S_{xx} - S_x^2 \quad (3.20)$$

Expanding Equations (3.13) and (3.14) results in Equations (3.21) and (3.22):

$$-2\sum_i \frac{y_i}{\sigma_i^2} + 2\cdot a_1 \sum_i \frac{1}{\sigma_i^2} + 2\cdot a_2 \sum_i \frac{x_i}{\sigma_i^2} = 0 \quad (3.21)$$

$$-2\sum_i \frac{x_i\cdot y_i}{\sigma_i^2} + 2\cdot a_1 \sum_i \frac{x_i}{\sigma_i^2} + 2\cdot a_2 \sum_i \frac{x_i^2}{\sigma_i^2} = 0 \quad (3.22)$$

The coefficients a_1 and a_2 are obtained by substituting Equations (3.15) to (3.20) into Equations (3.21) and (3.22).

$$a_1 = (S_y S_{xx} - S_{xy} S_x)/\Delta \quad (3.23)$$

$$a_2 = (S_1 S_{xy} - S_x S_y)/\Delta \tag{3.24}$$

Equations (3.9) to (3.24) for calculating a_1 and a_2 are actually the Batch Least Squares (BLS) method which is introduced in Section 2.2.1.

High precision OCXOs have an ageing rate of less than 1ppb per day, which corresponds to $1.157 \times 10^{-5}\ ppb/sec$. One can assume that the ageing effect is linear:

$$\left(\frac{\Delta f}{f_0}\right)_{ageing} = rate_{ageing} \cdot time \tag{3.25}$$

The cumulative time error according to ageing is obtained by integrating Equation (3.25) with respect to time.

$$CTE_{ageing} = \frac{1}{2} rate_{ageing} \cdot time^2 \tag{3.26}$$

If the holdover mode lasts for 8 hours and the ageing rate is exactly 1ppb per day, the cumulative time error in terms of ageing would be $4.8\ \mu s$. This time error can be tolerated by the WiMAX specification of $25\ \mu s$ and CDMA specification of $10\ \mu s$ for 8-hour holdover times. Accordingly, ageing effect is not considered in this adaptive algorithm.

Figure 3.4 illustrates the operation of the adaptive control algorithm.

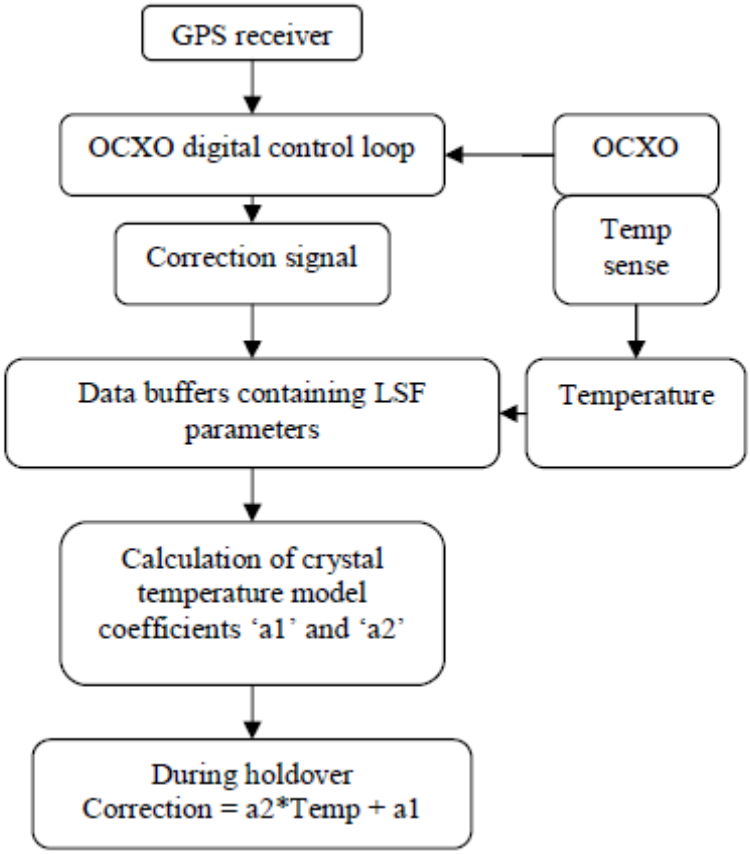

Figure 3.4 Flow Diagram for Adaptive Control Algorithm [17]

The data buffers store the values of the sums of Equations (3.15) to (3.19), which are periodically set to zero to prevent overflow errors. When a predetermined ambient temperature variation is met, one can consider that enough input samples are obtained. Then the BLS method can be considered to have converged to a sufficiently accurate level and the calculated model coefficients are considered applicable for use during holdover mode.

3.1.4 Simulation of the Adaptive Control algorithm

A software test platform which is based on Matlab is developed by [17] to confirm the correct operation of the adaptive control algorithm. An ideal OCXO model is created which contains the linear frequency dependence on temperature and linear frequency dependence on ageing. The ageing rate is set to 1ppb/day and the temperature sensitivity of the OCXO frequency stability is set to $4ppb/75℃$ respectively. The test platform switches between the locked mode and holdover mode. A second OCXO model which is not enhanced by the adaptive algorithm is run in parallel for comparing the results of the cumulative time error and showing the impact of the algorithm.

The ambient temperature variation profile is fixed and is illustrated in Figure 3.5. Figure 3.6 illustrates the simulation result of the BLS fit of the temperature model and the correction signal data readings. The solid straight line through the data points represents the BLS fit. Figure 3.7 graphs the cumulative time error for the uncorrected OCXO and the corrected OCXO respectively when the models are in the locked mode for 4 hours and then in holdover mode for 8 hours.

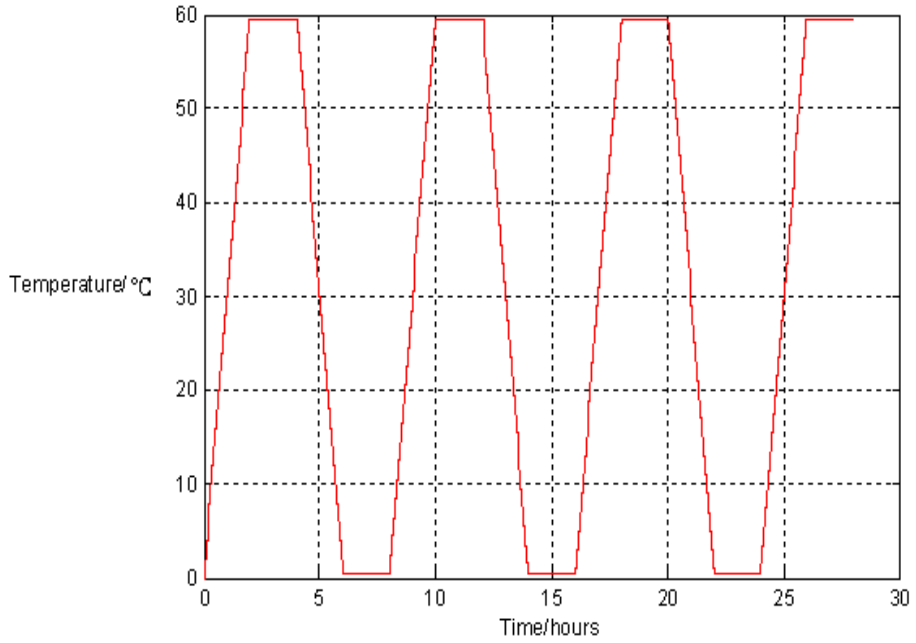

Figure 3.5 Temperature Profile

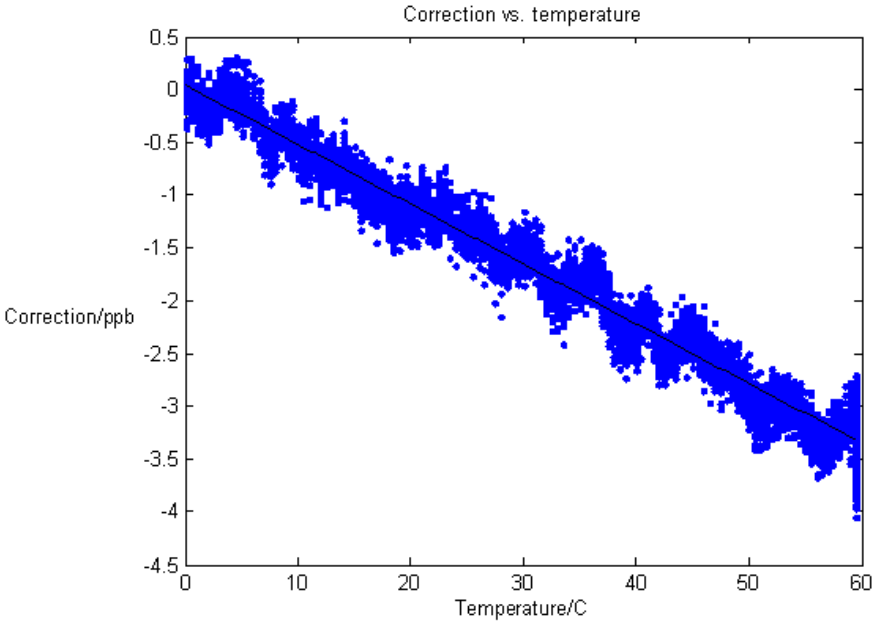

Figure 3.6 Correction Signal Data and the BLS Fit Line

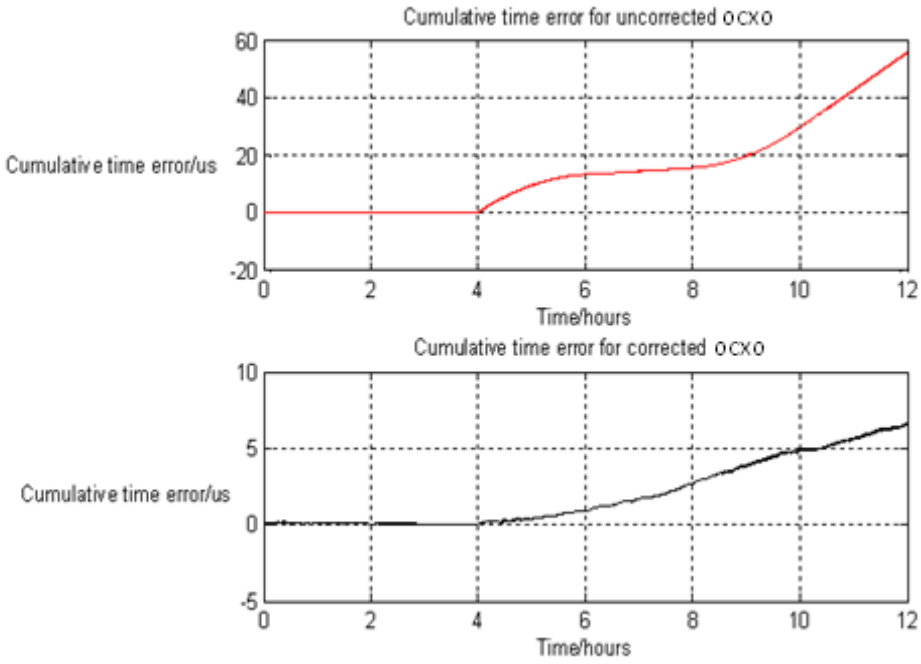

Figure 3.7 CTE for the Uncorrected and Corrected OCXO

From Figure 3.7, it can be found that the algorithm provides a 10-fold improvement in the cumulative time error for the corrected OCXO over the uncorrected OCXO. The WiMAX and CDMA specifications can be met with the corrected OCXO as the timing module when the ageing rate of the oscillator is not considered.

3.2 Deficiencies of the Adaptive Control Algorithm

The adaptive control algorithm for improving frequency accuracy of the OCXO described in the last section is very practical for the timing module of the WiMAX and

CDMA base transceiver stations. However, this control algorithm still has some deficiencies. First, the adaptive algorithm above uses the BLS method, which needs data buffers to contain the parameters. Actually, a Recursive System Identification method is more suitable for this on-line parameter estimation problem. In contrast, the BLS method is more suitable for off-line parameter estimation problems. By using a Recursive System Identification method, the required computation of the processor in the timing module is decreased significantly. Many Recursive System Identification methods are developed, and these methods are very mature, such as the Kalman Filter method and the Recursive Least Squares method. Therefore, we have more options in developing the adaptive control algorithm. According to the performance of these algorithms, the best one can be chosen.

Second, the adaptive algorithm above does not consider the ageing effect on the frequency stability of the OCXO in the timing module. In the last section, when the holdover mode lasts for 8 hours and the ageing rate of the OCXO is 1ppb/day, the cumulative time error with respect to the ageing effect is $4.8\ \mu s$. This result satisfies the specification of the timing modules of WiMAX and CDMA base stations. However, with the development of wireless communication technology, the requirement of the cumulative time error of the timing module is becoming more stringent. In addition, longer holdover periods are desirable. Therefore, the ageing effect of the OCXO should be considered in the adaptive control algorithm.

Third, a straight line fit is applied in the adaptive algorithm. The real relation

between the temperature and the frequency stability correction signal of the OCXO is not linear. In fact, it is more like a quadratic function. A linear fit only satisfies the requirement on the temperature sensitivity estimation for the OCXO. A quadratic curve fit is applied to the temperature and correction signal data for more accurate system identification.

Fourth, the upperbound of the cumulative time error of oscillator is another interesting problem for some applications. The parameters predicted cannot be one hundred percent accurate because, according to the system identification theory, the parameter estimates converge to the real coefficients when the training time approaches infinity. When the training time is finite, the distribution of parameter estimates can determine the error range of the predicted values of the parameters. Applying this error range to the timing module, the upperbound of the cumulative time error of the OCXO enhanced by the adaptive control algorithm is determined. Many engineering applications can apply this upperbound of the cumulative time error to determine if this adaptive algorithm is suitable, given their specifications.

The problems stated in this section are the main research tasks and solved in the subsequent chapters.

Chapter 4: Training Algorithms for a Simple Model

In this chapter, the Recursive Least Squares (RLS) method is evaluated and compared with the BLS method. Then the Kalman Filter method is evaluated and compared with the RLS method.

4.1 Adaptive Control Algorithm with RLS Method and BLS Method

In the last chapter, a BLS method is used to estimate the temperature sensitivity parameter to improve the frequency stability of an OCXO. In this section, the Recursive Least Squares method is used to develop the adaptive algorithm. For simplicity, a linear fit is applied to the temperature and correction signal data. The ageing effect is ignored and the initial frequency offset is zero. The OCXO model equation is of the general form:

$$y(t) = \theta \cdot x(t) + v(t) \tag{4.1}$$

Where the term $y(t)$ is the correction signal defined in Equation (3.8), θ is the temperature sensitivity of the OCXO. The term $x(t)$ represents the ambient temperature around the OCXO and the measurement noise in the system is $v(t)$. The task is to

estimate the value of θ.

First, the performance of the BLS method is compared with the RLS method. The performance of system identification methods is determined by the cumulative time error of the OCXO. The cumulative time error is calculated through simulation and Matlab is used as the simulation platform.

The BLS method uses Equation (2.6) to estimate the parameter and to calculate the CTE. All past input and output data are needed to operate Equation (2.6) and no initial values are needed.

The RLS method uses Equation (2.7) to estimate the parameter and to calculate the CTE. The initial values of $\hat{\theta}(t)$ and $P(t)$ are needed. Referring to Section 2.2.1, we initialize $\hat{\theta}(0) = 0$ and $P(0) = C \cdot I$. Here, C is a large constant and I is the identity matrix. In this thesis, C is set to 900. The estimated parameter $\hat{\theta}(t)$ is a scalar in this simple model, and $P(t)$ is proportional to the covariance matrix of $\hat{\theta}(t)$. Hence, $P(t)$ is a scalar and it equals to 900 in this section. The forgetting factor λ is set to 0.99999. In each step of the recursive calculation, one only needs to update $P(t)$ and $\hat{\theta}(t)$, and the total computation load is much lower than for the BLS method.

In order to update $P(t)$ and $\hat{\theta}(t)$ correctly in the RLS method, $P(t)$ needs to be positive definite symmetric in each step. This property of $P(t)$ may be corrupted in the recursive computations. We applied Potter's Square Root Algorithm on the Recursive Least Squares method to overcome this problem, as described in Section 2.2.2. In this algorithm, by setting $Q(0)Q^T(0) = P(0)$ and only updating $Q(t)$ in each step, we obtain

$P(t)$ via Equation (4.2).

$$P(t) = Q(t)Q^T(t) \tag{4.2}$$

By using this method, the matrix $P(t)$ is guaranteed to be positive definite symmetric.

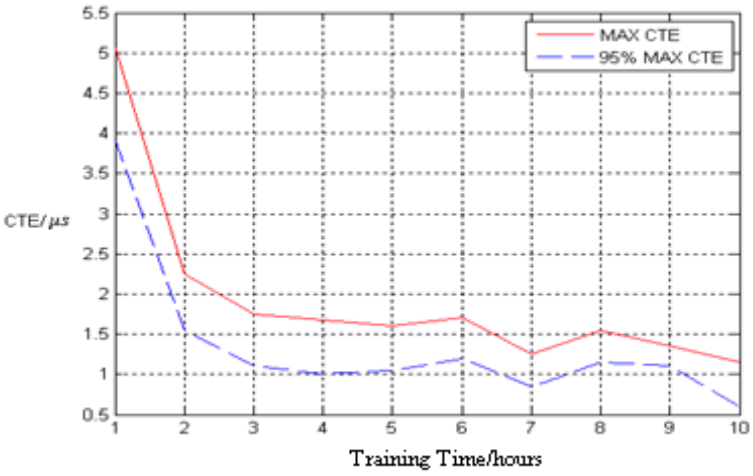

Figure 4.1 Maximum CTE and Fifth Maximum CTE for 8 Hours Holdover with BLS Method

The performance of the BLS method is shown in Figure 4.1 and the performance of RLS is shown in Figure 4.2.

In Figure 4.1 and 4.2, the holdover mode time is 8 hours and the training time is from 1 hour to 10 hours. The temperature profile which represents the input data is presented in Figure 3.5. For each training time, the simulations are run 100 times. The maximum cumulative time error and the fifth maximum cumulative time error are recorded. The X-axis shows how many hours the algorithm is trained. The Y-axis

represents the cumulative time error and the unit is μs. The solid line represents the maximum CTE in 100 simulations according to the different training time.

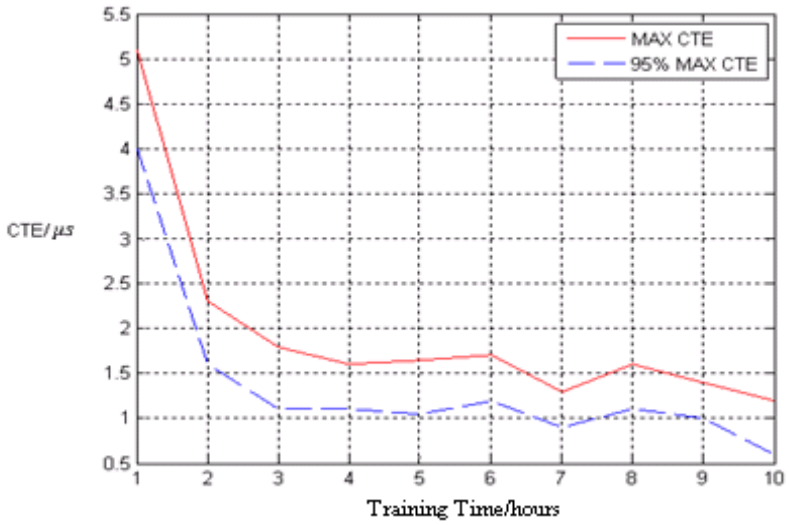

Figure 4.2 Maximum CTE and Fifth Maximum CTE for 8 Hours Holdover with RLS Method

For example, when the X-axis coordinate value is 5, the Y-axis value on the solid line is approximately 1.7 in Figure 4.2, which means that when the algorithm is trained 5 hours and the holdover mode lasts for 8 hours, the maximum CTE with the RLS method in 100 simulations is approximately 1.7 microseconds. The maximum CTE in 100 simulations can approximately represent the upperbound of CTE. The dashed line represents the fifth maximum CTE in 100 simulations according to the different training time. For example, when the X-axis coordinate value is 6, the Y-axis coordinate value on the dashed line in Figure 4.1 is approximately 1.2, which means when the algorithm is

trained 6 hours, the fifth maximum CTE with the BLS method in 100 simulations is approximately 1.2 microseconds. The fifth maximum CTE in 100 simulations can approximately represent the 95% probability CTE upperbound because there are 100 simulations. The 95% probability CTE upperbound corresponds to the 95% confidence interval of the parameter estimate. The 95% confidence interval is an important parameter investigated in subsequent chapters. We can see that in Figure 4.1 and Figure 4.2, there is an obvious drop when training time is 7 hours. In theory, the maximum CTE in 100 simulations should be monotone decreasing along with training time increasing. However, when training time is long enough, such as more than 4 hours in this example, the simulation results of maximum CTE are not necessary to be monotone decreasing because the CTE is low enough. These results are totally normal. Because the same simulated input and output data are used for training BLS and RLS methods and they produce similar parameters, Figure 4.1 and Figure 4.2 have similar maximum CTE and 95% maximum CTE plots. This situation also happens when Kalman Filter method is used below and the similar plot is produced because the same input and output data are still used for training Kalman Filter method.

The performance of BLS and RLS are determined by their CTE upperbound. One can see the performances of both methods are almost identical. However, the BLS method needs all past input and output data to estimate the parameters and needs the matrix inversion computation which is not needed in the RLS method Therefore, a large memory is needed and the computation load of the BLS method is also heavier than the

RLS method. In the RLS method, the covariance matrix of $\hat{\theta}(t)$ can be calculated through $P(t)$. This covariance matrix is important to calculate the analytical CTE upperbound of the OCXO. The BLS method cannot calculate the covariance matrix of the parameter estimate. Therefore, in this thesis we use the RLS method to replace the BLS method because of the advantages of the RLS method above.

4.2 Adaptive Control Algorithm with Kalman Filter Method

Besides the Recursive Least Squares method, the Kalman Filter method is another efficient recursive system identification method. Referring to Section 2.2.3, in addition to initializing $\hat{\theta}(0) = 0, P(0) = 900$, and $\lambda = 0.99999$, which are the same as for the Recursive Least Squares method, one also needs to know the covariance matrix $R_1(t)$ of the process noise $w(t)$, and the covariance matrix $R_2(t)$ of the measurement noise $v(t)$. For the convenience of the readers, the Kalman Filter Equations (2.16) to (2.18) are rewritten as Equations (4.3) to (4.5).

$$\boldsymbol{\theta(t+1) = \theta(t) + w(t)} \tag{4.3a}$$

$$\boldsymbol{y(t) = \varphi^T(t)\theta(t) + v(t)} \tag{4.3b}$$

We assume:

$$\boldsymbol{R_1(t) = Ew(t)w^T(t)} \tag{4.4}$$

$$\boldsymbol{R_2(t) = Ev(t)v^T(t)} \tag{4.5}$$

If $R_1(t)$ and $R_2(t)$ are known exactly, the Kalman Filter method is the best system identification method [20]. Unfortunately, these quantities are seldom known a priori. They are always the design parameters in the estimation algorithm. One can choose $R_1(t) \geq 0$ and $R_2(t) > 0$ in order to get the desired properties of the filter [21]. In this simple application, given that the parameters are time invariant, we set the term $R_1(t)$ to zero and set the term $R_2(t)$ to 4. The reason is that the Kalman Filter does not need an accurate model as long as the gain vector $L(t)$ in Equation (2.19) keeps away from zero [20]. A setting of $R_1(t) \geq 0$ and $R_2(t) > 0$ can guarantee this requirement.

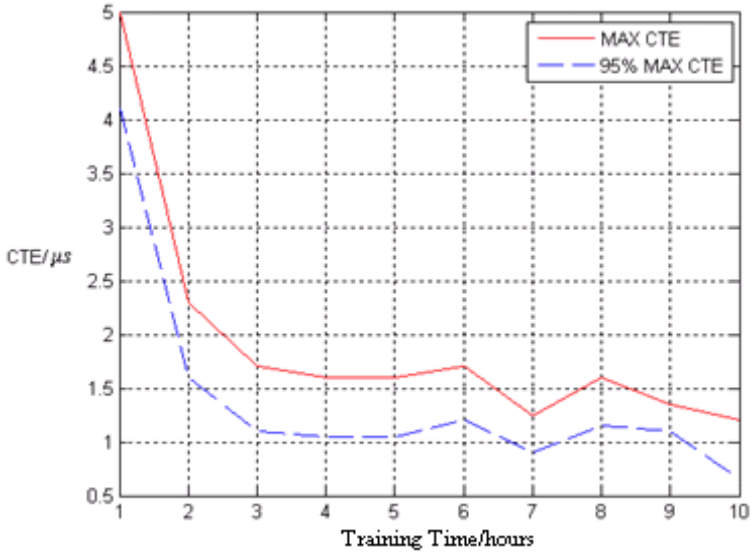

Figure 4.3 Maximum CTE and Fifth Maximum CTE for 8 Hours Holdover with Kalman Filter Method

By applying the initial values and the covariance matrix of the noise into Equation (2.19), the parameter estimates $\hat{\theta}(t)$ are obtained recursively. By applying the same settings used in Section 4.1, the performance of the Kalman Filter method is evaluated. The maximum cumulative time error and the fifth maximum cumulative time error in the 100 simulations are recorded. These results are shown in Figure 4.3. The X-axis and Y-axis in Figure 4.3 have the same meanings as in Figure 4.2.

4.3 Discussion

Comparing Figure 4.2 with Figure 4.3, we find that the performances of the RLS method and the Kalman Filter method are almost identical. However, the RLS method is simpler than the Kalman Filter method. It does not require us to set the covariance matrix of the process noise and measurement noise. In the RLS method, the covariance matrix of the parameter estimates determines the distribution of the parameter prediction. This covariance matrix is obtained by calculating the product of $P(t)$ and the variance of the prediction error. The prediction error is the difference between the correction signal $y(t)$ and $\hat{\theta}(t) \cdot \varphi(t)$ and is shown in Equation (4.6).

$$Prediction\ error(t) = y(t) - \hat{\theta}(t) \cdot \varphi(t) \tag{4.6}$$

The distribution of the parameter prediction is critical for computing the CTE

upperbound of the OCXO analytically. This analytical result of the CTE upperbound is an important task of the thesis. In different experiments, the oscillator may show different cumulative time errors. Sometimes the CTE is large and sometimes it is small. When the oscillator is manufactured, one does not know if this oscillator is suitable for the specific application because of the random CTE. The CTE upperbound is the maximum CTE of the time module over a period of time. The analytical CTE upperbound is the maximum CTE analytically derived.

In the Kalman Filter method, the different settings of $R_1(t)$ and $R_2(t)$ affect the final result of the covariance matrix of the parameter estimates. Computing the correct covariance matrix of the parameter estimates in Kalman Filter method is more difficult than in the RLS. Hence, computing the analytical CTE upperbound in Kalman Filter is more difficult than in the RLS. The use of the Kalman Filter method is not as convenient as the RLS method. In summary, the RLS method is better than the BLS method and the Kalman Filter method in developing the adaptive control algorithm for the oscillator.

Chapter 5: Modeling Temperature Effect and The CTE Upperbound Analysis

In this chapter, the CTE upperbound of the oscillator stated in Chapter 3 is investigated. The CTE upperbound determines the performance bound of the oscillator and this upperbound is a function of the covariance matrix of the parameter estimates. Only the linear frequency stability dependence on temperature is considered in this chapter. Equation (4.1) is used as the system model in Section 5.1 and the shortcoming of the model is described. This model cannot determine the correct covariance matrix of the parameter estimates. Therefore, the CTE upperbound of the oscillator cannot be correctly obtained. In Section 5.2 and 5.3, an ARMAX model is used to investigate the CTE upperbound. The RPEM algorithm is used as the estimator method to estimate the parameters of the ARMAX model.

5.1 Model 1: The System Model without Control Loop

In the last chapter, Equation (4.1) is used as the system model structure and the correction signal is used as the system output for evaluating the RLS method and the Kalman Filter method. In this chapter, we continue using this system model to investigate the CTE upperbound of the oscillator. For the convenience of the readers, Equation (4.1)

is rewritten as Equation (5.1) here.

$$y(t) = \theta \cdot x(t) + v(t) \tag{5.1}$$

When a Recursive Least Squares method is applied to identify the system parameters, the parameter estimate $\hat{\theta}(t)$ obeys the Gaussian distribution [11]. Hence, $\hat{\theta}(t) - \theta$ also obeys the Gaussian distribution. The mean value of $\hat{\theta}(t) - \theta$ is 0. The covariance matrix of $\hat{\theta}(t) - \theta$ is the product of $P(t)$ in Equation (2.7c) and the variance of the prediction error [11]. The prediction error is calculated from Equation (4.6). The variance of the prediction error is shown in Equation (5.2) and the covariance matrix of $\hat{\theta}(t) - \theta$ is shown in Equation (5.3).

$$E\left(\left(y(t) - \hat{\theta}(t) \cdot x(t)\right)^2\right) = \sigma^2 \tag{5.2}$$

$$cov\left(\hat{\theta}(t) - \theta\right) = \sigma^2 P(t) \tag{5.3}$$

One should notice that the $P(t)$ investigated in this chapter is a scalar, so the covariance matrix of $\hat{\theta}(t) - \theta$ is actually equal to the variance of $\hat{\theta}(t) - \theta$.

The cumulative time error comes from the deviation of $\hat{\theta}(t)$ from θ. According to the standard statistical table, when the variance of $\hat{\theta}(t) - \theta$ is known, a 95% confidence interval of $\hat{\theta}(t) - \theta$ can be obtained. We use $\sigma 1$ to represent the standard deviation of $\hat{\theta}(t) - \theta$ and then the deviation of $\hat{\theta}(t)$ from θ should be smaller than $1.96 * \sigma 1$ with

probability larger than 95%. When the deviation of $\hat{\theta}(t)$ from θ is just equal to $1.96 * \sigma 1$, the 95% probability CTE upperbound in holdover mode is obtained analytically through Equation (5.4).

$$95\%\ CTE\ upperbound = \sum_{t=0}^{k} 1.96 * \sigma 1 * x(t) \tag{5.4}$$

In Equation (5.4), $t = 0$ corresponds to the start of the holdover mode. k represents the time that the holdover mode maintains. The term $x(t)$ still represents the temperature. Then one can use Monte Carlo simulation method to verify this analytical 95% CTE upperbound [22]. One hundred simulations are run and the fifth maximum CTE is recorded. The training time is set 4 hours and holdover time is set 20 hours. This fifth maximum CTE can approximately represent the simulation result of 95% CTE upperbound. The comparison results are shown in Figure 5.1.

In Figure 5.1, X-axis represents the time and Y-axis represents the cumulative time error. The solid line represents the simulation result of 95% CTE upperbound. The dashed line represents the analytical 95% CTE upperbound. One can see that the analytical CTE upperbound has a huge discrepancy from the simulation result of CTE upperbound. The reason for this discrepancy is described below.

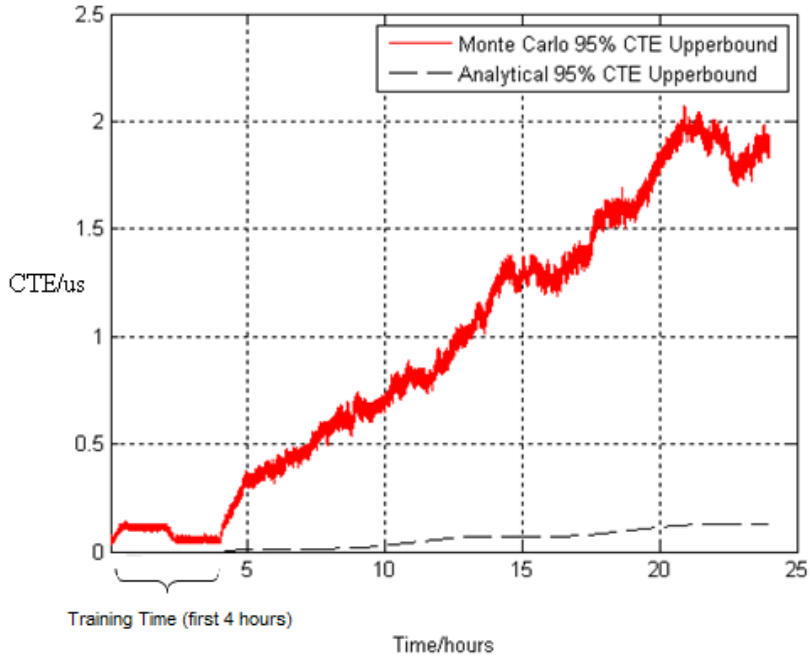

Figure 5.1 Comparison between Analytical and Simulation Results of 95% CTE Upperbound

The standard deviation of $\hat{\theta}(t) - \theta$ is used to determine the analytical CTE upperbound of the oscillator. The standard deviation is the square root of the variance of $\hat{\theta}(t) - \theta$. If this analytical variance is invalid, it cannot be used to determine the CTE upperbound.

The Monte Carlo method is applied to verify the validity of the variance of $\hat{\theta}(t) - \theta$ calculated from Equation (5.3). One hundred simulations are run and the $\hat{\theta}(t) - \theta$ in each simulation is recorded. Then the variance of $\hat{\theta}(t) - \theta$ from 100 simulations can be calculated through the definition of variance. If the analytical variance, which is calculated from Equation (5.3), is valid, it should be close to the variance calculated via

the Monte Carlo method.

The comparison which is shown in Figure 5.2 demonstrates that the variance calculated from the Monte Carlo method has a huge discrepancy from the variance calculated from the system identification method. The variance of $\hat{\theta}(t) - \theta$, calculated from Equation (5.3), is much lower than the variance obtained from the Monte Carlo method. Hence, the analytical CTE upperbound is much lower than the Monte Carlo result of the CTE upperbound, as the comparison result shown in Figure 5.1 confirms.

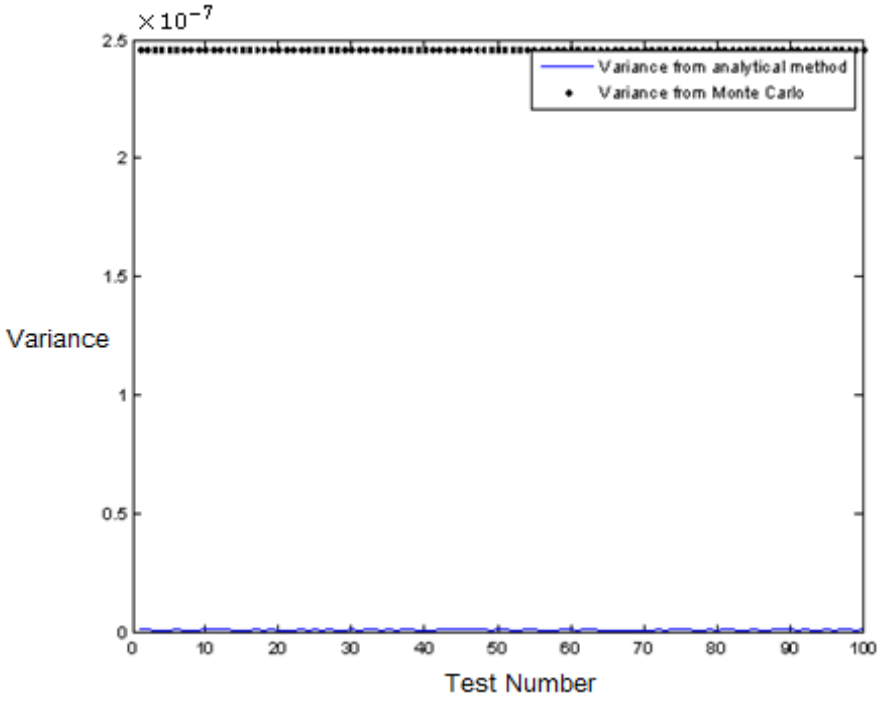

Figure 5.2 Variance Comparison between Analytical Method and Monte Carlo Method with RLS Applied

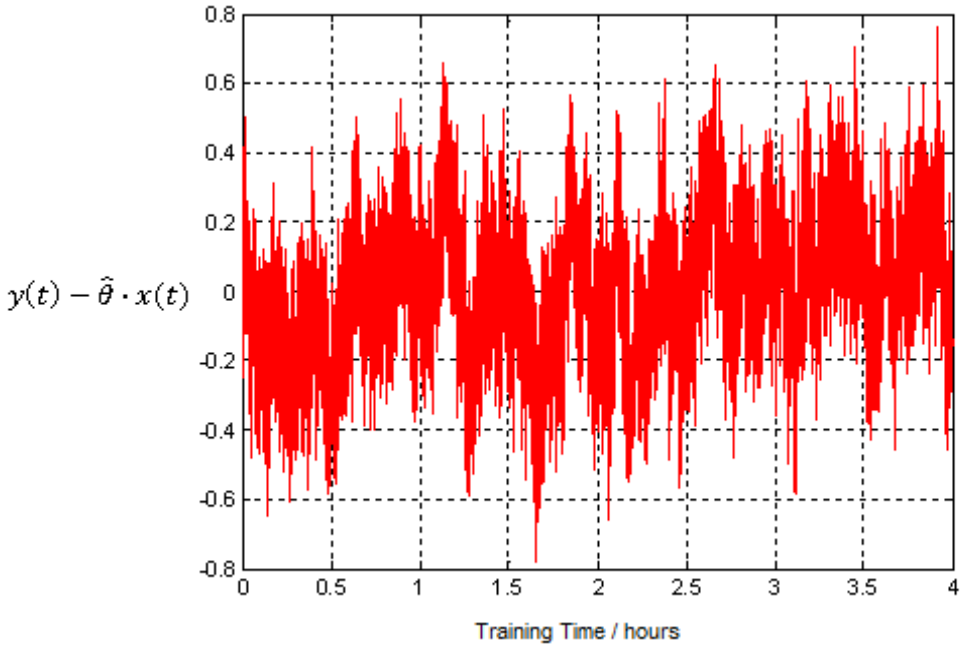

Figure 5.3 Prediction Error Plot for the Simple Model

The reason that the analytical variance of $\hat{\theta}(t) - \theta$ has a huge discrepancy from the variance calculated via the Monte Carlo method is described as follows. According to [7], the RLS method can be used validly if the prediction is unbiased. Then the plot of the prediction error should be Gaussian white noise. If the prediction is biased, the plot of the prediction error should be colored noise and the RLS method cannot be used validly.

Figure 5.3 shows the prediction error plot. In this figure, X-axis represents the time which is the total training time. Y-axis represents the prediction error which is calculated from Equation (4.6). This plot illustrates a colored noise and indicates that the prediction is biased. Therefore, the analytical variance calculated from Equation (5.3) is not close to the simulation result.

The system model structure must be modified to guarantee the prediction error is a Gaussian white noise. The RLS method or another system identification method then can be applied correctly and the valid analytical variance of $\hat{\theta}(t) - \theta$ can be obtained. The correct CTE upperbound then can be calculated.

Looking back at Equations (3.3) to (3.5), the correction signal is not obtained from Equation (5.1). The actual correction signal is created by a control loop and it is the combination of the former correction signals and the CTE [16]. Therefore, it can be assumed that the prediction error can be a Gaussian white noise and the estimate can be unbiased when the system model structure reflects the actual creation process of correction signal as far as possible [11]. In order to obtain the correct CTE upperbound, the control loop should be included in the system model structure.

5.2 Model 2: Including the Control Loop in the System Model

Referring to Section 3.1.2, the correction signal is created by the control loop which is represented by Equations (3.3) to (3.5). For the convenience of the readers, Equations (3.3) to (3.5) are rewritten as Equations (5.5) to (5.7) respectively.

$$CTE_k = CTE_{k-1} + \beta * \varepsilon_k = \sum_{t=0}^{k-1} \beta * \varepsilon_{t+1} \quad (5.5)$$

$$correct_{ref} = \frac{1}{N}\sum_{t=k-N}^{k-1} y(t) \quad (5.6)$$

$$y(k) = correct_{ref} - CTE_k/damp \quad (5.7)$$

The term CTE_k represents the cumulative time error at the *k-th* second. The term $y(t)$ represents the correction signal at time *t* and $y(k)$ represents the correction signal at time *k*. The term β represents the digital phase detector resolution and the term ε_k represents error counts at the *k-th* second. The term $\beta * \varepsilon_k$ represents the time error at the k_{th} second. For example, the system has run for 4 hours (14400 seconds) and the cumulative time error is $5\mu s$. Thus, k is 14400 and CTE_{14400} is $5\ \mu s$. If the time error at the third second is $1ns$, then $\beta * \varepsilon_3$ is $1\ ns$. Equation (5.5) shows that CTE_k is the integration of all time errors from the first second to the *k-th* second. Equation (5.5) also shows that the time error obtained at each second and CTE_k must be multiples of β. In Equation (5.6), the term $correct_{ref}$ is the average value of the last *N* correction signals and *N* should be a large constant. The term $damp$ is a constant which serves as a suppression of the GPS receiver noise. In the simulation, *N* is set to 2000 and $damp$ is set to 150. In Equation (5.7), $y(k)$ represents the correction signal at the *k-th* second when the timing module is locked by the GPS 1pps signal. Thus, by combining Equations (5.5) to (5.7), Equation (5.8) is obtained.

$$y(k) = \left(\frac{1}{2000}\right) * \sum_{t=k-2000}^{k-1} y(t) - \left(\frac{1}{150}\right) * CTE_k \quad (5.8)$$

The time error is caused by the GPS noise and the frequency stability of the oscillator. In this chapter, only the linear frequency stability dependence on temperature is considered for the oscillator and the initial frequency stability is set to zero. Equation (5.9) shows the relation between the frequency stability of the oscillator and the temperature.

$$Oscillator_{stab}(k) = \theta_0 \cdot u(k) \tag{5.9}$$

The term $Oscillator_{stab}(k)$ represents the oscillator frequency stability at the *k-th* second. The term $u(k)$ represents the temperature. The term θ_0 is the temperature sensitivity of the oscillator frequency stability. According to Equation (3.8), Equation (5.10) is obtained.

$$Oscillator_{stab}(k) = \frac{\Delta f_k}{f_0} \tag{5.10}$$

The term Δf_k represents the oscillator frequency drift at the *k-th* second and the term f_0 represents the nominal frequency. The time error Δt caused by the frequency stability and the time duration T for which the frequency stability is maintained is related to the oscillator stability $\Delta f / f_0$ through Equation (5.11)

$$\frac{\Delta t}{T} = \frac{\Delta f}{f_0} \tag{5.11}$$

$Oscillator_{stab}\,(k)$ is only maintained at the *k-th* second, so the time duration T is 1 second. Therefore, the time error Δt_k caused by the frequency stability at the *k-th* second is equal to $Oscillator_{stab}\,(k)$. In training mode, the oscillator stability is corrected by the correction signal $y(k-1)$ at the *k-th* second. The actual time error measured at the *k-th* second should be the summation of $Oscillator_{stab}\,(k)$, $y(k-1)$, and the measurement noise. The actual time error measured is shown in Equation (5.12):

$$\begin{aligned}\Delta t_{measured}\,(k) &= Oscillator_{stab}\,(k) + y(k-1) + v(k) - v(k-1)\\ &= \theta_0 \cdot u(k) + y(k-1) + v(k) - v(k-1)\end{aligned} \tag{5.12}$$

The term $v(k)$ represents the GPS noise at the *k-th* second. One should have noticed that the measurement noise in Equation (5.12) is $v(k) - v(k-1)$. The reason is as follows. The measurement noise of the system comes from the GPS noise jitters. The GPS receiver receives the GPS 1 pulse per second (pps) signal. If there are no GPS noise jitters, the distance between GPS pulses should be exactly 1 second. However, GPS noises always exist and the distortion of the jitter has to be added into the distance between pulses. For example, if both the first and second GPS 1 pps signals are distorted by a +10 *ns* jitter, both the first and second pulse edges move +10 ns. Then the distance between two pulse edges is still 1 second. The measurement noise perceived is 0 *ns*. If

the first 1 pps signal is distorted by a +10 *ns* jitter and the second 1 pps signal is distorted by a -10 *ns* jitter, the first edge moves +10 *ns* and the second edge moves -10 *ns*. The distance between two pulse edges is 1 second minus 20 *ns*. The measurement noise perceived is -20 *ns*. Hence, the measurement noise is given by Equation (5.13).

$$Measurement\ noise(k) = v(k) - v(k-1) \tag{5.13}$$

According to Section 3.1.2, $\beta * \varepsilon_k$ can represent the time error measured at the *k-th* second, so Equation (5.14) is obtained.

$$\beta * \varepsilon_k = \theta_0 \cdot u(k) + y(k-1) + v(k) - v(k-1) \tag{5.14}$$

Referring to Figure 3.3, in the control loop, a DAC is used to transfer the correction signal to a control voltage signal. This control voltage signal is fed into the oscillator tuning port to correct the oscillator. When the DAC resolution and the digital phase detector resolution are infinite, the ideal cumulative time error is obtained from Equation (5.15).

$$Ideal\ CTE_k = \sum_{t=0}^{k-1}(\theta_0 \cdot u(t+1) + y(t) + v(t+1) - v(t)) \tag{5.15}$$

However, the DAC resolution and the digital phase detector resolution are not

infinite. In the simulation, the DAC resolution is 0.0229 ppb. Hence, when the correct signal is used to tune the oscillator, it must be multiples of 0.0229. The digital phase detector resolution is set to 6.25 ns. Hence, the time error measured at each second must be multiples of 6.25. The cumulative time error must be multiples of 6.25 too. The parameter θ_0 needs to be identified. In the simulation, θ_0 is set to 0.0533 and $v(t)$ has a mean value of zero and standard deviation of 20 ns. Thus, the cumulative time error is obtained from Equation (5.16).

$$CTE_k = \sum_{t=0}^{k-1}\left(6.25 * fix\left(\frac{v(t+1)-v(t)+\theta_0 * u(t+1)+0.0229 * fix\left(\frac{y(t)}{0.0229}\right)}{6.25}\right)\right) \qquad (5.16)$$

In Equation (5.16), the term $6.25 * fix(\frac{\cdot}{6.25})$ guarantees that the time error measured at each second and the cumulative time error are multiples of 6.25 ns. The meaning of $fix(\cdot)$ is explained in Section 3.1.2. The term $0.0229 * fix(\frac{y(t)}{0.0229})$ guarantees that the correction signal working on the oscillator is in multiples of 0.0229 ppb. Combining Equations (5.8) and (5.16), the system model structure in Equation (5.17) is obtained.

$$y(k) = \left(\frac{1}{2000}\right) * \sum_{t=k-2000}^{k-1} y(t) - \left(\frac{1}{150}\right) * \sum_{t=0}^{k-1}\{6.25 * fix[(v(t+1) - v(t) + \theta_0 * u(t+1) + 0.0229 * fix(\frac{y(t)}{0.0229}))/6.25]\} \qquad (5.17)$$

Equation (5.17) represents the detailed mathematical form of the control loop which is described in Section 3.1.2. However, referring to Section 2.2.1, the system model must have a form similar to Equation (2.2) in order to use the system identification method. Therefore, Equation (5.17) must be rearranged to Equation (5.18).

$$Y1(k) = (-150) * y(k) + \left(\frac{150}{2000}\right) * \sum_{t=k-2000}^{k-1} y(t)$$
$$= \sum_{t=0}^{k-1}\{6.25 * fix[(v(t+1) - v(t) + \theta_0 * u(t+1) +$$
$$0.0229 * fix(\frac{y(t)}{0.0229}))/6.25]\} \quad (5.18)$$

The $\sum$ in Equation (5.18) can be removed by computing the difference between $Y1(k)$ and $Y1(k-1)$.

$$Y2(k) = Y1(k) - Y1(k-1)$$
$$= 6.25 * fix[(v(k) - v(k-1) + \theta_0 * u(k)$$
$$+0.0229 * fix(\frac{y(k-1)}{0.0229}))/6.25] \quad (5.19)$$

For simplicity, we introduce

$$B(k-1) = 0.0229 * fix(\frac{y(k-1)}{0.0229}) \quad (5.20)$$

So,

$$Y2(k) = 6.25 * fix[\frac{v(k)-v(k-1)+\theta_0 * u(k)+B(k-1)}{6.25}]$$
$$= [v(k) - v(k-1) + \theta_0 * u(k) + B(k-1)]$$

$$+6.25 * fix\left[\frac{v(k)-v(k-1)+\theta_0 * u(k)+B(k-1)}{6.25}\right]$$

$$-[v(k)-v(k-1)+\theta_0 * u(k)+B(k-1)] \quad (5.21)$$

We introduce

$$\delta Y2(k) = 6.25 * fix\left[\frac{v(k)-v(k-1)+\theta_0 * u(k)+B(k-1)}{6.25}\right] - [v(k)-v(k-1)+$$

$$\theta_0 * u(k)+B(k-1)] \quad (5.22)$$

So,

$$Y2(k) = v(k)-v(k-1)+\theta_0 * u(k)+B(k-1)+\delta Y2(k) \quad (5.23)$$

We introduce $Y3(k)$ which equals the difference between $Y2(k)$ and $B(k-1)$. Thus, Equation (5.24) is obtained.

$$Y3(k) = Y2(k) - B(k-1)$$

$$= v(k)-v(k-1)+\theta_0 * u(k)+\delta Y2(k) \quad (5.24)$$

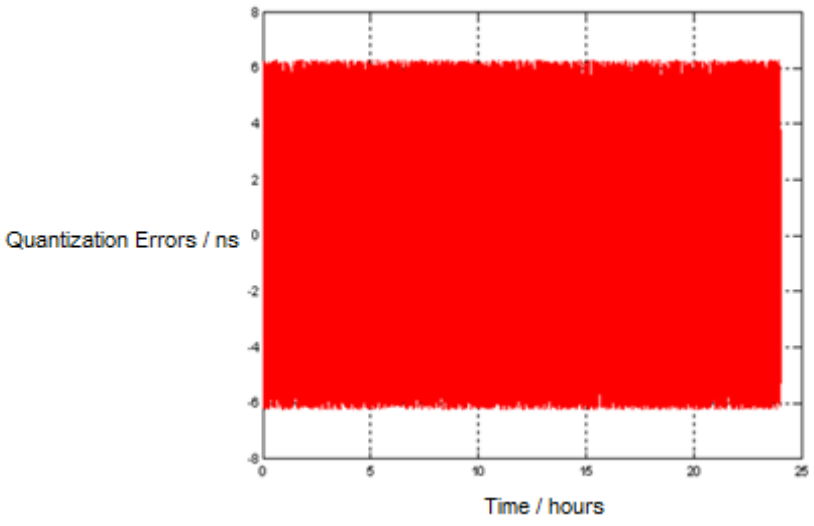

Figure 5.4 Quantization Error Caused by the Phase Detector Resolution

The term $\delta Y2(k)$ is the quantization error caused by the digital phase detector resolution, which is limited between -6.25 and +6.25 in the simulation. Figure 5.4 shows the graph of $\delta Y2(k)$. The X-axis represents the training time. The Y-axis represents the quantization error which is calculated from Equation (5.22).

In Equation (5.24), $Y3(k)$ becomes the new system output. $u(k)$ is still the system input and θ_0 is still the parameter that needs to be identified. Equation (5.24) has similar form and characteristics as Equation (2.2) and can therefore be analyzed using a standard system identification method. However, Equation (5.24) and Equation (2.2) also exhibit some differences. One of the differences is the measurement noise. In Equation (2.2), the measurement noise is independent white noise. In Equation (5.24), the measurement noise is $v(k) - v(k-1)$. These measurement noises are not independent from each other. Referring to Section 2.2.4, Equation (5.24) is an ARMAX model and should be solved by the Recursive Prediction Error Method (RPEM). Another difference is that Equation (5.24) contains a quantization error term $\delta Y2(k)$ which can also be solved by RPEM approximately.

A new ARMAX model is created and shown in Equation (5.25).

$$y_{RPEM}(k) = a * u(k) + v(k) + b * v(k-1) \tag{5.25}$$

The term $y_{RPEM}(k)$ is the $Y3(k)$ in Equation (5.24). The term $u(k)$ and $v(k)$ are still the temperature and the GPS receiver noise. The term a is the temperature sensitivity

which is the θ_0 in Equation (5.24). The term $b * v(k-1)$ represents the term $v(k-1)$ and the quantization error term $\delta Y2(k)$ in Equation (5.24). The parameters a and b constitute the new parameter vector θ_{RPEM}. The term $\hat{a}$ represents the estimate of a. By using RPEM, this ARMAX model is solved recursively and the distribution of parameter estimates is obtained. Simulation results are shown in the next section. Section 2.2.4 described RPEM in more detail.

The Monte Carlo method can be used to verify whether using the model of Equation (5.25) is better than using the model of Equation (5.1). The system model of Equation (5.25) is referred to as the system model including the control loop. The term system model without including the control loop refers to the model of Equation (5.1). The performance of the system model can be represented by the maximum cumulative time error in 100 simulations. Figure 5.5 shows the comparison result between the maximum CTE when using the system model of Equation (5.1) and the maximum CTE when using the system model of Equation (5.25). In Figure 5.5, the X-axis represents the time and the Y-axis represents the maximum CTE in 100 simulations. Training time is set 4 hours and holdover time is set to 20 hours.

Figure 5.5 shows that the performances of both system models are almost the same. This comparison is still true when the training time is changed. In the next chapter, when other factors such as ageing rate are introduced in the system model, one will see that the performance of the model including the control loop has advantages, which means that the model produces a lower maximum CTE.

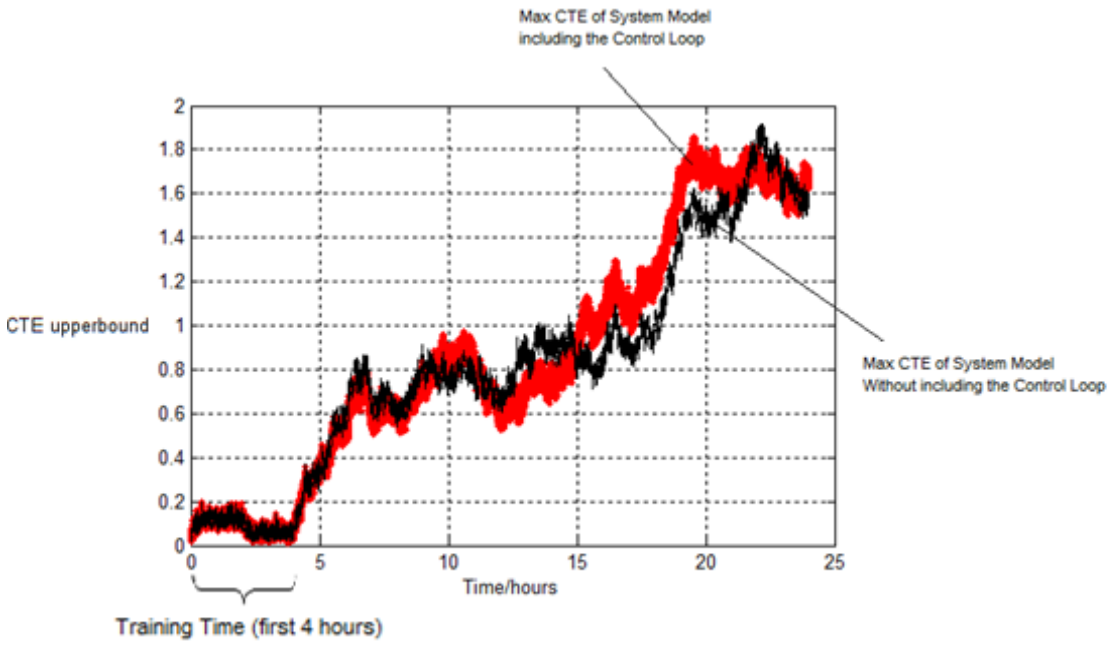

Figure 5.5 Simple System Model Performance Comparisons with Training 4 Hours

5.3 Simulation Result of the ARMAX Model

First, the analytical result for the distribution of parameter estimate $\hat{a}$ in one simulation is shown in Figure 5.6. In this simulation, the training time is set to 2 hours.

According to [11], the parameter estimate $\hat{a}$ should have a Gaussian distribution. The distribution of $\hat{a} - a$ has the form of

$$\hat{a} - a \in N(0, P_N) \tag{5.26}$$

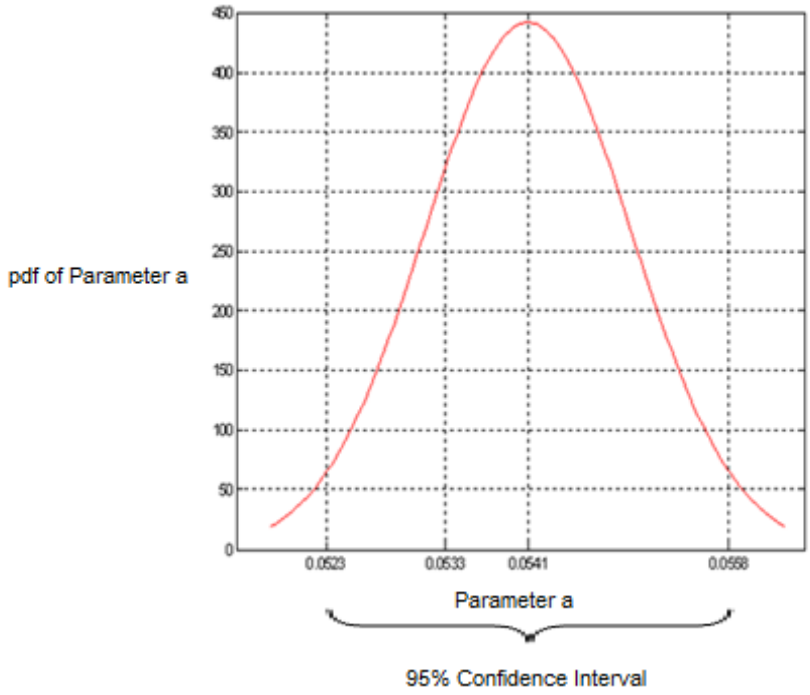

Figure 5.6 Distribution of Parameter Estimate $\hat{a}$

The system parameter estimates are obtained from Equation (2.34a). The first term of estimates is $\hat{a}$. The variance P_N of $\hat{a}$ is the product of $P_{11}(t)$ and the variance of $\varepsilon(t)$ in Equation (2.31). $P_{11}(t)$ is the term of first row and first column of $P(t)$ in Equation (2.34b). $\varepsilon(t)$ is the prediction error sequence. According to the standard statistical table, the 95% probability confidence interval of a is computed via:

$$\hat{a} - 1.96 * \sqrt{P_N} \leq a \leq \hat{a} + 1.96 * \sqrt{P_N} \tag{5.27}$$

In this simulation,

$$a = 0.0533 \tag{5.28a}$$

$$\hat{a} = 0.0541 \tag{5.28b}$$

$$P_N = 7.523 \times 10^{-7} \tag{5.28c}$$

$$\hat{a} - 1.96 * \sqrt{P_N} = 0.0523 \tag{5.28d}$$

$$\hat{a} + 1.96 * \sqrt{P_N} = 0.0558 \tag{5.28e}$$

Since the distribution of $\hat{a}$ is known, the 95% probability upperbound of the time error can be calculated analytically. The real parameter a is located on the 95% probability bound (upper or lower bound) when the 95% probability upperbound of the time error is reached. The corresponding time error is calculated through the difference between $\hat{a}$ and the bound $\hat{a} - 1.96 * \sqrt{P_N}$ or $\hat{a} + 1.96 * \sqrt{P_N}$. In either case, the difference is $1.96 * \sqrt{P_N}$. This shows that the analytical time error is only related to the variance of $\hat{a}$.

$$Time\ error = 1.96 * \sqrt{P_N} * temperature \tag{5.29}$$

The cumulative time error is also computed through the integration of the time error second by second. Figure 5.7 shows the analytical result for the 95% probability upperbound of the cumulative time error.

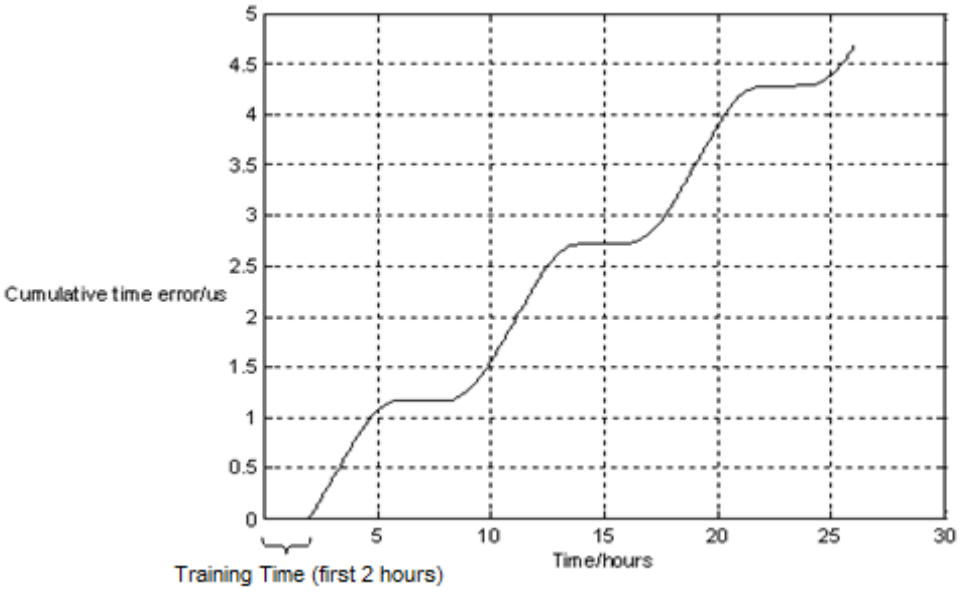

Figure 5.7 Analytical Result for the 95% Upperbound of CTE

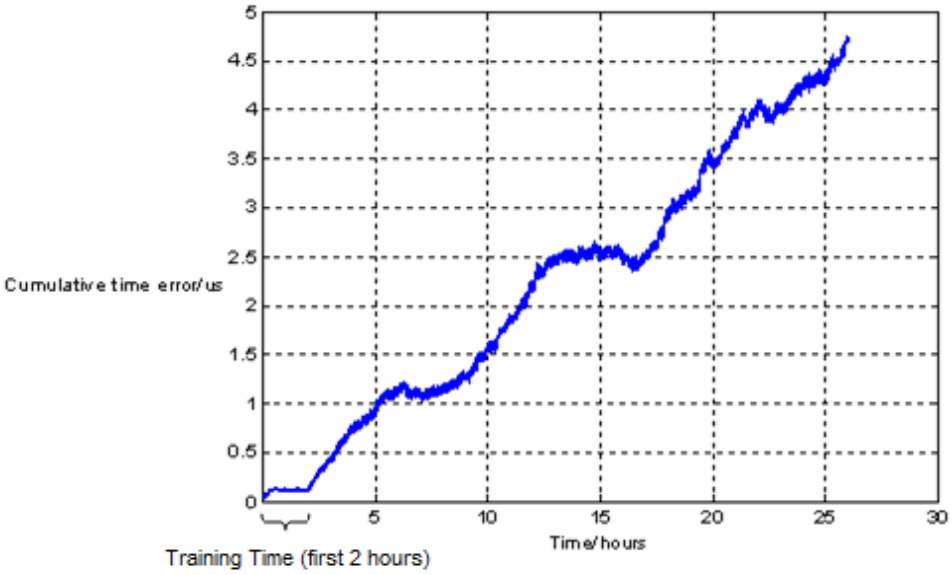

Figure 5.8 Monte Carlo Result for the 95% Upperbound of CTE

For verifying the CTE shown in Figure 5.7, 100 independent simulations have been run. In each simulation, the training time is set to 2 hours and holdover time is set to 24 hours. The fifth maximum CTE in 100 repetitions can approximately serve as the Monte Carlo result for the 95% upperbound of cumulative time error, which is shown in Figure 5.8. By comparing Figure 5.7 and 5.8, one can see the analytical CTE upperbound is very close to the simulation result of the CTE upperbound.

Figure 5.7 and Figure 5.8 show periodic phases. From the 6^{th} hour to the 8^{th} hour, from the 14^{th} hour to the 16^{th} hour, and from the 22^{th} hour to the 24^{th} hour, the CTE curve almost keeps flat. The temperature profile is needed to explain this periodic phases because the temperature is the only input data.

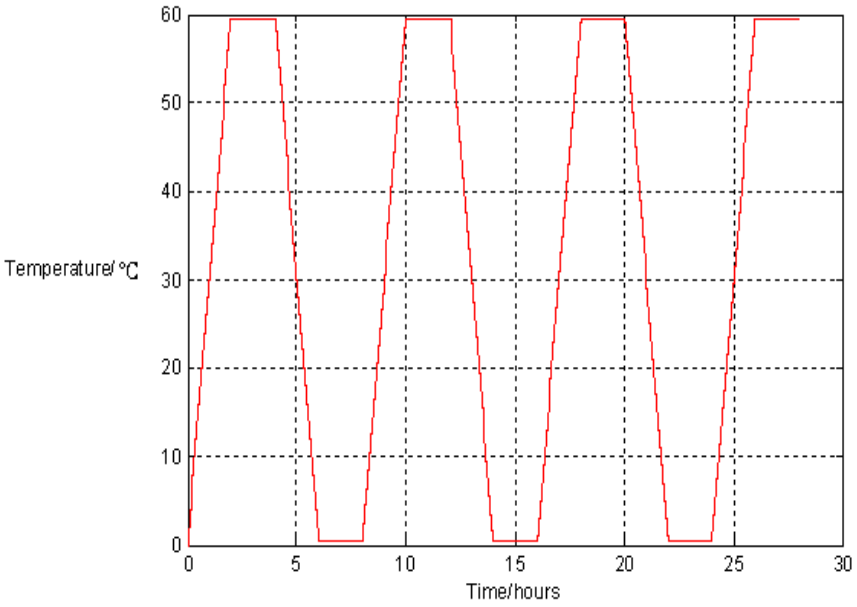

Figure 5.9 Temperature Profile

Figure 5.9 shows the temperature profile used in this chapter. One can see the

temperature is 0 when the time is from the 6^{th} hour to the 8^{th} hour, from the 14^{th} hour to the 16^{th} hour, and from the 22^{th} hour to the 24^{th} hour. Referring to Equation (5.29), when the temperature is zero, the time error is zero. Therefore, the cumulative time error does not increase during these times because the CTE is the integration of the time error.

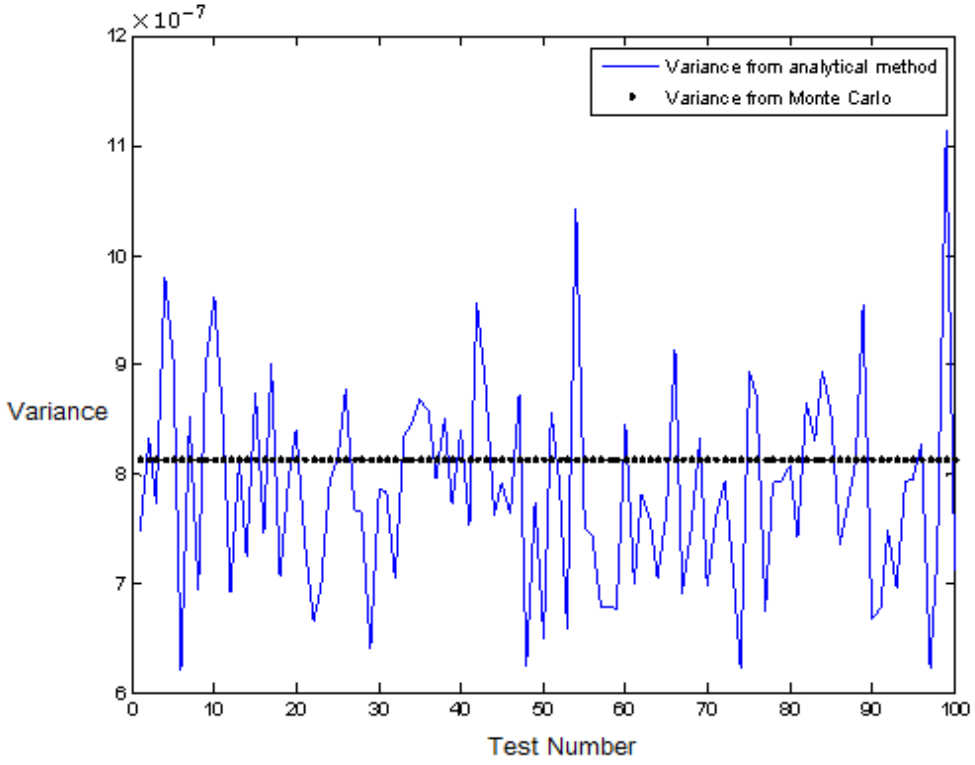

Figure 5.10 Variance Comparisons between Analytical Method and Monte Carlo Method with ARMAX Model and RPEM Applied

Figure 5.10 shows the comparison result of another verification method. 100 independent simulations have been run and in each simulation, the variance P_N is calculated analytically. Each P_N is connected with each other with the solid line. The

parameter estimate $\hat{a}$ is also calculated in each simulation. Via the variance definition, the variance of $\hat{a}$ calculated from the Monte Carlo method is obtained. It is represented by the dot. Figure 5.10 illustrates that the analytical parameter estimate variances fluctuate around the variance computed from the Monte Carlo method.

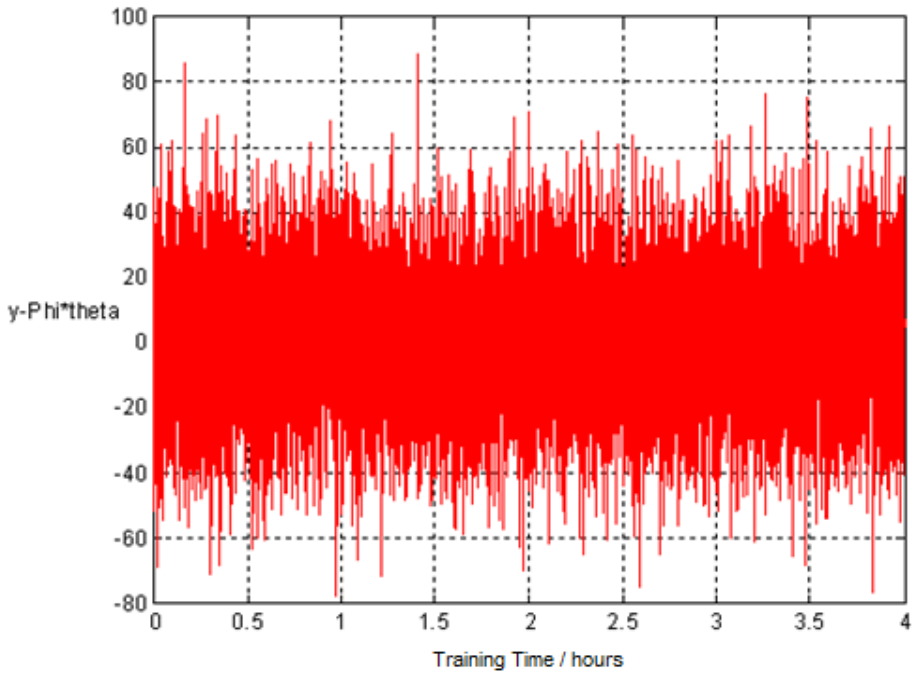

Figure 5.11 Prediction Error Plot for the Model Including the Control Loop

Figure 5.11 is the plot of the prediction error $\varepsilon(t)$ computed from Equation (2.31). The plot shows Gaussian white noise and indicates that the parameter estimation by using RPEM is unbiased.

Figure 5.12 shows the CTE upperbound computed analytically via RPEM. The X-axis coordinates represent the training time, varying from 1 hour to 24 hours. The Y-axis

coordinates represent the predicted 95% probability max CTE. The holdover time is fixed to 24 hours. One can observe that the max CTE becomes smaller and smaller, ideally approaching zero when the training time is arbitrarily long. However, the max cumulative time error cannot really be zero when the timing module is working in the base station because there are other factors affecting the accuracy of oscillator besides the temperature.

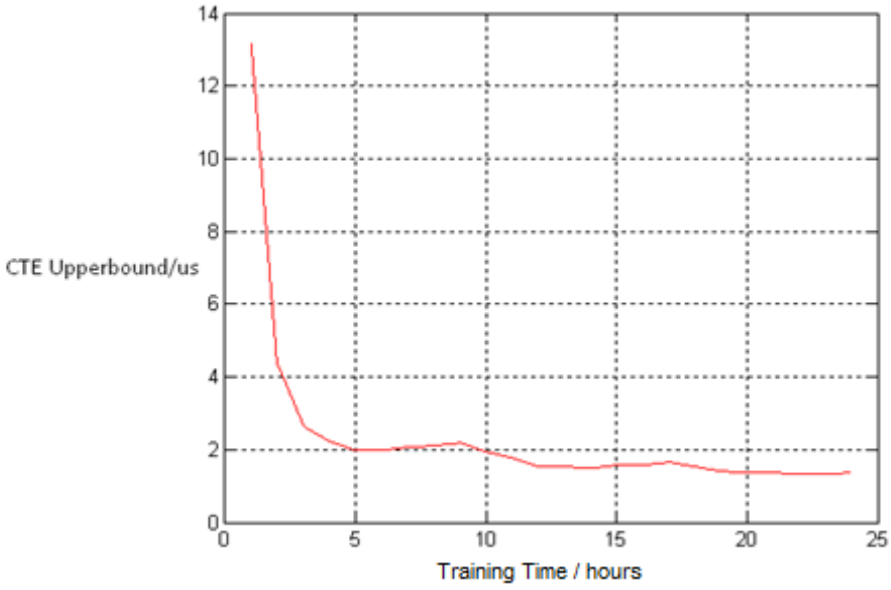

Figure 5.12 CTE Upperbound when Training Time is from 1 to 24 Hours and Holdover Time is 24 Hours

5.4 Discussion

In this chapter, the CTE upperbound of the oscillator is investigated. This oscillator is described in Chapter 3. The temperature is linear related to the frequency stability of the oscillator and is the only input. First, a simple system model of the oscillator

frequency stability is used. Equation (5.1) is the model equation. The simulation results show that the analytical CTE upperbound calculated via this simple model is not close to the Monte Carlo CTE upperbound. Analyzing the control loop, which creates the correction signal for the oscillator, the system model structure is modified to include the control loop in the model equation. The new system model is an ARMAX model and RPEM is used to estimate the parameter. The analytical CTE upperbound is obtained through the parameter estimates. The simulation results show that the analytical CTE upperbound is very close to the Monte Carlo CTE upperbound. The performance of applying the ARMAX model is compared with the model of Equation (5.1). Simulations show that their performances are close.

Chapter 6: Refined Model Including Temperature and Ageing Effect

In the last two chapters, temperature is considered as the only factor affecting the frequency stability of the oscillator. In this chapter, a more detailed frequency stability dependence model is presented and used for studying the CTE upperbound of the oscillator. The more detailed system model combines the temperature effect and the ageing effect. In this model, temperature has a quadratic relation with the frequency stability and there is a non-zero initial frequency stability offset in the model.

6.1 Refined Frequency Stability Dependence Model

In Chapter 5, we showed that the cumulative time error CTE_k is the summation of all time errors from the first second to the *k-th* second. The time error at each second is caused by the GPS noise and the frequency stability error of the oscillator. The oscillator frequency stability exhibits dependencies on many environmental factors. In Chapters 4 and 5, the temperature is considered as the only factor affecting the frequency stability and a straight line fit is applied to the temperature and the frequency stability. The initial frequency stability offset is set to zero. When the relation between the ambient temperature and the oscillator frequency stability is close to linear, a straight line fit is

suitable. However, in most situations, a higher order polynomial fit is more suitable than a linear fit. In this chapter, a quadratic fit replaces the linear fit. Furthermore, if the training time is long enough, the ageing effect on both training time and holdover time is not trivial. It should be included into the system model. Finally, the initial frequency stability offset is not zero in most situations.

Hence, a more detailed oscillator frequency stability model is presented in Equation (6.1).

$$Oscillator_{stab}(k) = a \cdot u^2(k) + b \cdot u(k) + c + d \cdot k \tag{6.1}$$

The term $Oscillator_{stab}(k)$ represents the frequency stability of the oscillator at the *k-th* second. The term $u(k)$ represents the temperature at the *k-th* second. The parameters a and b represent the temperature sensitivity of the frequency stability for the quadratic and the linear term, respectively. The parameter c represents the initial frequency stability offset and the term d represents the ageing rate of the frequency stability. According to Equations (3.1) and (3.8),

$$Oscillator_{stab}(k) = \frac{\Delta f_k}{f_0} = \frac{\Delta t_k}{T} \tag{6.2}$$

The term Δf_k represents the frequency error at the *k-th* second and the term f_0 represents the nominal frequency. The term Δt_k represents the time error caused by the

frequency stability at the *k-th* second and the term T represents the time duration. Actually, the time duration is 1 second at the *k-th* second. Thus, the time error Δt_k is equal to $Oscillator_{stab}(k)$. Since $Oscillator_{stab}(k)$ is corrected by the correction signal $y(k-1)$ in training mode and the GPS noise $v(k)$ is introduced, the total time error, measured at the *k-th* second is:

$$\begin{aligned}\Delta t_{measured}\ (k) &= Oscillator_{stab}(k) + y(k-1) + v(k) - v(k-1) \\ &= a \cdot u^2(k) + b \cdot u(k) + c + d \cdot k\ + y(k-1) \\ &\quad + v(k) - v(k-1) \end{aligned} \tag{6.3}$$

Equation (6.3) is equivalent to Equation (5.12), except for the different oscillator frequency stability model. The measurement noise is $v(k) - v(k-1)$. The reason of choosing $v(k) - v(k-1)$ rather than $v(k)$ is explained in Section 5.2. According to Equation (5.5), when the DAC resolution and the digital phase detector resolution are infinite, the ideal cumulative time error is:

$$\begin{aligned}Ideal\ CTE_k = \textstyle\sum_{t=0}^{k-1}(a \cdot u^2(t+1) + b \cdot u(t+1) + c + d \cdot (t+1) + y(t) + \\ v(t+1) - v(t)) \end{aligned} \tag{6.4}$$

However, the DAC has a resolution of 0.0229 ppb and the digital phase detector

resolution is 6.25 ns. The term $6.25 * fix(\frac{\cdot}{6.25})$ is used to guarantee that CTE_k is a multiple of 6.25 ns and the term $0.0229 * fix\left(\frac{y(t)}{0.0229}\right)$ is used to guarantee the correction signal for the oscillator is a multiple of 0.0229 ppb. Thus, Equation (6.5) is obtained.

$$CTE_k = \sum_{t=0}^{k-1}\{6.25 * fix[(v(t+1) - v(t) + a * u^2(t+1) + b * u(t+1) + c + d * (t+1) + 0.0229 * fix(\frac{y(t)}{0.0229}))/6.25]\} \quad (6.5)$$

According to Equation (5.6) and (5.7), the correction signal $y(k)$ is the difference between the average value of the last N correction signals and $CTE_k/damp$. When N is 2000 and $damp$ is 150, combining Equations (6.5) and (5.8), Equation (6.6) is obtained.

$$y(k) = \left(\frac{1}{2000}\right) * \sum_{t=k-2000}^{k-1} y(t) - \left(\frac{1}{150}\right) * \sum_{t=0}^{k-1}\{6.25 * fix[(v(t+1) - v(t) + a * u^2(t+1) + b * u(t+1) + c + d * (t+1) + 0.0229 * fix(\frac{y(t)}{0.0229}))/6.25]\} \quad (6.6)$$

In Equation (6.6), the parameters a, b, c and d are four parameters that need to be identified. In the simulation, a is set to -3.1966×10^{-4}, b is set to 0.0533, c is set to 21 and d is set to 1.1574×10^{-5} ppb/s. The value of d represents the 1ppb frequency drift per day for the ageing effect. Equation (6.6) is a more detailed model which can simulate

the physical process of the control loop more precisely than Equation (5.17). In Section 5.2, Equation (5.17) is rearranged to Equation (5.24) in order to use the RPEM method. Equation (6.6) also needs rearrangement to a suitable form similar to Equation (2.2) to use the RPEM method.

Similar to the deduction steps from Equation (5.18) to (5.24), Equation (6.6) is rearranged to Equation (6.7).

$$\begin{aligned} Y1(k) &= (-150) * y(k) + \left(\frac{150}{2000}\right) * \sum_{t=k-2000}^{k-1} y(t) \\ &= \sum_{t=0}^{k-1}\{6.25 * fix[(v(t+1) - v(t) + a * u^2(t+1) + \\ &\quad b * u(t+1) + c + d * (t+1) + 0.0229 * fix(\frac{y(t)}{0.0229}))/6.25]\} \end{aligned} \tag{6.7}$$

The $\sum$ in Equation (6.7) can be removed by computing the difference between $Y1(k)$ and $Y1(k-1)$.

$$\begin{aligned} Y2(k) &= Y1(k) - Y1(k-1) \\ &= 6.25 * fix[(v(k) - v(k-1) + a * u^2(k) + b * u(k) \\ &\quad c + d * k + 0.0229 * fix(\frac{y(k-1)}{0.0229}))/6.25] \end{aligned} \tag{6.8}$$

For simplicity, we introduce

$$B(k-1) = 0.0229 * fix(\frac{y(k-1)}{0.0229}) \tag{6.9}$$

So,

$$Y2(k) = 6.25 * fix[\frac{v(k)-v(k-1)+a*u^2(k)+b*u(k)+c+d*k+B(k-1)}{6.25}]$$

$$= [v(k) - v(k-1) + a * u^2(k) + b * u(k) + c + d * k + B(k-1)]$$

$$+6.25 * fix[\frac{v(k)-v(k-1)+a*u^2(k)+b*u(k)+c+d*k+B(k-1)}{6.25}]$$

$$-[\, v(k) - v(k-1) + a * u^2(k) + b * u(k) + c$$

$$+d * k + B(k-1)] \quad (6.10)$$

We introduce

$$\delta Y2(k) = 6.25 * fix\left[\frac{v(k)-v(k-1)+a*u^2(k)+b*u(k)+c+d*k+B(k-1)}{6.25}\right] - [v(k) -$$

$$v(k-1) + a * u^2(k) + b * u(k) + c + d * k + B(k-1)] \quad (6.11)$$

So,

$$Y2(k) = v(k) - v(k-1) + a * u^2(k) + b * u(k) + c + d * k + B(k-1)$$

$$+\delta Y2(k) \quad (6.12)$$

We thus define $Y3(k)$ as the difference between $Y2(k)$ and $B(k-1)$

$$Y3(k) = Y2(k) - B(k-1)$$

$$= v(k) - v(k-1) + a * u^2(k) + b * u(k)$$

$$+c + d * k + \delta Y2(k) \quad (6.13)$$

Similar to Equation (5.22) in Chapter 5, the term $\delta Y2(k)$ is the quantization error caused by the digital phase detector resolution, which is limited between -6.25 and +6.25 in the simulation.

Equation (6.13), which is equivalent to Equation (6.6), is the system model. Equation (6.13) is an ARMAX model except for the inclusion of a quantization error $\delta Y2(k)$. The corresponding ARMAX model structure for identifying this system is given by Equation (6.14).

$$y_{RPEM}(k) = a * u^2(k) + b * u(k) + c + d * k + v(k) + e * v(k-1) \quad (6.14)$$

The parameter vector is

$$\theta_{RPEM} = [a \;\; b \;\; c \;\; d \;\; e]^T \quad (6.15)$$

Correspondingly, the parameter estimate vector of Equation (6.14) is:

$$\hat{\theta}_{RPEM} = \left[\hat{a} \;\; \hat{b} \;\; \hat{c} \;\; \hat{d} \;\; \hat{e}\right]^T \quad (6.16)$$

The value of $y_{RPEM}(k)$ is equal to $Y3(k)$ in Equation (6.13). The term $e * v(k-1)$ represents the term $\delta Y2(k) - v(k-1)$ in Equation (6.13). The Recursive Prediction Error Method is used to solve Equation (6.14). The parameters a, b, c, d and e are 5 parameters that need to be identified.

In the remainder of this document, the model developed above will be referred to as the system model including the control loop. The term system model without including the control loop refers to the model structure of Equation (4.1). Because the model structure of Equation (4.1) only considers the temperature effect, a multi-parameter

model without including the control loop is created below for comparing the performance with the model including the control loop.

$$y(t) = a * x^2(t) + b * x(t) + c + d * t + v(t) \quad (6.17)$$

Similar to Chapter 4, the term $y(t)$ represents the correction signal. The term $x(t)$ represents the temperature. The term $v(t)$ represents the measurement noise. The terms a and b represent the temperature sensitivity for the quadratic term and linear term. The term c represents the initial frequency stability offset. The term d represents the ageing rate. The RLS method is used to solve Equation (6.17).

The Monte Carlo method is used to verify whether using the model of Equation (6.14) is better than using the model of Equation (6.17). The training time is set 4 hours and the holdover time is 20 hours. One hundred simulations are run when including the control loop in the system model and one hundred simulations are run without including the control loop in the system model. The maximum CTE is recorded. Similar to previous chapters, we use the maximum CTE in 100 simulations to represent the performance of the system model. A lower maximum CTE indicates better performance of the system model. The comparison result is shown in Figure 6.1.

In Figure 6.1, the X-axis represents the time and the Y-axis represents the maximum CTE in 100 simulations. The dashed line represents the maximum CTE when using the system model of Equation (6.17), which means that the system model does not include

the control loop. The solid line represents the maximum CTE when using the system model of Equation (6.14), which means that the system model includes the control loop. Obviously, the maximum CTE of using the system model of Equation (6.17) is larger than using the system model of Equation (6.14).

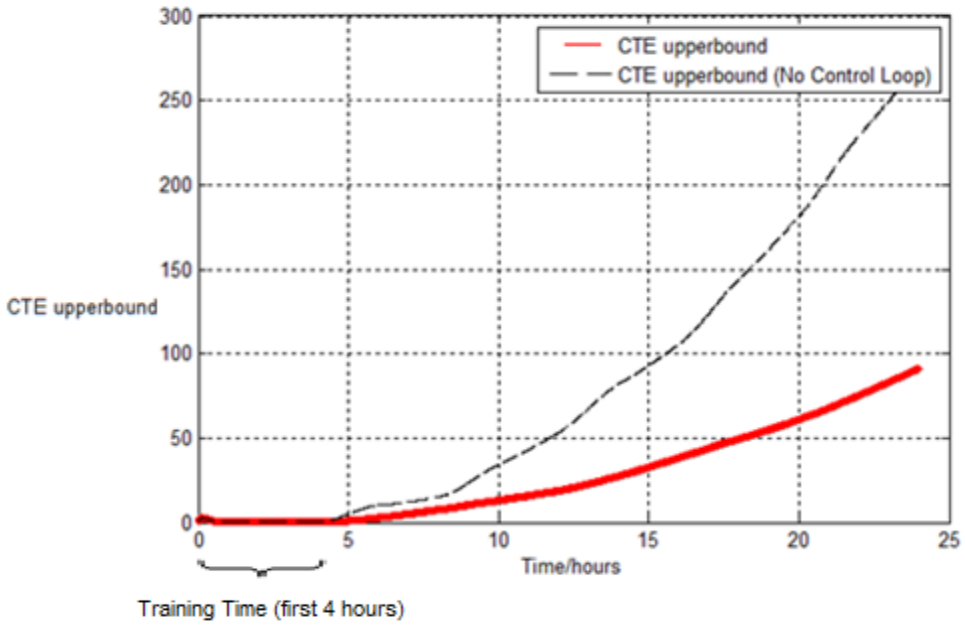

Figure 6.1 System Model Performance Comparisons with Training 4 Hours

The training time can be changed to further show the performance of the two system models. In Figure 6.2, the training time is set 10 hours and the holdover time is set 20 hours. The dashed line still represents the maximum CTE of using the system model of Equation (6.17). The solid line represents the maximum CTE of using the system model of Equation (6.14). The comparison again shows that the maximum CTE of using

the system model of Equation (6.17) is larger than using the system model of Equation (6.14). Both Figure 6.1 and Figure 6.2 show that including the control loop allows for the system model to result in a lower maximum CTE.

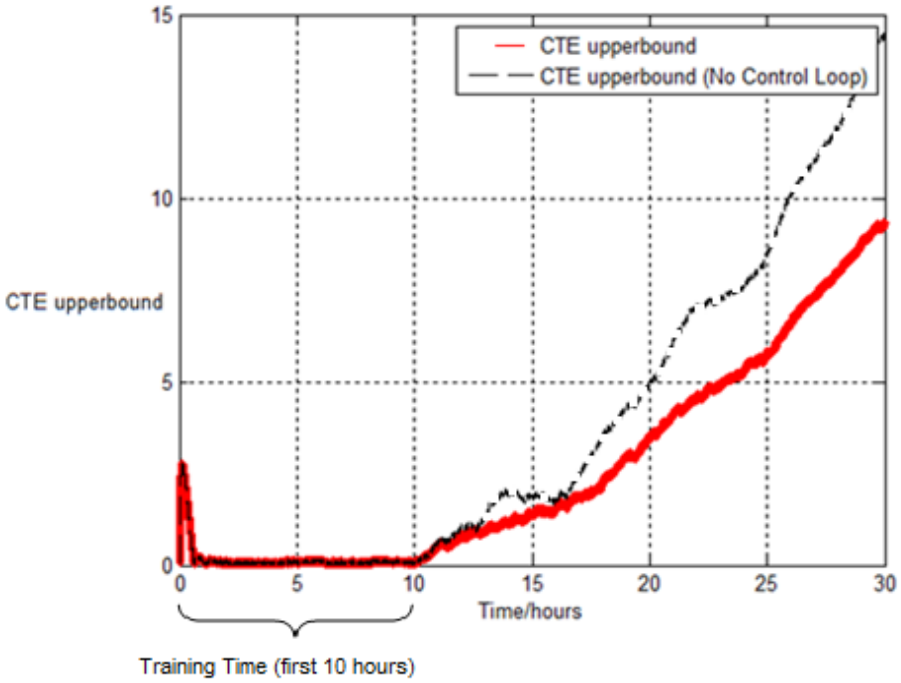

Figure 6.2 System Model Performance Comparisons with Training 10 Hours

We can notice that in Figure 6.2 there is a sharp spike when training process just begins. The reason is that a non-zero initial offset exists in the system model. Actually, this sharp spike also exists in Figure 6.1, but we cannot see it because the Y-axis scale of Figure 6.1 is much bigger than Figure 6.2.

The multi-parameter ARMAX model of Equation (6.14) is more complicated than the simple ARMAX model of Equation (5.25) for investigating the cumulative time error.

In Equation (5.25), the time error is determined by the variance of only one parameter estimate $\hat{a}$. In Equation (6.14), not only the variance of the first four parameter estimates, but also the covariance between them are important to determine the cumulative time error. The information of the variance of the parameters and the covariance between them are all contained in the covariance matrix of the parameter estimates vector $\hat{\theta}$. The joint distribution information and the corresponding confidence intervals of $\hat{\theta}$ are investigated in the next section.

6.2 Confidence Intervals of the Parameter Estimates

According to Equation (6.14), there are 5 parameters which need to be estimated. However, there are only 4 parameters in Equation (6.13). The parameter e in Equation (6.14) is the parameter for the noise, which does not relate to creating the correction signal. We only need the parameter estimates $\hat{a}, \hat{b}, \hat{c}$ and $\hat{d}$ to create the correction signal for compensating the oscillator. Hence, in the remainder of this chapter, we set $\theta_0 = [a \;\; b \;\; c \;\; d]^T$ and $\hat{\theta} = \left[\hat{a} \;\; \hat{b} \;\; \hat{c} \;\; \hat{d}\right]^T$. Through analyzing θ_0 and $\hat{\theta}$, the characteristics of the parameter estimates can be obtained.

According to [11], a four-dimensional parameter estimates vector $\hat{\theta}$ has a Gaussian distribution with mean value θ_0 and covariance matrix P_N.

$$\hat{\theta} \in N(\theta_0, P_N) \tag{6.18}$$

The covariance matrix P_N comes from $P(t)$ in Equation (2.34b). One should notice that $P(t)$ is a 5×5 matrix because the ARMAX model of Equation (6.14) has 5 parameters. Because we only need parameter estimates $\hat{a}, \hat{b}, \hat{c}$ and $\hat{d}$ to create the correction signal, only the first 4 rows and first 4 columns of $P(t)$ are needed to determine the distribution of $\hat{\theta}$. We set $P_{4\times4}(t)$ to equal the first 4 rows and first 4 columns of $P(t)$. The covariance matrix P_N is the product of $P_{4\times4}(t)$ and the variance of prediction error $\varepsilon(t)$ in Equation (2.31).

The parameter θ_0 is unknown and $\hat{\theta}$ is known after the system identification process. The distribution of $\hat{\theta} - \theta_0$ is given as,

$$\hat{\theta} - \theta_0 \in N(0, P_N) \tag{6.19}$$

For the i^{th} component of θ_0, the distribution is,

$$\hat{a} - a \in N(0, P_N^{(11)}) \tag{6.20}$$

$$\hat{b} - b \in N(0, P_N^{(22)}) \tag{6.21}$$

$$\hat{c} - c \in N(0, P_N^{(33)}) \tag{6.22}$$

$$\hat{d} - d \in N(0, P_N^{(44)}) \tag{6.23}$$

The term $P_N^{(ii)}$ indicates the i^{th} diagonal element of the covariance matrix P_N. Thus the probability distribution by which $\theta_0^{(i)}$ deviates from $\hat{\theta}^{(i)}$ can be calculated from standard statistical tables. Since P_N is the covariance matrix of the joint distribution of the parameter estimates vector $\hat{\theta}$, the covariance and correlation between the different components of $\hat{\theta}$ are obtained. We know that

$$\left(\hat{\theta} - \theta_0\right)^T P_N^{-1} (\hat{\theta} - \theta_0) \in \chi^2(d) \tag{6.24}$$

Equation (6.24) is a direct application of the definition of the χ^2 distribution. The probability of $|\hat{\theta} - \theta_0|^2_{P_N^{-1}}$ can be represented by $P(\left|\hat{\theta} - \theta_0\right|^2_{P_N^{-1}})$. Hence,

$$P(\left|\hat{\theta} - \theta_0\right|^2_{P_N^{-1}}) = P((\hat{\theta} - \theta_0)^T P_N^{-1} (\hat{\theta} - \theta_0)) \geq \alpha \tag{6.25}$$

is $\chi_\alpha^2(d)$ at the α level of the $\chi^2(d)$ distribution [11]. Equation (6.25) defines the confidence ellipsoids in R^d. The shape of the ellipsoid is determined by P_N. Figure 6.3 shows the confidence ellipsoid of $(\hat{\theta} - \theta_0)$ in the two-dimensional space, which is an ellipse.

In Figure 6.3, the shaded area of the ellipse is determined by a constant from the α level of the $\chi^2(d)$ distribution.

$$|\hat{\theta} - \theta_0|^2_{P_N^{-1}} \leq constant \tag{6.26}$$

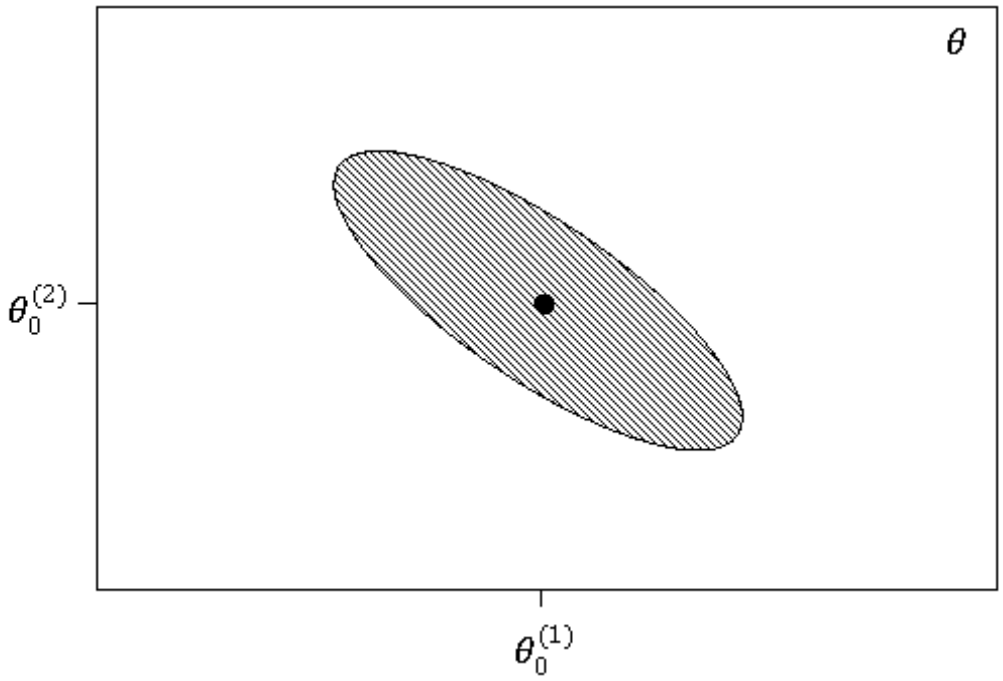

Figure 6.3 Confidence Ellipsoid for the Joint Gaussian Distribution [11]

This constant can be obtained from the χ^2 statistics table. For example, when the degrees of freedom (df) of the χ^2distribution is 4, and α is 5%, this constant is 9.49 [23]. The degrees of freedom are the number of components in the vector $\hat{\theta}$, which are 4 for the models in this chapter. The α represents the probability that $\hat{\theta}$ is outside the shaded area. In this thesis, the constant is 9.49 for a 95% probability value in the shaded area, and the confidence ellipsoid is defined in R^4.

6.3 The Eigenvector Method for Obtaining the Upperbound of the CTE

Referring to Equation (6.1), the estimate of the frequency stability is calculated through Equation (6.27).

$$\hat{y}(t) = \hat{a} \cdot x^2(t) + \hat{b} \cdot x(t) + \hat{c} + \hat{d} \cdot t \tag{6.27}$$

The term $x(t)$ represents the temperature. The parameters $\hat{a}, \hat{b}, \hat{c}$ and $\hat{d}$ represent the parameter estimates. The term $y(t)$ is used to denote the oscillator frequency stability. Thus, the cumulative time error can be obtained from Equation (6.28).

$$CTE = |\sum_{i=1}^{N}\left(\hat{y}(t_i) - y(t_i)\right)| \tag{6.28}$$

The 95% probability upperbound of the cumulative time error is the maximum value of CTE calculated from Equation (6.28) subject to Equation (6.26) when the constant in Equation (6.26) is 9.49.

$$\max_{\hat{\theta}} \left|\sum_{i=1}^{N}\left(\hat{y}(t_i) - y(t_i)\right)\right| \ \ such\ that\ \left(\hat{\theta} - \theta_0\right)^T P_N^{-1}(\hat{\theta} - \theta_0) \leq 9.49 \tag{6.29}$$

An eigenvector method is used to solve Equation (6.29). First, the column vectors $Z = \hat{\theta} - \theta_0$ and $R = [\sum_{i=1}^{N} x^2(t_i) \ \sum_{i=1}^{N} x(t_i) \ N \ \sum_{i=1}^{N} t_i]^T$ are defined. The problem of finding the maximum value of $\left|\sum_{i=1}^{N}\left(\hat{y}(t_i) - y(t_i)\right)\right|$ is equivalent to Equation (6.30).

$$\max_{\hat{\theta}}\left(\sum_{i=1}^{N}\left(\hat{y}(t_i) - y(t_i)\right)\right)^2 = max(Z^T * R)^2 = max(Z^T * (R * R^T) * Z)$$
$$such\ that\ Z^T P_N^{-1} Z \leq 9.49 \qquad (6.30)$$

Potter's Square root algorithm introduced in Chapter 2 can be used in RPEM to guarantee that P_N is invertible. We make $PI = P_N^{-1}$. The generalized eigenvalue problem of $R * R^T$ can be solved by Equation (6.31).

$$R * R^T * V = PI * V * D \qquad (6.31)$$

D is a diagonal matrix with the generalized eigenvalues of $R * R^T$ on the main diagonal. *V* is a full matrix whose columns are the corresponding eigenvectors of *D*. The value of *V* and *D* can be solved through matrix computation.

The maximum value of the elements on *D*'s main diagonal can be found, which is denoted h, and the corresponding index is denoted k. Now let v_k denote the *k-th* column of *V*, which corresponds to the maximum eigenvalue *h*. From Equation (6.31),

$$R * R^T * v_k = h * PI * v_k \tag{6.32}$$

Equation (6.32) multiplied by ${v_k}^T$ on the left side gives

$${v_k}^T * R * R^T * v_k = {v_k}^T * h * PI * v_k \tag{6.33}$$

Z is calculated as follows:

$$Z = \sqrt{\frac{9.49}{{v_k}' * PI * v_k}} * v_k \tag{6.34}$$

Equation (6.34) guarantees that $Z^T * PI * Z = 9.49$, because θ_0 which creates the maximum time error must be on the border of the ellipsoid. Thus,

$$\begin{aligned} max\left(\sum_{i=1}^{N}\left(\hat{y}(t_i) - y(t_i)\right)\right)^2 &= max(Z^T * R * R^T * Z) \\ &= \sqrt{\frac{9.49}{{v_k}' * PI * v_k}} * {v_k}' * R * R' * \sqrt{\frac{9.49}{{v_k}' * PI * v_k}} * v_k \\ &= \frac{9.49}{{v_k}' * PI * v_k} * {v_k}' * R * R' * v_k \\ &= \frac{9.49}{{v_k}' * PI * v_k} * {v_k}' * PI * v_k * h = 9.49 * h \end{aligned} \tag{6.35}$$

The maximum $|CTE|^2 = \left|\sum_{i=1}^{N}\left(\hat{y}(t_i) - y(t_i)\right)\right|^2$ is computed from Equation (6.35). Therefore, the maximum cumulative time error can be obtained by extracting the root of the maximum $|CTE|^2$ when $\hat{\theta}$ is located on the 95% probability confidence ellipsoid

boundary. Hence, this maximum CTE can be called the analytical 95% probability CTE upperbound.

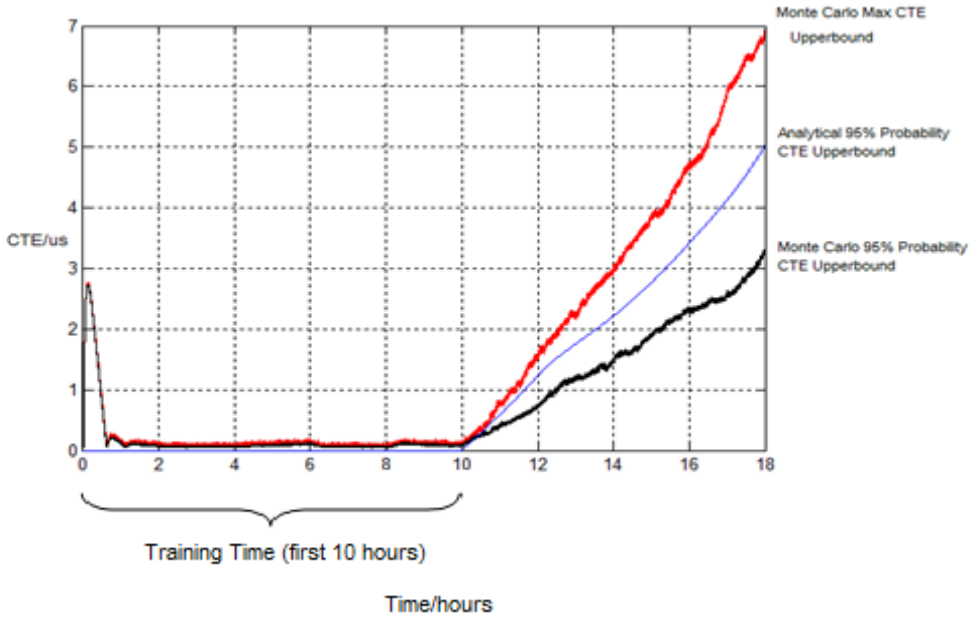

Figure 6.4 Comparison Result between Analytical CTE Upperbound and Monte Carlo CTE Upperbound

In order to verify the analytical result, 100 independent Monte Carlo simulations are run. Training time is set 10 hours and holdover time is set 8 hours. We use the 5^{th} maximum CTE of 100 simulations to represent the Monte Carlo 95% probability CTE upperbound and the maximum CTE of 100 simulations to represent the Monte Carlo maximum CTE upperbound. Figure 6.4 compares the 95% probability analytical CTE upperbound, Monte Carlo maximum CTE upperbound, and Monte Carlo 95% probability CTE upperbound. We can notice that there is a sharp spike when training process starts.

The reason is that a non-zero initial offset exists in the system model, which is the same as Figure 6.2.

The Monte Carlo 95% probability CTE upperbound is less than the analytical upperbound. The analytical upperbound of the CTE actually lies between the maximum CTE and the 95% upperbound of CTE computed from 100 Monte Carlo simulations. When the training time and the holdover time are changed, this result still holds. The reason is that the four parameters system identification is different from the one parameter system identification. A $\hat{\theta}$ which is located outside the 95% probability confidence ellipsoid does not always result in a larger CTE than all $\hat{\theta}$ in the 95% probability confidence ellipsoid. A simple example can be used to illustrate this.

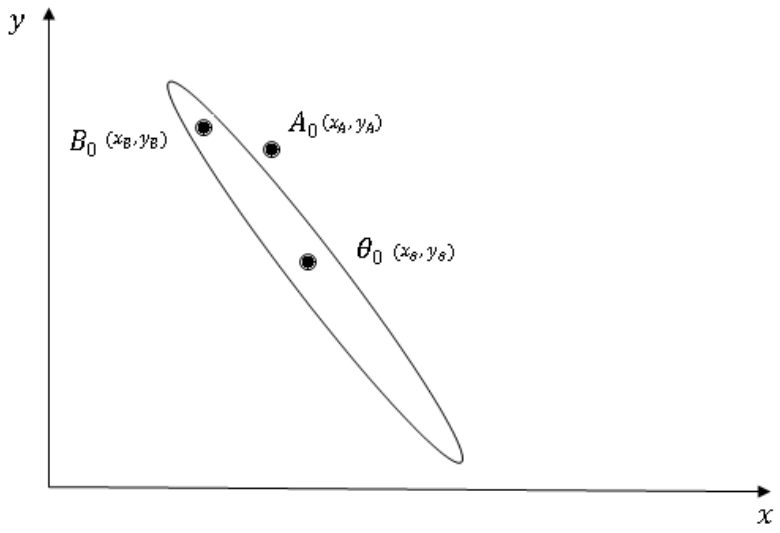

Figure 6.5 Simple Example for Illustrating a Problem of Determining CTE Upperbound for Multi-parameter System Model

$$z(t) = x_\theta \cdot t + y_\theta + v(t) \tag{6.36}$$

Figure 6.5 shows a two-parameter system confidence ellipse and Equation (6.36) shows the corresponding system model. The parameter θ_0 represents the true parameter value. We set

$$\theta_0 = [x_\theta \ y_\theta]^T = [1 \ \ 1]^T \tag{6.37}$$

The parameter estimate vector is represented by $\hat{\theta} = [\hat{x} \ \ \hat{y}]^T$. According to Equation (6.19), we know

$$\hat{\theta} - \theta_0 \in N(0, P_N) \tag{6.38}$$

$\hat{\theta} - \theta_0$ obeys a χ^2 distribution. P_N is the covariance matrix of $\hat{\theta} - \theta_0$. The shape of the ellipse is determined by P_N. We set $P_N = \begin{pmatrix} 0.0001 & -0.004 \\ -0.004 & 0.1938 \end{pmatrix}$. From the χ^2 statistics table, for the two-parameter system, $\left(\hat{\theta} - \theta_0\right)^T P_N^{-1}(\hat{\theta} - \theta_0)$ is less than 5.99 with the probability 95%. In Figure 6.5, the ellipse represents the 95% probability border.

Any point outside the ellipse results in

$$\left(\hat{\theta} - \theta_0\right)^T P_N^{-1}\left(\hat{\theta} - \theta_0\right) > 5.99 \tag{6.39}$$

Any point inside the ellipse results in

$$\left(\hat{\theta} - \theta_0\right)^T P_N^{-1}\left(\hat{\theta} - \theta_0\right) < 5.99 \tag{6.40}$$

The term A_0 is one parameter estimate. We set

$$A_0 = [x_A \ \ y_A]^T = [0.9861 \ 1.9332]^T \tag{6.41}$$

A_0 is a point outside the confidence ellipse because

$$(A_0 - \theta_0)^T P_N^{-1}(A_0 - \theta_0) = 6.14 > 5.99 \tag{6.42}$$

The term B_0 is another parameter estimate. We set

$$B_0 = [x_B \;\; y_B]^T = [0.98 \;\; 1.98]^T \tag{6.43}$$

B_0 is a point inside the confidence ellipse because

$$(B_0 - \theta_0)^T P_N^{-1} (B_0 - \theta_0) = 4.96 < 5.99 \tag{6.44}$$

The absolute value of the cumulative time error is

$$|CTE| = |(x_\theta - \hat{x})t + (y_\theta - \hat{y})| \tag{6.45}$$

In this example, when $t < 7.67$ or $t > 56.44$, the cumulative time error of A_0 is less than the CTE of B_0. Hence, in some situations, the CTE of some point inside the 95% probability confidence ellipsoid is larger than the CTE of the point outside the confidence ellipsoid.

Based on the analysis above, the analytical upperbound of the cumulative time error for $\hat{\theta}$ based on the 95% probability confidence ellipsoid is not the 95% upperbound of the CTE for the system model. It is to be expected that this CTE is larger than the 95% CTE upperbound and less than the maximum CTE. This is different from the single parameter system identification. In the single parameter system model, any $\hat{\theta}$ outside the 95% confidence interval must have a larger CTE than the parameter estimate within the 95% confidence interval.

6.4 Discussion

In Section 6.1, a refined system model is introduced. In this model, the oscillator frequency stability is affected by both temperature and ageing. Temperature has a quadratic relation with the frequency stability. Ageing has a linear relation with the frequency stability. The initial frequency stability offset is also included in the model. The model has 4 parameters that need to be identified. Similar to Chapter 5, the model reflects the effect of the control loop. The model is also transformed appropriately to apply RPEM. The performance of applying this refined model including the control loop is compared with the model without including the control loop. Simulation results show that the model including the control loop has a better performance.

In Section 6.2, the confidence interval for multi-parameter estimates is presented. One can see that the confidence interval for multi-parameter estimates is different from the single parameter estimate. The confidence interval for the single parameter estimate is a section of a line. The confidence interval for multi-parameter estimates is an ellipsoid in R^d. The term R^d means d-dimensional space. In this chapter, d is four, so a confidence ellipsoid in R^4 is investigated.

In Section 6.3, an eigenvector method is presented. This method can obtain the analytical upperbound of the cumulative time error. Simulations of 100 Monte Carlo runs are used to obtain the max CTE and 95% probability max CTE. Figure 6.4 shows that the analytical upperbound of the cumulative time error lies between the max CTE and 95%

probability max CTE. Then a simple example is presented to explain the result shown by Figure 6.5. This eigenvector method can only obtain an approximate CTE upperbound. If the requirement for the analytical CTE upperbound is not strict, this method is suitable. Otherwise, other more accurate methods for analytically deriving the CTE upperbound of the oscillator need to be developed.

Chapter 7: Conclusions and Future Work

7.1 Conclusions

In this thesis, a new adaptive OCXO frequency drift correction algorithm is proposed. This algorithm can enhance the accuracy of the OCXO. The enhanced OCXO can replace the more expensive DOCXO in the WiMAX and CDMA base transceiver stations. An ultra-low-cost base station timing module can be created by using this enhanced OCXO. The recursive system identification method is used to develop the adaptive correction algorithm. The recursive system identification method replaces the previous Batch Least Squares method. The new adaptive algorithm shows significant improvement for the cumulative time error of the timing module. The adaptive algorithm can also provide the CTE upperbound of the OCXO. The CTE upperbound is an important parameter which can determine the range of applications of the enhanced OCXO. The timing module system in base stations and the digital control loop, which is the core of the timing module, are reviewed for describing the adaptive algorithm.

Some system identification methods are reviewed. These methods are used to develop the adaptive algorithm. Two different system identification methods are evaluated. They are the recursive least squares (RLS) methods and the Kalman Filter method. The characteristics and performance of the methods are investigated. Finally, the

RLS method is chosen to develop the algorithm.

The OCXO frequency stability model is created to develop the adaptive algorithm and calculate the CTE upperbound of the OCXO. First, a simple model is used in Chapter 4. The temperature is linear related to the OCXO frequency stability in this simple model and the digital control loop through which the correction signal is created has not been taken into account. Figure 5.1 shows that the analytical CTE upperbound computed from this simple model is not close to the Monte Carlo CTE upperbound. The CTE upperbound cannot be predicted correctly because the prediction error in this model is biased. The digital control loop is included in the model in Chapter 5 to guarantee that the prediction error is unbiased. From simulation results, the CTE upperbound can be obtained very precisely by including the control loop in the model. The performance of using the system model including the control loop is compared with the model without including the control loop. Simulation results show that the performances of both models are close.

Then, a refined model structure is created in Chapter 6. This high-level model includes the temperature effect, the ageing effect and the initial frequency offset. The temperature is quadratic related to the OCXO satiability and the digital control loop is still included. The confidence ellipsoid of the parameter vector estimation is used to determine the distribution of the parameter estimates. An eigenvector method is developed for obtaining the CTE upperbound of the oscillator based on the confidence ellipsoid. The performance of using the high-level system model including the control

loop is also compared with the high-level model without including the control loop. Simulation results show that the performance of the model including the control loop is better than the model without including the control loop.

7.2 Future Work

The contents in the last section are what are covered in this thesis. It still leaves some problems as areas of future research.

- First, the 95% probability CTE upperbound for the detailed model is not precise enough, which is verified by the Monte Carlo method. A bound larger than the 95% Monte Carlo CTE upperbound is obtained because of the nature of the parameter vector estimates distribution. In some precise application, a more accurate upperbound of the cumulative time error may be needed.
- Second, the Monte Carlo method is used to verify the simulation results of the adaptive frequency drift correction algorithm in the research. Actually, the Monte Carlo method itself can be used to obtain the upperbound of the cumulative time error. After the timing module in the base station enters into the holdover mode, the analytical results of the temperature sensitivity and the ageing rate of the OCXO can be obtained. These data can be used to create a system model and simulate the training mode and the holdover mode by using

the same temperature and time profile as the real environment. By applying a sufficient number of Monte Carlo simulations, the upperbound of the CTE with the required precision can be obtained.

- Third, all work in this thesis is conducted in a simulation environment. Matlab is used to simulate the oscillator stability, the temperature and the ageing, etc. In future research, actual hardware and instruments are needed to verify the simulation results.
- Fourth, the adaptive OCXO frequency drift correction algorithm is only the first step in the research. The timing module in the base station is working within a large network. In the training mode, the timing module is locked by the satellite timing signal. When the lock is interrupted, the timing module is enhanced by the adaptive correction algorithm and provides a time reference to other clocks which cannot be locked by the satellite timing signal in the wireless network. The timing module enhanced by the algorithm serves as the standard timing reference. Transmitting the timing signal in the wireless network introduces signal delay and cause errors for the clocks. Solving the timing signal transmission problem in the wireless network is the next step in the research.

References

[1] XO5120 specification on OCXOs products list from http://www.mtronpti.com/pdf/xo5120_datasheet_exp21908.pdf (accessed October 5, 2009).

[2] L. Ljung, “Perspectives on System Identification”, Proceedings of 17th IFAC World Congress, pp. 7172-7184, July, 2008.

[3] “Recommended Minimum Performance Standards for cdma 2000 Spread Spectrum Base Stations,” 3GPP2 Standard C.S0010-C, release C, version 2, release data February 24, 2006.

[4] J. Vig and A. Ballato, “Ultrasonic Instruments and Devices”, Academic Press. pp. 637–701 (Chapter 7: Frequency Control Devices) 1999. ISBN-10: 0125319517

[5] J. Vig, “Quartz Crystal Resonators and Oscillators for Frequency Control and Timing Applications - A Tutorial”, 2004 IEEE International Frequency Control Symposium Tutorials, May 2004.

[6] W. Zhou, “Time, Frequency Measurement and Control Technology”, Xidian University Press, 2006. ISBN-10: 7560616720

[7] L. Ljung and T. Söderström, "Theory and Practice of Recursive Identification", MIT Press, 1983. ISBN-10: 026212095X

[8] F. Euler and N.F. Yannoni, “Frequency Retrace of Quartz Oscillators”,

Proceedings of the 35th Annual Frequency Control Symposium, pp. 492-520, 1981.

[9] T. Pialis and K. Phang, “Analysis of Timing Jitter in Ring Oscillators Due to Power Supply Noise”, Circuits and Systems, ISCAS’03, Proceedings of the 2003 International Symposium, pp. I-685 – I-688, 2003.

[10] J. Vig, “Introduction to Quartz Frequency Standards” Technical Report SLCET-TR-92-1, Army Research Laboratory, Electronics and Power Sources Directorate, 1992. [Online], available: http://www.ieee-uffc.org/frequency_control/teaching.asp?name=vigtoc (accessed October 5, 2009).

[11] L. Ljung, “System Identification: Theory for the User”, Prentice Hall PTR, 1999. ISBN-10: 0136566952

[12] R.E. Kalman, “A New Approach to Linear Filtering and Prediction Problems,” Transactions of the ASME Journal of Basic Engineering, vol. 82, pp. 35-45, 1960.

[13] B.M. Penrod, “Adaptive Temperature Compensation of GPS Disciplined Quartz and Rubidium Oscillators”, Proceedings of the IEEE International Frequency Control Symposium, pp. 980 – 987, 1996.

[14] S.R. Stein, “Kalman Filter Analysis of Precision Clocks with Real-Time Parameter Estimation”, Proceedings of the 43rd Annual Symposium on Frequency Control, pp. 232-236, 1989.

[15] W. Su and R.L. Filler, “A New Approach to Clock Modeling and Kalman Filter Time and Frequency Prediction”, Proceedings of the IEEE International Frequency

Control Symposium, pp. 331-334, 1993.

[16] C.W.T. Nicholls and G.C. Carleton, “Adaptive OCXO Drift Correction Algorithm”, Proceedings of the IEEE International Frequency Control Symposium and Exposition, pp. 509 – 517, 2004.

[17] C.W.T. Nicholls and P. Wu, “Ultra Low Cost Base Station Timing Module”, Proceedings of the IEEE International Frequency Control Symposium and Exposition, pp. 1318-1323, 2007.

[18] W.H. Press, “Numerical Recipes in C: The Art of Scientific Computing”, Cambridge University Press, pp. 657-661, 1992. ISBN-10: 0521431085

[19] J.M. Powers, “Method of Least Squares”, University of Notre Dame, February 2003, unpublished. [Online], available: http://www.nd.edu/~powers/ame.332/leastsquare/leastsquare.pdf (accessed October 5, 2009).

[20] L. Cao and H.M. Schwartz, “Analysis of the Kalman Filter Based Estimation Algorithm: An Orthogonal Decomposition Approach”, Automatica, vol. 40(1), pp. 5-19, 2004.

[21] M. Evestedt and A. Medvedev, “Stationary Behavior of an Anti-Windup Scheme for Recursive Parameter Estimation under Lack of Excitation”, Automatica, vol. 42, pp 151-157, January, 2006

[22] I. Manno, “Introduction to the Monte Carlo Method”, Akademiai Kiado Press, 1999. ISBN-10: 9630576155

[23] χ^2 statistics table from http://home.comcast.net/~sharov/PopEcol/tables/chisq.html (accessed October 5, 2009).