Robustification and Optimization in Repetitive Control For Minimum Phase and Non-Minimum Phase Systems

Pitcha Prasitmeeboon

Submitted in partial fulfillment of the
requirements for the degree
of Doctor of Philosophy
in the Graduate School of Arts and Sciences

COLUMBIA UNIVERSITY
2017

ABSTRACT

Robustification and Optimization in Repetitive Control For Minimum Phase and Non-Minimum Phase Systems

Pitcha Prasitmeeboon

Repetitive control (RC) is a control method that specifically aims to converge to zero tracking error of a control systems that execute a periodic command or have periodic disturbances of known period. It uses the error of one period back to adjust the command in the present period. In theory, RC can completely eliminate periodic disturbance effects. RC has applications in many fields such as high-precision manufacturing in robotics, computer disk drives, and active vibration isolation in spacecraft.

The first topic treated in this dissertation develops several simple RC design methods that are somewhat analogous to PID controller design in classical control. From the early days of digital control, emulation methods were developed based on a Forward Rule, a Backward Rule, Tustin's Formula, a modification using prewarping, and a pole-zero mapping method. These allowed one to convert a candidate controller design to discrete time in a simple way. We investigate to what extent they can be used to simplify RC design. A particular design is developed from modification of the pole-zero mapping rules, which is simple and sheds light on the robustness of repetitive control designs.

RC convergence requires less than 90 degree model phase error at all frequencies up to Nyquist. A zero-phase cutoff filter is normally used to robustify to high frequency model error when this limit is exceeded. The result is stabilization at the expense of failure to cancel errors

above the cutoff. The second topic investigates a series of methods to use data to make real time updates of the frequency response model, allowing one to increase or eliminate the frequency cutoff. These include the use of a moving window employing a recursive discrete Fourier transform (DFT), and use of a real time projection algorithm from adaptive control for each frequency. The results can be used directly to make repetitive control corrections that cancel each error frequency, or they can be used to update a repetitive control FIR compensator. The aim is to reduce the final error level by using real time frequency response model updates to successively increase the cutoff frequency, each time creating the improved model needed to produce convergence zero error up to the higher cutoff.

Non-minimum phase systems present a difficult design challenge to the sister field of Iterative Learning Control. The third topic investigates to what extent the same challenges appear in RC. One challenge is that the intrinsic non-minimum phase zero mapped from continuous time is close to the pole of repetitive controller at +1 creating behavior similar to pole-zero cancellation. The near pole-zero cancellation causes slow learning at DC and low frequencies. The Min-Max cost function over the learning rate is presented. The Min-Max can be reformulated as a Quadratically Constrained Linear Programming problem. This approach is shown to be an RC design approach that addresses the main challenge of non-minimum phase systems to have a reasonable learning rate at DC.

Although it was illustrated that using the Min-Max objective improves learning at DC and low frequencies compared to other designs, the method requires model accuracy at high frequencies. In the real world, models usually have error at high frequencies. The fourth topic addresses how one can merge the quadratic penalty to the Min-Max cost function to increase robustness at high frequencies. The topic also considers limiting the Min-Max optimization to

some frequencies interval and applying an FIR zero-phase low-pass filter to cutoff the learning for frequencies above that interval.

Table of Contents

Acknowledgements

I would like to take this opportunity to express my gratitude to all those who helped me to complete this thesis.

I would like to express my appreciation and sincere gratitude to my advisor Professor Richard Longman for his support, inspiration and motivation. He always patiently encourages and provides guidance through the research with his dedication and knowledge. His advice on both academia and life experience have been priceless. I would not be able to accomplish this dissertation without him.

My sincere thanks also goes to my co-advisor Professor Dan Ellis for his guidance and help. I thank the member of my doctoral committee.

I thank my colleagues, Henry Yau, Bing Song, Jianzhong Zhu, Te Li, Xiaoqiang Ji, and Francesco Vicario, for sharing great times in the lab and wonderful times during our road trips to conferences. I feel thankful for their help and support during hard times.

I would like to thank the Royal Thai Government for the financial support throughout my study.

I thank Nukul Prawatyotin and Sam Prawatyotin for their love, kindness and generosity. They always make me feel like home during the time I stay in NYC.

Last but not the least, I would like to thank my family, my parents and my brothers, for their unconditional love, support, and encouragement at all times.

Chapter 1

Introduction

1.1 Background

Repetitive Control (RC) is a relatively new field in control that aims to converge to zero error for systems that execute a periodic command, or have a periodic disturbance, or both with the same period. RC is specifically designed to take advantage of the information that there is a known period of the periodic command and periodic disturbance. The repetitive control law can be implemented as an extra loop around an existing feedback control system. Each time step, it examines the error in the previous period and uses this information to update the command to the feedback control system. It aims to converge to that command that eliminates the error produced by the disturbance, or that command that produces the desired output. Unlike other methods that do not specifically use knowledge of the period in such applications, it is theoretically possible for repetitive control to completely eliminate the effects of a periodic disturbance, and a deterministic tracking error in following a periodic command. Repetitive control was initially developed to eliminate residual 60Hz related errors in physics particle accelerators [1-2]. Other early works in the field are References [3-10]. Reference [11] gives an overview of the methods for designing repetitive control systems recommended by the control research group at Columbia University and Reference [12] presents various earlier design methods.

The simplest form of RC uses the following thinking. If at some point during the last period, the output was 2 units too small, then add 2 units (or 2 units times a repetitive control gain) to the command at the appropriate point during the current period. A particularly simple RC design method (and a similar iterative learning control method) is generated in Reference [12]. An analogy is made with PID control design in classical control systems where one tunes three gains, and one can do the tuning in hardware without involving a model. The approach in Reference [12], again involves tuning only three parameters, and again they can be tuned in hardware. A linear phase lead is used as a compensator for the phase lag of the system. The amount of phase lead is one parameter, and the overall gain is a second parameter. A smaller gain allows a higher cutoff frequency. Experimentally observing the frequency content of the error signals as iterations or periods progress allows one to tune these two parameters to produce convergence to the highest possible frequency for this compensator, and then one uses a zero-phase low-pass filter to cut off the learning above this point. This simple design process usually allows one to substantially improve the performance of exciting feedback control systems that repetitively perform the same task. The simplicity makes it possible to create a self tuning version as discussed in References [13-14].

If the transfer function of the feedback control system from command to output were unity, then the simplest RC law discussed above could immediately eliminate the error in the second period in a deadbeat fashion (or it can eliminate a given fraction of the error if the gain is not unity). This suggests the use of a compensator that is the inverse of the feedback control system transfer function. However, this is nearly always impossible to use in practice because the inverse of a discrete time transfer function is nearly always unstable. Instead, what is done in Reference [15] is to design a compensator whose frequency response aims to be the same as the inverse of

the frequency response of the feedback control system. In this way cancellation of the system dynamics is accomplished after transients have become negligible. The recommended design method is based on a cost function fit of the frequency response using a finite impulse response filter (FIR) as presented in Reference [15]. This filter can also be thought of as a compensator that the RC designer must create.

For stability one needs reasonably good knowledge of the system dynamics, and to mimic this inverse of the frequency response of the dynamics with the compensator, up to Nyquist frequency. In practice, one also includes a zero-phase low-pass filter to cut off the learning at high frequencies, when the model and/or the compensator becomes too inaccurate for the learning process to converge [16-17]. The cutoff must be adjusted in the real world application, after observing the hardware performance and the frequency content of the error signals with time, because one does not know what is wrong with one's model.

1.2 Thesis Outline

This dissertation presents methods to simplify and robustify RC designs. The content of this dissertation contains four topics. The first two Chapters give approaches for application in minimum phase systems. Chapters 4 and 5 specifically shed light on the challenges of RC designs for non-minimum phase systems.

Chapter 2 investigates to what extent the emulation methods can be used for RC design. The emulation methods that are considered here are a Forward Rule, a Backward Rule, Tustin's Formula, a modification using prewarping, and a pole-zero mapping method. Reference [18] presents the emulation methods. It is shown that the first two methods are simple and can work when using an appropriate cutoff of the learning process, but the methods similar to PID in Reference [12] would be preferable. The other methods fail in this application. However, making

use of knowledge obtained in RC compensator design, the pole-zero mapping rules can be modified to produce a particularly simple and effective RC design method.

Chapter 3 proposes methods that address the problem when systems have unmodeled high frequency dynamics, often described as parasitic poles or residual modes. As a result, the real world model and the system model might be sufficiently different to create instability requiring a cutoff. This chapter develops several approaches that have an adaptive process to improve the performance while the system is running. The new methods perform real time updates of magnitude and phase in the RC law proposed by References [19-20], allowing higher cutoff frequency or an elimination of the cutoff.

The vast majority of repetitive control methods are designed for minimum phase systems. Non-minimum phase systems have unusual characteristic that complicates the feedback design for such systems. Chapter 4 investigates how effective the existing RC methods from References [15, 21, 22], perform on non-minimum phase systems and aims to find effective methods for non-minimum phase systems. The improved design of Taylor series approach originally developed by Reference [22] is proposed. However, the method requires that the information of pole and zero locations is known. A new method is proposed that minimizes the maximum of the error over all frequencies to Nyquist. This method can deal with the problem of slow learning rate at DC and low frequencies in non-minimum phase systems and only requires knowledge of system steady state frequency response.

Although the Min-Max approach addresses the most fundamental difficulty for RC of non-minimum phase systems, it introduces some extra difficulties. The purpose of Chapter 5 is to improve the performance of the design method and increase robustness to model errors. It proposes a new design that includes the optimization of quadratic cost into the Min-Max design to increase

robustness at high frequencies where models usually have error. Another solution to the problem that this dissertation considers is to optimize the Min-Max design up to some frequencies and apply the FIR zero-phase low-pass filter from References [16-17] to stop learning after cutoff. These two approaches can improve learning at low frequencies and become more robust at high frequencies. The materials presented in Chapters 2, 3, 4, and 5 also appear in References [23], [24], [25], and [26], respectively.

Finally, Chapter 6 presents the conclusions of the dissertation.

1.3 References

[1] T. Inoue et al., "High Accuracy Control Magnet Power Supply of Proton Synchrotron in Recurrent Operation," *The Trans. of the Institute of Electrical Engineering of Japan*, Vol. 100, 1980, pp. 234–240.

[2] T. Inoue, M. Nakano, and S. Iwai, "High Accuracy Control of a Proton Synchrotron Magnet Power Supply," *Proceedings of the 8th World Congress of IFAC*, 1981, pp. 216-221.

[3] T. Omata, M. Nakano, and T. Inoue, "Applications of Repetitive Control Method to Multivariable Systems," *Transactions of SICE*, Vol. 20, 1984, pp. 795-800.

[4] R. H. Middleton, G. C. Goodwin, and R. W. Longman, "A Method for Improving the Dynamic Accuracy of a Robot Performing a Repetitive Task," *International Journal of Robotics Research*, Vol. 8, 1989, pp. 67-74. Also, University of Newcastle, Newcastle, Australia, Department of Electrical Engineering Technical Report EE8546, 1985.

[5] K-K. Chew and M. Tomizuka, "Steady-State and Stochastic Performance of a Modified Discrete-Time Prototype Repetitive Controller," *ASME Journal of Dynamic Systems, Measurement and Control*, 1990, pp. 35-41.

[6] S. Hara, and Y. Yamamoto, "Synthesis of Repetitive Control Systems and its Applications," *Proceeding of the 24th IEEE Conference on Decision and Control*, 1985, pp. 326-327.

[7] M. Tomizuka, T.-C. Tsao, and K.-K. Chew, "Analysis and Synthesis of Discrete Time Repetitive Controllers," *Journal of Dynamic Systems, Measurement, and Control*, Vol. 111, 1989, pp. 353-358.

[8] T. C. Tsao and M. Tomizuka, "Robust Adaptive and Repetitive Digital Tracking Control and Application to a Hydraulic Servo for Non-Circular Machining," *Journal of Dynamic Systems, Measurement and Control*, Vol. 116, 1994, pp. 24-32.

[9] T. Omata, S. Hara, and M. Nakano, "Nonlinear Repetitive Control with Application to Trajectory Control of Manipulators," *J Robot Syst,* 1987, pp. 631–652.

[10] M-C. Tsai, G. Anwar, M. Tomizuka, "Discrete Time Repetitive Control for Robot Manipulators," *Proceedings of the 1988 IEEE International Conference on Robotics and Automation*, 1988, pp. 1341–1346.

[11] R. W. Longman, "On the Theory and Design of Linear Repetitive Control Systems," *European Journal of Control*, Special Section on Iterative Learning Control, Guest Editor Hyo-Sung Ahn, Vol. 16, No. 5, 2010, pp. 447-496.

[12] R. W. Longman, "Designing Iterative Learning and Repetitive Controllers," chapter in *Iterative Learning Control: Analysis, Design, Integration and Applications*, Bien and Xu editors, Kluwer Academic Publishers, Boston, 1998, pp. 107-146.

[13] R. W. Longman and S.-L. Wirkander, "Automated Tuning Concepts for Iterative Learning and Repetitive Control Laws," *Proceedings of the 37th IEEE Conference on Decision and Control*, Tampa, Florida, Dec. 1998, pp. 192-198.

[14] S.-L. Wirkander and R. W. Longman, "Limit Cycles for Improved Performance in Self-Tuning Learning Control," *Advances in the Astronautical Sciences*, Vol. 102, 1999, pp. 763-781.

[15] B. Panomruttanarug and R. W. Longman, "Repetitive Controller Design Using Optimization in the Frequency Domain," *Proceedings of the 2004 AIAA/AAS Astrodynamics Specialist Conference*, Providence, RI, August 2004.

[16] B. Panomruttanarug and R. W. Longman, "Frequency Based Optimal Design of FIR Zero-Phase Filters and Compensators for Robust Repetitive Control," *Advances in the Astronautical Sciences*, Vol. 123, 2006, pp. 219-238.

[17] J. Bao and R. W. Longman, "Enhancment of Repetitive Control using Specialized FIR Zero-Phase Filter Designs," *Advances in the Astronautical Sciences*, Vol. 129, 2008, pp. 1413-1432.

[18] G. F. Franklin, J. D. Powell, A. Emami-Naeini, *Feedback Control of Dynamic Systems*, Prentice Hall, 2009.

[19] H. Yau and R. W. Longman, "Frequeny Response Based Repetitive Control Design for Linear Systems with Periodic Coefficients." *Proceedings of the AIAA/AAS Astrodynamics Specialist Conference*, Vail, Co, August 2015.

[20] Y. Shi, R. W. Longman, and M. Nagashima, "Small Gain Stability Theory for Matched Basis Function Repetitive Control." *Acta Astronautica*, Vol. 95, 2014, pp. 260 -271.

[21] K-K. Chew and M. Tomizuka, "Steady-State and Stochastic Performance of a Modified Discrete-Time Prototype Repetitive Controller," 1988 ASME Winter Annual Meeting, December

1988, also in ASME *Journal of Dynamic Systems, Measurement and Control*, March 1990, pp. 35-41.

[22] K. Xu and R. W. Longman, "Use of Taylor Expansions of the Inverse Model to Design FIR Repetitive Controllers," *Advances in the Astronautical Sciences*, Vol. 134, 2009, pp. 1073-1088.

[23] P. Prasitmeeboon and R. W. Longman, "Investigation of Discrete Time Emulation Techniques to Simplify Repetitive Control Design," *Advances in the Astronautical Sciences*, Vol. 150, 2014, pp. 1941-1958.

[24] P. Prasitmeeboon and R. W. Longman, "Repetitive Control Using Real Time Frequency Response Updates for Robustness to Parasitic Poles," *Advances in the Astronautical Sciences*, Vol. 158, 2016, pp. 2259-2272.

[25] P. Prasitmeeboon and R. W. Longman, "Using Quadratically Constrained Quadratic Programming to Design Repetitive Controllers: Application to Non-Minimum Phase Systems," *Advances in the Astronautical Sciences*, Vol. 156, 2016, pp. 1647-1666.

[26] P. Prasitmeeboon and R. W. Longman, "Min-Max Merged with Quadratic Cost for Repetitive Control of Non-Minimum Phase Systems," *Advances in the Astronautical Sciences*, to appear.

Chapter 2

Investigation of Discrete Time Emulation Techniques to Simplify Repetitive Control

2.1 Introduction

Historically, in the 1950's and 1960's, classical control system designers have developed considerable expertise in the design of continuous time systems, and as discrete time digital control became a practical approach, people wanted some simple method of converting a design based on continuous time thinking to discrete time. Reference [1] presents a set of such simple conversion methods which we examine for use in the RC problem. The purpose of this chapter is to examine whether discrete time control emulation methods can be helpful in producing particularly simple design methods, i.e. can these methods be used to design the compensator needed in RC. Reference [1] has a particularly good treatment of emulation methods.

2.2 Repetitive Control Background

A repetitive control system can have the block diagram structure as shown in Figure 2-1. Generally, the feedback control system can be considered to be totally digital and given by the *z*-transfer function $G(z)$. The repetitive controller $R(z)$ adjusts the command $U(z)$ given to the feedback controller. The $Y_D(z)$ is the desired output, which is either a constant or is periodic with period p time steps. The $V(s)$ is any periodic disturbance considered to have a period measured in time steps equal to integer p steps. Reference [2] indicates what to do when the period is not an

integer number of steps. The purpose of the repetitive controller is to eliminate the influence of this disturbance, and/or eliminate error in following a command of the same period. Of course, the command could be a constant. The actual periodic disturbance can occur anywhere around the closed loop of the feedback control system, but it results in a periodic disturbance to the output, and we write it in terms of this equivalent disturbance. The discrete time equations only need the disturbance at the sample times and this sequence has z-transform $V(z)$.

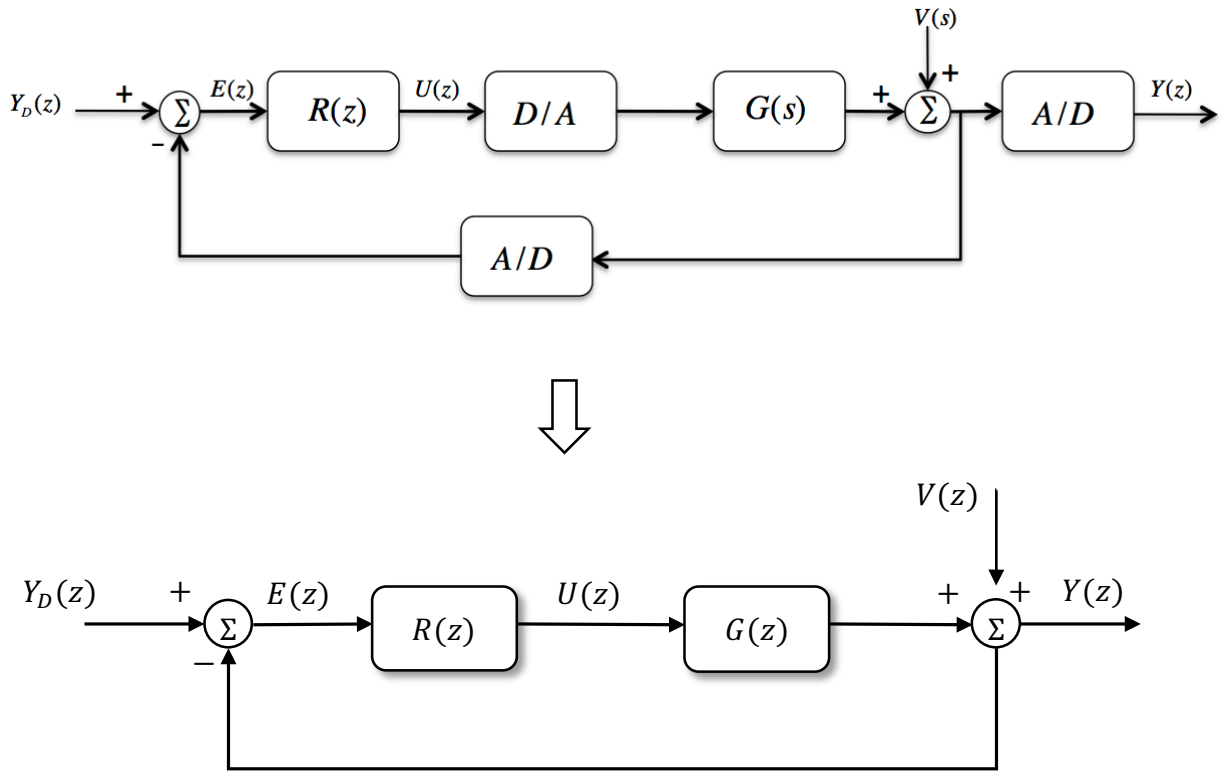

Figure 2-1. Block diagrams of a repetitive control system where the repetitive controller modifies the command to a feedback control system in continuous time and equivalent block diagram in discrete time.

The simplest form of repetitive control adjusts the command $u(kT)$ in the time domain according to

$$u(kT) = u\big((k-p)T\big) + \phi e\big((k-p+1)T\big) \tag{2-1}$$

where ϕ is a repetitive control gain, k is the time step, T is the sample time interval, and p is the number of time steps in a period. The command in the current period are adjusted based on error of one period back, adjusted for the assumed one-time step delay through the system. In the transform domain of (2-1) becomes $U(z) = [\phi z/(z^p - 1)]E(z)$.

The general repetitive control law has the form

$$R(z) = \frac{F(z)H(z)}{z^p - H(z)} \tag{2-2}$$

$$U(z) = z^{-p}H(z)[U(z) + F(z)E(z)] \tag{2-3}$$

In words, this says that the command used at the current time step is the command used in the previous period plus the error observed in the previous period after it has gone through the compensator $F(z)$. The $H(z)$ is a finite impulse response zero-phase low-pass filter designed to minimize a cost summed over a discrete set of frequencies between zero and Nyquist

$$H(z) = \sum_{k=-n}^{n} a_k\, z^k \tag{2-4}$$

$$J_H = \gamma \sum_{j=0}^{j_p} \big[1 - H\big(e^{i\omega_j T}\big)\big]\big[1 - H\big(e^{i\omega_j T}\big)\big]^* + \sum_{j=j_s}^{N-1} \big[H\big(e^{i\omega_j T}\big)\big]\big[H\big(e^{i\omega_j T}\big)\big]^* \tag{2-5}$$

The first term in the sum tries to make the FIR filter output look like unity for frequencies in the passband, and the second term tries to make it look like zero for frequencies in the stopband. See References [2-4], and note the possible subtleties discussed in References [2], [5], [6]. The low-pass filters considered in this dissertation is developed by Reference [4], and include an inequality constraint to prevent any amplification in the passband, which allows a higher cutoff.

From the block diagram one can write the difference equation governing the performance of the repetitive control system

$$[(z^p - H(z)) + \phi H(z)F(z)G(z)]E(z) = (z^p - H(z))[Y_D(z) - V(z)] \tag{2-6}$$

Note that if there is no frequency cutoff, then the right hand side is zero since the desired trajectory and the disturbance are periodic with period p time steps. In any case, stability is determined by the homogeneous difference equation which can be written as

$$z^p E(z) = \{H(z)[1 - \phi F(z)G(z)]\}E(z) \tag{2-7}$$

This form suggests that the expression in the curly brackets is a transfer function from one period to the next. Convert this to a frequency transfer function by setting $z = e^{i\omega T}$, and this suggests that if the magnitude of this frequency response function is less than unity for all frequencies from zero to Nyquist, then every frequency component of the error decays from one period to the next, and the error decays either to zero, or to the particular solution created by the forcing function when there is a cutoff frequency. This thinking is not rigorous, but one can rigorously prove that the conclusion of convergence is correct [2]. The repetitive control system of Figure 2-1 is asymptotically stable for all possible initial periods p, if and only if

$$\left|H\left(e^{i\omega T}\right)[1 - \phi F\left(e^{i\omega T}\right)G\left(e^{i\omega T}\right)]\right| < 1 \qquad \forall\omega \tag{2-8}$$

The aim is to find $F(z)$ whose frequency response is close to the reciprocal of the frequency response of $G(z)$ for all frequencies, making the square bracket term near zero. For high frequencies when our model error in the $G(z)$ used for design is too large, or our design of $F(z)$ is not accurate enough, the cutoff filter $H(z)$ is employed to satisfy this stability condition.

2.3 Emulation Methods

For simplicity of understanding, we now consider that the feedback control system $G(z)$ is a continuous time control system $G(s)$ fed by a zero order hold. As before the periodic disturbance

somewhere in this continuous time feedback loop can be converted to an equivalent periodic disturbance $V(z)$ to the output of the feedback control system at the sample times. Then our design objective is to use emulation methods to find a simple discrete time transfer function that emulates the continuous time behavior, and then invert it for the design of the RC compensator. We consider the following emulation methods from Reference [1].

(i) The Forward Rule: Replace *s* in the Laplace transfer function by $s = (z-1)/T$ to produce a discrete time *z*-transfer function. Let *T* be the sample time interval. The relationships are

$$s = \frac{z-1}{T} \qquad z = 1 + Ts \tag{2-9}$$

(ii) The Backward Rule:

$$s = \frac{z-1}{Tz} \qquad z = \frac{1}{1-Ts} \tag{2-10}$$

(iii) The Trapezoidal Rule, also known as Tustin's formula, or the *w*-plane (an example of a bilinear transformation):

$$s = \frac{2}{T}\frac{z-1}{z+1} \qquad z = \frac{1+Ts/2}{1-Ts/2} \tag{2-11}$$

This rule is sometimes modified by prewarping in order to place the half-power point at frequency ω_1 by modifying the formula for *s* to be

$$s = \left(\frac{\omega_1}{\tan(\omega_1 T/2)}\right)\frac{z-1}{z+1} \tag{2-12}$$

The conclusions made below will be the same with or without the use of prewarping.

(iv) Pole-Zero Mapping

(1) Given the Laplace transfer function, find the zeros and the poles (assumed to have more poles than zeros).

(2) For every factor $s - s_k$ of the denominator of $G(s)$, create a corresponding factor $z - z_k$ in the denominator of $G(z)$, where $z_k = \exp(s_k T)$.

(3) Do the same for every factor in the numerator.

(4) Introduce as many factors $z + 1$ into the numerator as needed, such that the highest power of z in the numerator is one less than the highest power in the denominator.

(5) Find the DC gain of $G(s)$ by setting $s = 0$. Find the current corresponding gain of the z-transfer function by setting $z = 1$. Then introduce the reciprocal of this gain as a constant multiplying the z-transfer function so that its DC gain now matches that of the original $G(s)$.

We will find that the first two methods, (i) and (ii), can be used provided that one has a sufficiently low cutoff frequency in the zero-phase low-pass filter. The third and fourth methods, (iii) and (iv), create a compensator that if infinite at Nyquist frequency. To make (iii) work, we create a new cutoff filter with the property that its output is zero at Nyquist frequency. But in implementation one might have difficulties ensuring that the filter zero times the compensator infinity produces zero or something smaller than unity. Hence, the approach is not recommended.

The pole zero mapping approach above from Reference [1] also produces an infinite compensator response at Nyquist, but we examine the pole-zero mapping rules in light of insight gained in RC design. From this we are able to create a new pole-zero mapping procedure that is simple to use and will produce a simple RC design that will nearly always be a stable design relative to the given model. The cutoff filter can stabilize in the presence of model error as always. These results are also of importance in that they shed light on the stability robustness properties associated with different aspects of RC compensator designs.

2.4 Pole-Zero Mapping by Emulation Method (iv)

First we examine the pole-zero mapping emulation method (iv) to examine its behavior. Then we will create a new related pole-zero mapping method to fix the difficulties. Throughout this chapter we consider a third order system

$$G(s) = \left(\frac{\alpha}{s+\alpha}\right)\left(\frac{\omega_o^2}{s^2 + 2\zeta\omega_o s + \omega_o^2}\right) \tag{2-13}$$

where α is equal to 8.8, ω_o is 37, and ζ is 0.5. We will also consider other transfer function, a 2nd order system that eliminates the one real root above, a 4th order system that squares the complex conjugate pair term, and a 5th order transfer function formed by squaring the second factor on the right including the real root. Directly applying the pole-zero emulation method (iv) above to the third order model produces the magnitude $\left|1 - F\left(e^{i\omega T}\right)G\left(e^{i\omega T}\right)\right|$ shown in Figure 2-2. Because rule (4) places two zeros at -1, the compensator $F\left(e^{i\omega T}\right)$ goes to infinity as ω goes to infinity, which guarantees that the system is unstable (for all periods p). The plot is truncated before reaching Nyquist frequency of 50 Hz. To make a RC system stable for all periods requires an $H(z)$ designed to cut off the learning process before this curve reaches unit, a bit above 30Hz. In other applications of emulations methods this issue might not arise, but having this plot tend to infinity is fatal in RC unless the cutoff filter can bring the infinity to below +1. Hence, the cutoff filter needs to have two zeros at -1 in order to be a candidate for design, and this constraint may adversely influence performance in the passband and stopband. This chapter develops a new pole-zero mapping method that avoids this issue, making use of more complete knowledge about the zeros introduced by discretization.

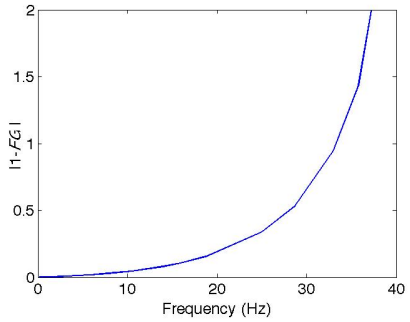

Figure 2-2. The magnitude $\left|1 - F(e^{i\omega T})G(e^{i\omega T})\right|$ using the pole-zero mapping (iv).

2.5 Precise Conversion — Continuous to Discrete

2.5.1 Converting Homogeneous Equations

The denominator of a Laplace transfer function is the characteristic polynomial of the associated differential equation. Similarly, the denominator of a *z*-transfer function is the characteristic polynomial of the associated scalar difference equation. The roots of these polynomials determine the solutions of the associated homogeneous equations.

Consider a differential equation and its characteristic equation in factored form

$$\frac{d^2y}{dt^2} + a_1\frac{dy}{dt} + a_2 y = 0 \qquad (2\text{-}14)$$

$$s^2 + a_1 s + a_2 = (s - s_1)(s - s_1) = 0$$

Then the general solution can be written in terms of continuous time t and at sample times kT as

$$y(t) = C_1 e^{s_1 t} + C_2 e^{s_2 t} \qquad y(kT) = C_1 (e^{s_1 T})^k + C_2 (e^{s_2 T})^k \qquad (2\text{-}15)$$

We can find a difference equation

$$y\big((k+2)T\big) + \alpha_1 y\big((k+1)T\big) + \alpha_2 y(kT) = 0 \qquad (2\text{-}16)$$

whose solution is the same at the sample times, by making the coefficients from the following polynomial

$$z^2 + \alpha_1 z + \alpha_2 = (z - e^{s_1 T})(z - e^{s_2 T}) = 0 \tag{2-17}$$

Thus the conversion from *s* to *z* for poles of Laplace and *z*-transfer functions satisfies the rule $z_j = e^{s_j T}$.

2.5.2 Exact Conversion for Systems Fed by a Zero Order Hold

When there is a forcing function, one must know what the input is doing between sample times. If the input is held constant, then one can convert transfer functions according to

$$G(z) = (1 - z^{-1}) Z\{G(s)/s\} \tag{2-18}$$

where the large *Z* indicates taking the *z*-transform of the samples of the time function specified by the argument, which is the unit step response of the continuous time system. In this formula it is specified by its Laplace transfer function.

Another approach to the conversion is to convert the scalar differential equation to state space form, and then convert the state space differential equation to the equivalent state space difference equation

$$\begin{gathered} \dot{x}(t) = A_c x(t) + B_c u(t) \quad ; \quad y(t) = Cx(t) \\ x\big((k+1)T\big) = A_d x(kT) + B_d u(kT) \quad ; \quad y(kT) = Cx(kT) \\ A_d = e^{A_c T} \quad ; \quad B_d = \int_0^T e^{A_c T} d\tau \, B_c = (A_d - I) A_c^{-1} B_c \end{gathered} \tag{2-19}$$

Taking the *z*-transform of this state space difference equation produces the transfer function and the scalar difference equation in the forms

$$\begin{gathered} Y(z) = [C(zI - A_d)^{-1} B_d] U(z) \\ |zI - A_d| Y(z) = C adj(zI - A_d)(A_d - I) A_c^{-1} B_c U(z) \end{gathered} \tag{2-20}$$

The *adj* indicates the adjoint matrix. When using state space equations, it is not obvious how to observe the zeros of the transfer function. In controllable canonical form or observable canonical

form, the coefficients of the zero polynomial can be found in the input or output matrices, *B* and *C*. However, the last equation above is an expression for the numerator polynomial of the *z*-transfer function, and it is clear that there is dependence on the poles of the transfer function, i.e. the eigenvalues of the *A* matrix, both discrete and continuous. Hence, the zeros do not have the same simple mapping that the poles had.

2.5.3 Locations of Zeros and Poles of $G(z)$

Reference [7] developed the following understanding of the locations of the zeros and poles of the discrete time transfer function $G(z)$ for a transfer function with more poles than zeros (Reference [8] gives an alternative proof of the asymptotic zero locations):

(1) The poles of $G(s)$ transform to the poles of $G(z)$ according to $z_j = e^{s_j T}$.

(2) Any finite zeros of $G(s)$ transform to zeros of $G(z)$ approximately according to $z_j = e^{s_j T}$. The actual location of each zero matches the Taylor series expansion of this exponential up through powers of *T* equal to the number of poles in the system.

(3) The pole excess of $G(s)$ is the number of poles minus the number of zeros. Additional zeros are introduced in $G(z)$ to create a pole excess of one, i.e. the number of zeros is one less than the number of poles. These new zeros are on the negative real axis. The asymptotic locations as *T* tends to zero for the zeros outside and on the unit circle are given in Table 1. There are additional zeros, for each zero outside the unit circle there is a corresponding zero inside the unit circle at the reciprocal position.

Pole Excess	Zero Locations
2	-1.0000
3	-3.7321
4	-9.8990, -1.0000
5	-23.2039, -2.3225
6	-51.2184, -4.5419, -1.0000
7	-109.3052, -8.1596, -1.8682
8	-228.5110, -13.9566, -3.1377, -1.0000
9	-471.4075, -23.1360, -4.9566, -1.6447
10	-963.8545, -37.5415, -7.5306, -2.5155, -1.0000
11	-1958.6431, -59.9893, -11.1409, -3.6740, -1.5123

Table 2-1. Asymptotic zero locations for zeros outside and on the unit circle as a function of the pole excess of $G(s)$

2.6 Continuous Time and Discrete Time Frequency Response

To find the frequency response of a Laplace transfer function, one substitutes $s = i\omega$ into the transfer function and computes the magnitude which represents the amplitude response. The phase change through the system is given by angle this complex number makes with the positive real axis. To find the frequency response of a z-transfer function, one substitutes $z = e^{i\omega T}$, and again finds the magnitude and phase in the same way. The contribution to the overall phase from a zero inside the unit circle and a zero outside the unit circle, can be visualized from Figure 2-3. At Nyquist frequency, i.e. when $z = -1$, a zero inside contributes +180 deg, e.g. θ_1, and a zero outside contributes 0 deg, e.g. θ_2. Similarly, a pole inside contributes -180 deg, and we consider only stable feedback control systems $G(s)$ so there cannot be any poles outside the unit circle.

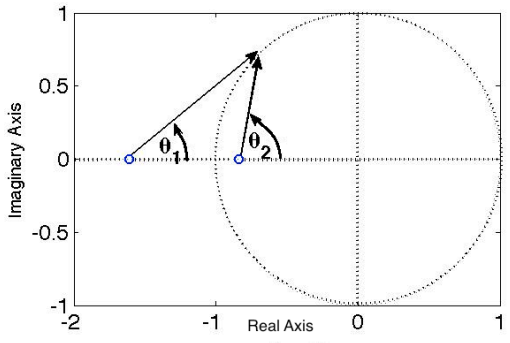

Figure 2-3. The phase angle contribution of a zero inside and a zero outside the unit circle.

2.7 A New Simple Pole Zero Mapping Design Method

In this section we modify emulation method (iv), in particular we modify step (4) making use of the understanding above of the zero locations introduced by the discretization process. We divide the mapping into 4 different cases, depending on whether there are zeros in $G(s)$, and whether the pole excess is odd or even. The subsections below treat these different cases.

2.7.1 $G(s)$ Has Odd Pole Excess, More Poles than Zeros, All Zeros in Left Half Plane

(1) For every pole s_j of stable transfer function $G(s)$ introduce a zero in $F(z)$ at the location $z_j = e^{s_j T}$. Introduce a pole in $F(z)$ for each zero inside the unit circle using the same formula.

(2) Introduce a number of poles at the origin equal to $(p_e - 1)/2$ where p_e is the pole excess.

(3) Determine the reciprocal of the DC gain of $G(s)$ by setting $s = 0$. Then adjust a gain in $F(z)$ so that $F(1)$ matches this reciprocal.

The main objective is to create an $F(z)$ that cancels the phase change through $G(z)$ so that the product $G(z)F(z)$ is real and positive. Then the stability condition is satisfied (unless one decides to use a gain greater than 2). The approach cancels the poles of $G(z)$. But the zeros

introduced in the discretization remain. Asymptotically there are an equal number of zeros outside the unit circle as inside, and they are reciprocals of each other.

Consider a pair of zeros, one at $-a$, and the other at the reciprocal location $-1/a$, for a positive, i.e. a zero on the negative real axis. For each such pair, Step (2) puts a zero at the origin. Then the transfer function and its zero phase frequency response obtained by setting $z = e^{i\omega T}$ are given by

$$\frac{(z+a)(z+1/a)}{z} = (a + 1/a) + 2cos\omega T \tag{2-21}$$

This magnitude response is always positive and bounded, so that one can keep it less than 2 for stability. For future reference, note that if the zero is on the positive real axis, one must introduce a minus sign in the equation for stability. Since this is a positive quantity up to Nyquist, the placement of a pole at the origin has accomplished the desired phase cancellation. Reference [9] makes deliberate use of this concept to cancel the phase influence of zeros outside the unit circle. Here we rely on nature to supply zeros at approximately reciprocal locations, so that all we have to do is supply to pole at the origin. This pole at the origin can be thought of in a different way, it simply represents adjusting the number of time steps delay through the system. Hence, the adjustment of this delay is all that one needs to do to stabilize the influence of zeros outside the unit circle, provided the zeros are sufficiently close to being reciprocals. We will see that they usually are. The gain adjustment is not critical.

Asymptotically as same time *T* goes to zero, the zeros are reciprocals of each other cancelling phase perfectly, but as the sample time interval increases they need not stay perfect reciprocals. Figure 2-4 presents this for the third order model, giving the two zero locations as a function of the sample rate, and compares the location of the zero inside the unit circle to the

reciprocal of the zero located outside the unit circle. We can watch these two curves converge to the same curve. They are reasonably close for all but very low sample rates.

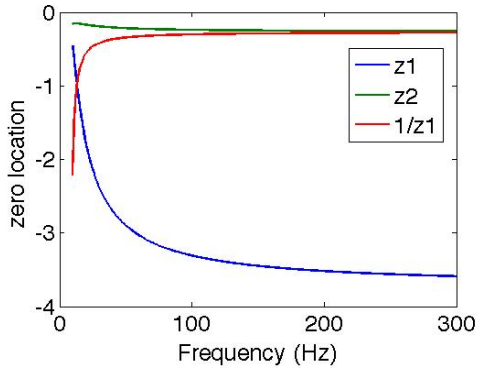

Figure 2-4. Locations of zeros introduced by discretization for third order model as a function of sample rate.

Note that the phase of any factors in the numerator of a transfer function are additive, and the phase of any factors in the denominator subtract to form the overall phase of a transfer function. In the case of a third order system considered here, we have the zero inside, the zero outside, and the pole at the origin. The other poles and zeros are essentially cancelled. Figure 2-5 gives the corresponding phase for sample rates from 25 Hz to 300 Hz. The zeros are not quite reciprocals so that there are phase contribution from these terms, but the maximum contribution observed for the slow sample rate is about 18 degrees, and this sample rate is about as slow as one would be willing to use for this system. The stability condition, when the repetitive control gain gets small, is able to tolerate a maximum of plus or minus 90 degrees phase in the product $G(z)F(z)$. We see that the phase from this term in the 3rd order case is well within such a tolerance. The 5th order system has two pairs of zeros, and for particularly slow sample rates can produce more that 40 degree error. This is still within the convergence limit. The magnitude plots are an indication of the learning rate, i.e. the decay per period for each frequency, as suggested by the form of the homogeneous difference equation give previously, and made rigorous in References [2] and [8].

The learning rate is similar to that produced by the compensator design method of Reference [9], the learning becomes slower at high frequencies. The design method of Reference [10] nominally creates a learning rate that is more uniformly at all frequencies. This is a price that we pay for using the simple design technique.

2.7.2 $G(s)$ Modification of 2.7.1 for Even Pole Excess

The extra consideration associated with even pole excess is that there is an odd number of zeros introduced, all but one is paired at reciprocal location, and the remaining zero asymptotically approaches $z = -1$. This approach is normally from inside the unit circle. From Figure 2-3 we know that a zero near -1 will produce a fast phase change near Nyquist frequency. When the frequency puts z straight above the zero, the phase contribution is 90 degrees, and it becomes 180 degrees when reaching Nyquist. One can take several approaches to deal in a simple way with the phase contribution of this zero.

It is possible that the phase error from this zero remains small enough for convergence up to a relatively high frequency. And then one may be quite willing to use the zero-phase low-pass FIR filter to cut off the learning when the phase becomes too large. This cutoff can be tuned in hardware. One may want the cutoff for other reasons related to energy consumption, actuator limits, etc. Figure 2-6 studies the phase contribution for different sample rates for a second order system obtained by deleting the first order factor in the third order model. It also gives results for a 4^{th} order system that is just the square of the quadratic factor. We observe again that the phase error can be tolerable, but is perhaps larger than one would like.

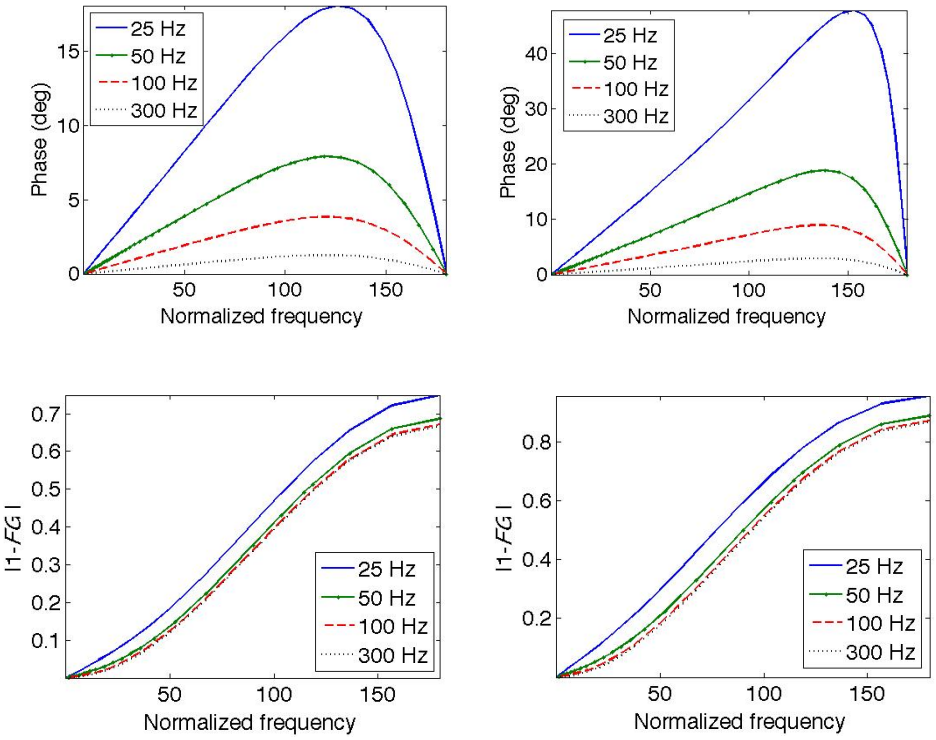

Figure 2-5. Phase deviation from zero and corresponding magnitude |1- *FG* |as the sample time is changed for the 3rd order system (left) and the 5th order system (right).

Another very simple thing that one can do is simply place a pole near -1 to nullify the effects of the zero without particular concern for the actual zero location. Figure 2-7 considers the 4th order system, and simply places a pole at -0.9 as a way to nullify much of the phase contribution. Of course, one could aim to put the pole underneath the zero, but the design process we are developing designs the digital RC directly from the continuous time transfer function without finding the actual zero location. This location is -0.541, -0.741, -0.862, and -0.952 for 25, 50, 100, and 300 Hz, respectively. Remember that 25 Hz and even 50 Hz are rather slow sample rates to use for this system. The numbers are very similar for the second order system. The design produced by this process is stable and converges to zero error.

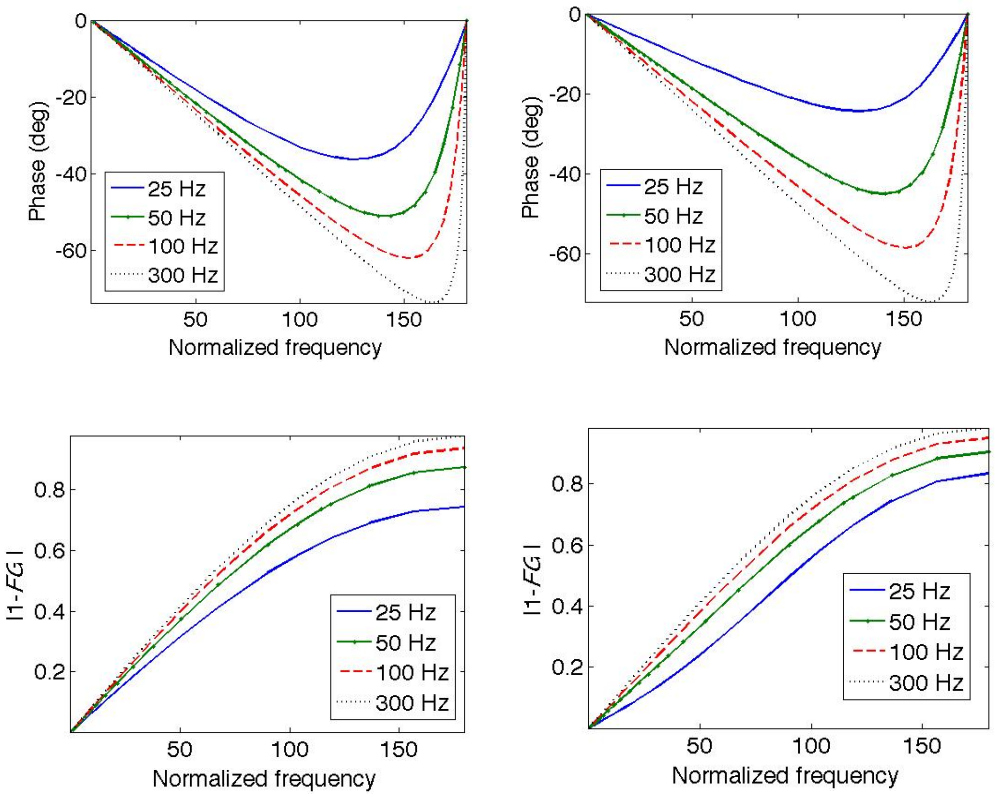

Figure 2-6. Phase contribution and the corresponding magnitude of |1- *FG* | from the zero that tends to $z = -1$ for the second order (left) and the 4th order (right) systems.

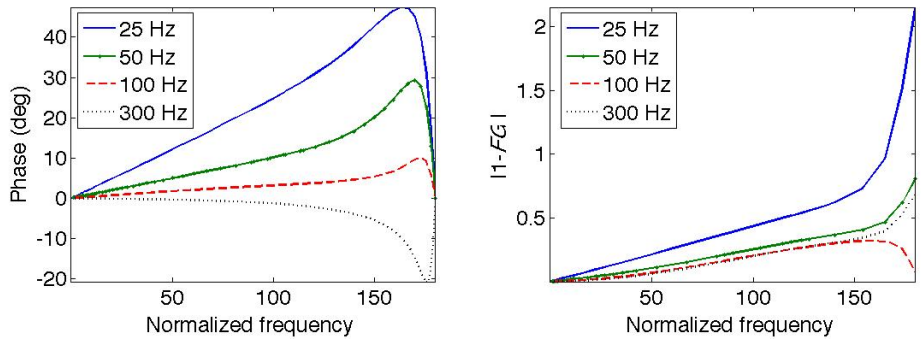

Figure 2-7. Phase contribution and the corresponding magnitude of |1- *FG* | from the zero that tends to $z = -1$ for the 4th order system when a pole is placed at -0.9.

2.7.3 $G(s)$ Has a Zero or Zeros in the Left Half Plane

If there is a zero s_j of $G(s)$, then one can place a pole at $z_j = e^{s_j T}$. Technically, this is an approximate cancellation because the actual zero location is also a function of the pole locations

as described above. However, this approximate mapping is good through powers of *T* in the Taylor series expansion equal to the order of the system. A third order system sampled at 100Hz will start with in the T^4 term in the Taylor series where $T^4 = 10^{-8}$. Thus, this approximation will be quite accurate in most applications.

2.7.4 $G(s)$ Has a Zero or Zeros in the Right Half Plane

When we have a non minimum phase system, we should know that it has this property. Knowing the zero location in the right half of the *s* plane, we can pick from three approaches. Figure 2-8 considers a system $G(z)$ consisting of a single zero outside the unit circle on the positive real axis at 1.1. It is interesting to note that if we pick $F(z)$ as -1 times a normalizing gain that makes the maximum value of the positive product $F(z)G(z)$ equal to unity, then we get curves *a* in Figure 2-8. The left plot shows that the phase of $F(z)G(z)$ stays less than 30 degrees, and the right plot shows that the resulting RC system is stable. The second method makes use of the technique discussed above that eliminates phase for zeros introduced by discretization that are outside the unit circle on the negative real axis. Again, take the reciprocal of the zero and supply a pole at the origin. Because the zero is on the positive real axis we need to multiply by -1. Then we find the maximum value, this time at Nyquist instead of at DC, and introduce a normalization to unity of the product $F(z)G(z)$ at this frequency. This is plot *b* in Figure 2-8. it does a perfect job of cancelling the phase, but does not produce a reasonably uniform learning rate at all frequencies. In fact, the right hand side of the plot indicates that the learning rate goes to zero at DC. The third approach following Reference [11] imitates the zero in multiple locations evenly spaced around a circle of the radius of the zero image in the *z* plane. Again one needs to multiply by -1, and then normalize. One finds the maximum magnitude over all frequencies for the product with $G(z)$, and introduces a normalization so that the maximum value of $F(z)G(z)$ becomes unity.

Figure 2-8 considers three cases of repeating the zero so that there is a total number of zeros equal to 4, 8, and 16 (plots *c*, *d*, *e* in Figure 2-8 respectively). The more times the zero is repeated, the closer the phase stays to zero phase. The right plot shows the magnitude plots. Each plot goes to zero at a frequency that could be used for the normalization. The advantage of this approach is that the magnitude plot gives a more uniform learning at all frequencies, and the more zeros used the closer the plot stays to the ideal zero value of learning as fast as possible. The first approach is the simples, but the last approach can give the best result over all frequencies. To use this thinking on more general systems, note that the phase of the zero term considered here is additive in $F(z)G(z)$ with any other factors, and the gain of $F(z)G(z)$ is multiplicative. Chapter 4 and 5 further study non-minimum phase systems. In Chapter 4, we investigate how methods proposed here and other existing methods perform on several non-minimum phase systems. The solutions in plots *a*, *c*, *d,* and *e*, are equivalent to Approach 4, and the method in plot *b* is the same as Approach 6 in Chapter 4. Chapters 4 and 5 propose methods to deal with slow learning at low frequencies which perform well even when the image of the zero location gets closer to the unit circle than the example studied in this section.

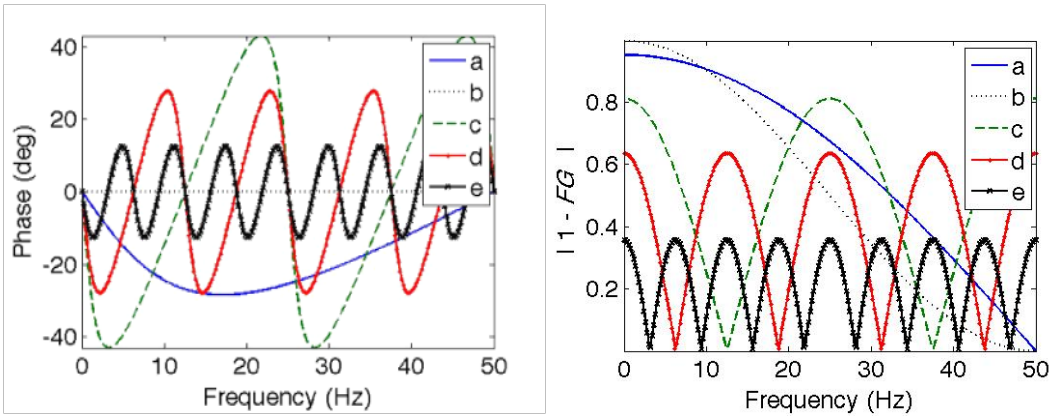

Figure 2-8. Canceling the phase contribution of a non minimum phase zero at z = 1.1.

2.8 Comments on Other Emulation Methods

2.8.1 Relationship to $z = e^{sT}$

Recall that the mapping of poles and zeros from continuous time to discrete time follow the exponential formula

$$z = e^{sT} \quad s = \frac{1}{T}\ln z \tag{2-22}$$

The second expression is not one that could be substituted into a Laplace transfer function to produce a *z*-transfer function of a difference equation. The emulation methods need this property, and note that the first 3 methods are all approximations of this exponential.

The Taylor series expansion of this exponential is

$$z = 1 + sT + \frac{1}{2!}(sT)^2 + \frac{1}{3!}(sT)^3 + \ldots \tag{2-23}$$

Emulation method (i) is just the first two terms in this expansion. Using the Taylor series expansion result $(1-x)^{-1} = 1 + x + x^2 + x^3 + \ldots$ on emulation method (ii) produces

$$z = 1 + sT + (sT)^2 + (sT)^3 + \ldots \tag{2-24}$$

For method (ii), again the first two terms match, but later terms do not. Then using the same expansion for the denominator of emulation method (iii) and multiplying produces the following

$$z = 1 + sT + \frac{1}{2}(sT)^2 + \frac{1}{6}(sT)^3 + \frac{1}{24}(sT)^4 + \ldots \tag{2-25}$$

For emulation method (iii), the first 3 terms match. However, we comment that these relationships go from *s* to *z* for all *s*, which is different than mapping just poles and zeros using the exponential relationship, and the Taylor series expansion properties only establish a close approximation for

sufficiently small sample time T. We need an approximation that is good to Nyquist frequency in terms of phase, or otherwise we cut off the learning.

2.8.2 Stability

It is very desirable that the emulation method produces a stable discrete time system if it is emulating a stable continuous time system. This means we would like the left half of the s-plane to map inside the unit circle in the z-plane. Figure 2-9 shows the image of the left half s-plane in the z-plane using the emulations methods (i), (ii), and (iii). The forward emulation can easily produce instability in discrete time. The backward guarantees stability but seriously constrains the possible dynamics. And the trapezoidal rule which coincides with the w-transformation does what this transformation is designed to do. It maps the closed left half plane onto the closed unit circle. In this respect the trapezoidal rule appears far superior.

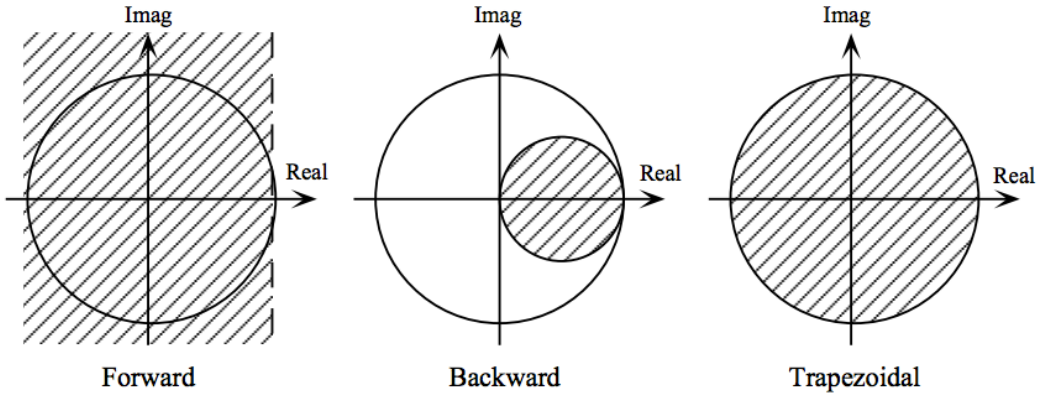

Figure 2-9. The image of the left half s-plane in the z-plane for Forward, Backward, and Trapezoidal emulation methods.

2.8.3 Time Delay Issues

When a continuous time transfer function $G(s)$ is fed by a zero order hold and the output sampled, generically there will be a one time step delay from the time step at which the input is changed to the time step at which the sampled output first changes. This means that the correct

$G(z)$ must have one more pole than zero. Our RC applications need a zero order hold input to the physical world, so our compensator which tries to be the inverse of this system, will need to have a one-time step phase lead.

Examining emulation method (i) we observe that the number of time steps delay in the resulting digital system is equal to the pole excess in $G(s)$, i.e. it is equal to the number of poles minus the number of zeros. Unless this pole excess is unity, there will be fatal phase errors approaching Nyquist. At Nyquist frequency there are two samples per period of oscillation. An extra time step delay at this frequency corresponds to an extra phase lag of 180 degrees. According to the stability condition, a 180 degree phase difference between $G(z)$ and $F^{-1}(z)$ will definitely violate the stability condition, unless $H(z)$ is able to cut off the learning process at higher frequencies to stabilize the RC system at the expense of not addressing the error components above the cutoff.

Examining emulations methods (ii) and (iii) we find that the number of zeros in the resulting $G(z)$ is equal to the number of poles. Thus, the resulting discrete time transfer function has no time delay in it. The same comments apply again here, that we know that we will not be able to have a stable design all the way to Nyquist frequency without the use of a cutoff.

2.8.4 Performance of Emulation Methods (i) and (ii)

Consider the third order system and apply emulation method (i), the forward rectangular rule. The plot of $|1 - F(z)G(z)|$ for z on the unit circle from zero to Nyquist is shown on the left in Figure 2-10 and we see that it goes above unity somewhere between 15 and 20 Hz. We can design a cutoff filter $H(z)$ to stabilize this design, resulting in the plot of $|H(z)[(1 - F(z)G(z)]|$ on the right of the figure, which will result in zero error for error frequency components roughly up to this cutoff (there are subtleties related to the effective cutoff [5]. It is reasonably likely that

eliminating errors up to this cutoff would eliminate the majority of the error, making this design a simple and effective design.

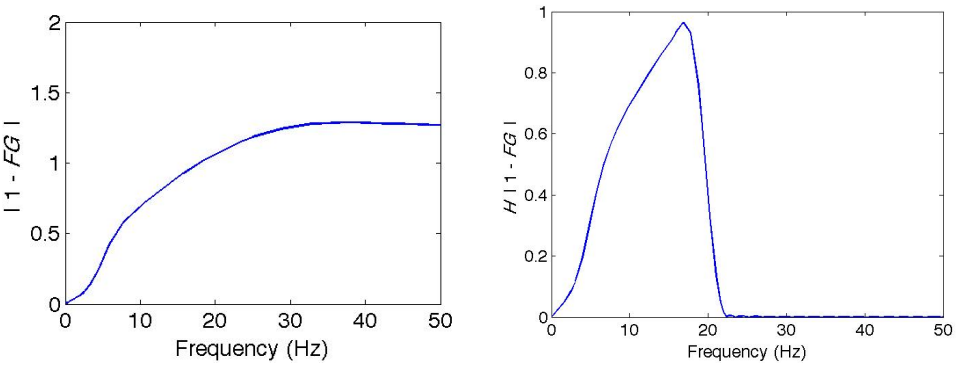

Figure 2-10. The stability condition for emulation method (i) applied to the 3rd order system. Without cutoff (left), and with stabilizing cutoff (right).

The corresponding figures for emulations method (ii), the backward rectangular rule, are given in Figure 2-11. Similar comments apply, but note that this time the cutoff frequency must be considerably lower, making the design less effective.

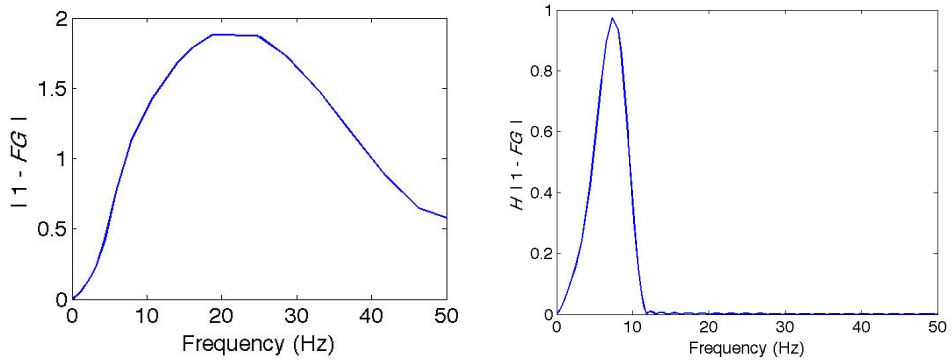

Figure 2-11. Plots corresponding to Figure 2-10 for emulation method (ii).

2.8.5 Effectiveness of a Frequency Cutoff Filter

We expect to need to use a frequency cutoff in real world applications because it is hard to have a good system model at high frequencies, yet RC needs to know the model phase to within plus or minus 90 degrees to be able to guarantee stability. We also may want a cutoff to respect

the limitations of the hardware, correcting periodic errors far above the bandwidth of the system can ask for unreasonable actuator output. In cases when we want to obtain zero error up to the maximum possible frequency, we should use a sophisticated RC design as in References [2] and [12]. In this case we are forced to tune the cutoff filter $H(z)$ in hardware since we do not know what is wrong with our model. We let the hardware tell us at what frequency the learning process no longer works.

The above emulation methods could be convenient to use when we are not concerned to learn to the maximum possible frequency, and apply the emulation to design the compensator and then cut off the learning in hardware. This thinking applies to emulation methods (i) and (ii), but method (iii) has additional difficulties. The denominator of emulation method (iii) has the factor $(z+1)$ in the denominator of the expression for *s*. One sets $z = \exp(i\omega T)$ in the resulting $G(z)$ to find frequency response, and this term becomes zero when reaching Nyquist frequency where $z = -1$. Assuming there are more poles than zeros in $G(s)$, when one clears the fractions in $G(z)$, there will be a factor of $(z+1)$ to the power of the pole excess that multiplies the numerator. The compensator $F(z)$ uses the reciprocal of this transfer function, and therefore has a singularity at Nyquist frequency. Our FIR design for $H(z)$ will not produce a zero at Nyquist. We could insist that this filter have the requisite number of zeros at Nyquist frequency to kill the singularity. To do so, we modify the design of $H(z)$. The design currently is a simple least squares problem and requires only the solution of a linear set of equations to determine the gains. To impose the requirement that the FIR filter produce zero at Nyquist, i.e. when $z = -1$, we need to require that $H(-1) = 0$. This means we require the following equality constraint be satisfied

$$a_0 - 2a_1 + 2a_2 - \ldots + 2(-1)^n a_n = 0 \tag{2-26}$$

This eliminates one unknown coefficient in the design equations, again leaving a linear set of equations to solve. If one needs two zeros at Nyquist one can produce the corresponding equation $dH(z)/dz|_{z=-1} = 0$. But this is perhaps not a robust process in real world applications, trying to kill something going to infinity with something that is going to zero.

2.9 Conclusions

When considering the use of emulations methods (i), (ii), and (iii) in the design of repetitive control compensators, a number of issues appear. None of the methods produces a model that is good all the way to Nyquist frequency, and hence to be used one must employ a frequency cutoff filter. This can be practical, but one might conclude that there is not enough benefit of simplicity of design to motivate one to use this approach.

The original pole-zero emulation method also had related difficulties at Nyquist frequency. However, we are able to create a modified version that has considerable appeal. One simply cancels the continuous time images of poles inside the unit circle, and if present we do the same for any zeros inside in continuous time. Then we simply adjust the time delay through the system, and in many cases this will create a workable RC law. It is evaluated how effective this very simple approach is, and one sees that it can be very effective. Some special considerations are used for even pole excess. The evaluation of how well this design approach works, sheds light on the robustness properties of more general RC design methods.

2.10 References

[1] G. F. Franklin, J. D. Powell, A. Emami-Naeini, *Feedback Control of Dynamic Systems*, Prentice Hall, 2009.

[2] R. W. Longman, "On the Theory and Design of Linear Repetitive Control Systems," *European Journal of Control*, Special Section on Iterative Learning Control, Guest Editor Hyo-Sung Ahn, Vol. 16, No. 5, 2010, pp. 447-496.

[3] B. Panomruttanarug and R. W. Longman, "Frequency Based Optimal Design of FIR Zero-Phase Filters and Compensators for Robust Repetitive Control," *Advances in the Astronautical Sciences*, Vol. 123, 2006, pp. 219-238.

[4] J. Bao and R. W. Longman, "Enhancment of Repetitive Control using Specialized FIR Zero-Phase Filter Designs," *Advances in the Astronautical Sciences*, Vol. 129, 2008, pp. 1413-1432.

[5] R. W. Longman and W. Kang, "Issues in Robustification of Iterative Learning Control Using a Zero-Phase Filter Cutoff," *Advances in the Astronautical Sciences*, Vol. 127, 2007, pp. 1683-1702.

[6] M. C. Isik and R. W. Longman, "Explaining and Evaluating the Discrepancy Between the Intended and the Actual Cutoff Frequency in Repetitive Control," *Advances in the Astronautical Sciences*, Vol. 136, 2010, pp. 1581-1598

[7] Y. Li and R. W. Longman, "Better Holds Can Make Worse Intersample Error in Digital Learning Control," *Proceedings of the 2006 AIAA/AAS Astrodynamics Specialist Conference*, Keystone, CO, Aug. 2006.

[8] J. W. Yeol and R. W. Longman, "Time and Frequency Domain Evaluation of Settling Time in Repetitive Control," *Proceedings of the AIAA/AAS Astrodynamics Specialist Conference*, Hawaii, August 2008.

[9] K-K. Chew and M. Tomizuka, "Steady-State and Stochastic Performance of a Modified Discrete-Time Prototype Repetitive Controller," 1988 ASME Winter Annual Meeting, December 1988, also in ASME *Journal of Dynamic Systems, Measurement and Control*, March 1990, pp. 35-41.

[10] B. Panomruttanarug and R. W. Longman, "Repetitive Controller Design Using Optimization in the Frequency Domain," *Proceedings of the 2004 AIAA/AAS Astrodynamics Specialist Conference*, Providence, RI, August 2004.

[11] K. Xu and R. W. Longman, "Use of Taylor Expansions of the Inverse Model to Design FIR Repetitive Controllers," *Advances in the Astronautical Sciences*, Vol. 134, 2009, pp. 1073-1088.

[12] R. W. Longman and S.-L. Wirkander, "Automated Tuning Concepts for Iterative Learning and Repetitive Control Laws," *Proceedings of the 37th IEEE Conference on Decision and Control*, Tampa, Florida, Dec. 1998, pp. 192-198.

Chapter 3

Repetitive Control Using Real Time Frequency Response Updates for Robustness to Parasitic Poles

3.1 Introduction

Chapter 2 proposes a simple method to create repetitive controllers directly from the continuous time model. Reference [1] develops the mathematical theory saying that this approach will produce a repetitive control system that converges to zero error if the system model and the real world model do not differ too much. However, in the physical world, one always expects that there will be some unmodeled high frequency dynamics, sometimes described as parasitic poles, other times it represents residual modes. One missing pole at high frequency can potentially produce a model error approaching the 90 degree limit, and this makes RC subject to instabilities from missing high frequency dynamics. In practice, this is addressed by introducing a zero-phase low-pass filter to cut off the learning at frequencies where the model is not sufficiently accurate for convergence [2-3].

If one wants the minimum possible tracking error, one wants to push this cutoff to as high a frequency as possible. Then one can aim for zero error at all harmonics below the cutoff, and must tolerate the remaining error. Note that one cannot design this cutoff frequency before building the hardware. The needed cutoff frequency is dictated by what is wrong with one's model. If one knew what was wrong with one's model, one would fix it. Hence, one must tune this part of the

repetitive controller by observing the hardware behavior, and then one can observe how much error one is able to eliminate.

If a frequency below the cutoff used happens to have too much phase lag, and produces instability, it is interesting to note that the instability is usually a very slow instability. At high frequencies the magnitude frequency response can be very low, so the error grows very slowly. Instability in feedback control systems very often is a sudden catastrophic growth in error, but in typical RC the instability is slow. This allows us to let the instability grow slowly, bringing error out of the noise level, and producing data that is precisely the data needed in order to know what is wrong with the phase information at that frequency. Reference [4] exploits this aspect of repetitive or learning control. It suggests use of RC to create an alternative to algorithms for optimal experiment design for purposes of identification. One intentionally uses a model putting one near the stability boundary of the nominal model. Errors in the model easily produce growing error data that can be used to correct the model.

The purpose of this chapter is to develop several approaches to producing repetitive controllers that observe the input and the output histories while repetitive control is running, and develops the frequency response knowledge needed for convergence. If one applies the inverse of the system steady state frequency response as a compensator applied to the error history, one produces a convergent RC law. A series of different approaches are treated.

(1) The first approach uses a moving window of one period of input-output data. One can perform a real time discrete Fourier transform (DFT) of this window, for each frequency one can see in the data, DC, the fundamental, and the harmonics. Applied to the input and the output, this allows one to find the frequency response or the inverse of the frequency response, provided the data is not changing fast. Then the DFT of the error tells us what signal to apply to cancel error.

This is a linear constant coefficient version of the repetitive control law created in Reference [5] for RC applied to linear systems with periodic coefficients. Such a moving window has been used in a number of RC approaches, including Reference [6]. Note that the DFT of a moving window can be performed easily in real time. For each frequency, one needs to add the newest input or output signal, delete the oldest signal, and multiply the remainder by a constant. Note that one can easily have a cutoff filter when using this approach, simply do not perform the DFT for frequencies that one does not want to address. This does not however eliminate the need for a cutoff filter on the signal applied to the control system, because leakage can produce frequencies above the cutoff that need to be eliminated from the input before it is applied. This approach is a direct way to produce the desired compensator that is the inverse of the frequency response of the system, and for sufficiently slow learning rate it should converge. At faster learning rates, there will be leakage effects due to the nonstationary nature of the signal.

(2) An alternative method of finding the frequency response information makes use of the results obtained in Reference [7] for matched basis function repetitive control. This reference seeks to find input or output components on sine and cosine basis functions. Output basis functions are shifted relative to the input basis functions by the amplitude and phase change through the system. In place of the recursive DFT at each frequency, one finds the frequency components updated in real time from the projection algorithm of adaptive control, using one projection algorithm of each frequency of interest. This information defines the current magnitude and phase information for each frequency.

Reference [7] creates the RC law that projects the error onto output basis functions, then replaces the basis functions with the corresponding input basis functions that have the magnitude and phase inverted. This creates the repetitive control update of the control action for that

frequency. The equations involved are linear, but make projections onto sines and cosines, and this produces periodic coefficients. Reference [7] shows by analytical manipulation in Mathematica, that the resulting equations can be written as constant coefficient linear equations with specific locations for poles and zeros based on the system amplitude and phase response at the frequency considered. To make the RC law by this approach one computes the frequency response at the frequencies of interest using the projection algorithm. The resulting amplitude and phase is inserted in the equation for the RC pole-zero compensator to obtain the error cancelling control action based on the most recent model. This RC design approach is good because it allows one to address any isolated frequency or set of frequencies independently. Experience shows that when applied to too large a number of frequencies, one may have to keep the overall learning gain low producing a slow learning rate.

(3) The repetitive control law developed in Reference [8] develops an FIR filter to perform the compensator representing the inverse of the steady state frequency response. It is observed that only 12 gains in this compensator, corresponding to multiplying 12 time steps of error in the previous period, is sufficient to produce an inverse that is good to between 2 and 3 decimal digits on a third order example system. The 12 gains are obtained by minimizing the square of the difference between unity and the product of compensator times model, summed over the frequencies of interest. Hence, this RC law is very easy to apply. To move toward this type of RC law from that described in (1) above, once we have computed the current frequency response from the DFT results, one computes the FIR coefficients, as described in Reference [8] or [1].

(4) Finally we consider what may be the real practical problem of interest. Suppose that we have an initial model from which we perform the simple and effective FIR compensator as in (3) above. When initially implemented in hardware, one normally will observe the frequency content

of the error, looking for any frequencies for which the error is growing due to unmodeled high frequency dynamics, then picks a cutoff filter to cut off the learning before reaching these frequencies. Suppose one has done this, but is a bit unsatisfied with the final error level, and wants to raise the cutoff frequency. Instead of trying to learn everything that is wrong with the model immediately, one learns to raise the cutoff gradually, stopping when satisfied with the error level reached.

One raises the cutoff some chosen amount to include one or several extra harmonics. While using an FIR design that is aims to model the inverse of the frequency response, as in (2) above, one will observe growth of the error at the extra harmonics. By running a separate projection algorithm on the input and the output for each of these harmonics, one can determine the magnitude and phase relationships for these frequencies. And this information prescribes the corresponding pole-zero compensator to use on these frequencies. To do this the projection algorithm is finding the magnitude and phase difference from command input to the feedback control system, to the FIR compensated output, i.e. finding what is wrong with the compensated signal at the extra harmonics. And then one applies the pole-zero compensators for these harmonics to the already compensated signal. This approach allows one to raise the cutoff successively, quitting when one is satisfied with the error performance.

This is a particularly attractive new ability. Previously, the performance of a repetitive control system was limited by the accuracy of the model used in designing it, and how much it was limited is only known when one applies the RC law to the real world and must pick a cutoff to stabilize. The result may be disappointing. Normally, to get better results one must work to find a better model, redesign, and try again. Here, we have an adaptive process that allows one to raise the cutoff, and improve the performance, while the repetitive control system is running. In the

following sections, the equations needed for these various approaches are developed, and examples are produced to examine their behavior and effectiveness.

3.2 Approaches to Real Time Determination of Frequency Response from Repetitive Control Data

3.2.1 Moving Window

We wish to correct RC designs developed based on a feedback control system frequency response model, to account for high frequency model errors. One can estimate the steady state frequency response of the feedback control system using a moving window DFT containing p time steps, as utilized in various references, such as References [5-6]. One applies the DFT to both the input and the output, and from the ratio determines the frequency response. The way to propagate the frequency content to the next time step by using a shifting matrix is presented.

The Vandermonde matrix for a p time step window is

$$V = \begin{bmatrix} 1 & 1 & 1 & \cdots & 1 \\ 1 & z_0^{-1} & z_0^{-2} & \cdots & z_0^{-(p-1)} \\ 1 & z_0^{-2} & z_0^{-1} & \cdots & z_0^{-2(p-1)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & z_0^{-(p-1)} & z_0^{-2(p-1)} & \cdots & z_0^{-(p-1)(p-1)} \end{bmatrix} \tag{3-1}$$

and its complex conjugate V^* is obtained by replacing the minus signs by plus signs in the exponents. The frequency components from one period time history of input $u(kT)$ can be obtained as

$$\underline{U_k} = \begin{bmatrix} U_0(k) \\ U_1(k) \\ \vdots \\ U_{p-1}(k) \end{bmatrix} = \frac{1}{p} V \begin{bmatrix} u((k-p+1)T) \\ u((k-p+2)T) \\ \vdots \\ u(kT) \end{bmatrix} \tag{3-2}$$

The recursive form for computation in real time updates of the input frequency component *n* from time step *k* to time step *k*+1 can be propagated as

$$U_n(k+1) = z_o^n \left\{ U_n(k) + \frac{1}{p} [u\big((k+1)T\big) - u\big((k-p+1)T\big) \right\} \tag{3-3}$$

where $z_o = e^{i\omega_o T}$, $\omega_o = \frac{2\pi}{pT}$, and $\omega_n = n\omega_0$. The recursive form of the frequency components of the output and the error can be computed in the same way.

The magnitude response $r_n(k)$ is then given by

$$\left|\hat{G}_{k+1}(e^{i\omega_n T})\right| = \frac{|Y_{k+1}(n)|}{|U_k(n)|} \tag{3-4}$$

and the phase response $\tau_n(k)$ of the identified system is

$$\measuredangle \hat{G}_{k+1}\big(e^{i\omega_n T}\big) = \measuredangle Y_{k+1}(n) - \measuredangle U_k(n) \tag{3-5}$$

If there is a periodic disturbance of period *p* time steps, then one needs to apply this approach to the change in input and change in output from one period back to the present period, in order to eliminates its influence.

3.2.2 Projection Algorithm

The moving window DFT can work well on a steady state signal to determine frequency content. When applied to a *p* time step window observing some decay of the error, the DFT will produce a set of sinusoids that repeat the signal each *p* time steps. If there is decay during the p steps, then a discontinuity is produced from one period to the next, resulting in some inaccuracy.

Reference [7] uses a projection algorithm for matched basis function repetitive control that projects the error onto sinusoids and applies the matched sinusoids to the system. Here we use the projection algorithm to determine the frequency components of the input and the output to compute the frequency response, i.e. to find the input basis functions that match the output basis functions. The projection algorithm does not use a window, but does offer a forgetting factor that one tunes. Hence, some of the difficulties of the moving window are avoided. We use one projection algorithm for each frequency of the given period, of for each harmonic frequency requiring adjustment in the model to obtain convergence.

To find the frequency response at frequency $\phi_n = \omega_n T$, find the components of input $u(k)$ and of output $y(k)$ on both sine and cosine of this frequency

$$H_n(k) = [\cos(\emptyset_n k) \quad \sin(\emptyset_n k)] \tag{3-6}$$

The $\phi_n = \omega_n T$ is the normalized addressed frequencies for n = 0,1 ,…, N and range from 0 to π, and T is the sample time interval of the digital control system. The coefficients of the output and input components on these sines and cosines at time step k and frequency n are $\beta_n(k) = \begin{bmatrix}\beta_{nc}(k)\\ \beta_{ns}(k)\end{bmatrix}$ and $\gamma_n(k) = \begin{bmatrix}\gamma_{nc}(k)\\ \gamma_{ns}(k)\end{bmatrix}$, respectively.

The frequency coefficients of the output are updated from time step k to step k+1 according to the projection algorithm

$$\begin{aligned}\beta_n(k+1) &= \beta_n(k) + \frac{aH_n^T(k+1)}{c + H_n(k+1)H^T(k+1)}[y(k+1) - H_n(k+1)\beta_n(k)] \\ &= [I - aH_n^T(k+1)aH_n(k+1)]\beta_n(k) + aH_n^T(k+1)y(k+1)\end{aligned} \tag{3-7}$$

The input components $\gamma_n(k)$ are updated similarly. The magnitude response of the system for frequency n at time step k+1 assuming a one time-step delay through the system is

$$\hat{r}_n(k+1) = \sqrt{\frac{\beta_{nc}^2(k+1) + \beta_{ns}^2(k+1)}{\gamma_{nc}^2(k) + \gamma_{ns}^2(k)}} \tag{3-8}$$

Also, we know that the phase of output and input are equal to $\text{atan2}(\beta_{ns}(k+1), \beta_{nc}(k+1))$ and $\text{atan2}(\gamma_{ns}(k), \gamma_{nc}(k))$. The estimated phase of the frequency response of the system for frequency *n* is equal to

$$\tau_n(k+1) = \text{atan2}(\beta_{ns}(k+1), \beta_{nc}(k+1)) - \text{atan2}(\gamma_{ns}(k), \gamma_{nc}(k)) \tag{3-9}$$

3.3 Repetitive Control Laws Based on System Frequency Response

Without a cutoff filter, Eq. (2-8) in Chapter 2 becomes

$$\left|1 - F\left(e^{i\omega T}\right)G\left(e^{i\omega T}\right)\right| < 1 \qquad \forall\omega \tag{3-10}$$

which suggests designing the compensator *F*(*z*) so that the left hand side of the equation is zero, i.e. the compensator should be the inverse of the steady state frequency response of the feedback control system.

3.3.1 FIR Compensator RC Design

Reference [8] creates a compensator in the form of an FIR filter that sums a number of errors in the previous period multiplied by gains

$$F(z) = a_1 z^{m-1} + a_2 z^{m-2} + \cdots + a_m z^0 + \cdots + a_{n-1} z^{-(n-m-1)} + a_n z^{-(n-m)} \tag{3-11}$$

where the gains are chosen to minimize the left hand side of Eq. (3-10) summed over a suitably chosen set of frequencies from zero to Nyquist frequency

$$J = \sum_{j=0}^{N} \left[1 - G(e^{i\omega_j T})F(e^{i\omega_j T})\right] W_j \left[1 - G(e^{i\omega_j T})F(e^{i\omega_j T})\right]^* \tag{3-12}$$

where the asterisk indicates complex conjugation. This FIR filter can be a surprisingly good approximation of the inverse frequency response with relatively few gains. Note that the cost function can choose to ignore high frequencies where the model is suspect. The zero-phase low-pass cutoff filter is designed similarly as in Eq. (2-4) and (2-5)

If one wants to use this approach, and incorporate updated frequency response information, one can be using the moving window DFT for fundamental and all harmonics and apply the optimization criterion summed over those frequencies. Alternatively, one can do the same when using projection algorithms for all frequencies. But both methods do not need to compute an FIR filter, and can go directly to computing the frequency updates needed in the control action.

On the other hand, one can combine the approaches. For example, after increasing the cutoff frequency such that there is now a harmonic below the cutoff that has growing amplitude with repetition, one can use the projection algorithm to compute the frequency response of the product for this frequency component. Then use this information to make a pole-zero compensator for $G(e^{i\omega_j T})F(e^{i\omega_j T})$ to apply to the error, that addresses this frequency, aiming to now converge to zero error for this previously unstable frequency.

3.3.2 RC Design Directly from Moving Window Data

Once one obtained the estimated model from methods presented in previous section, one can adjust input according to current magnitude and phase of the estimated frequency response of each frequency. In the model, we assume that we select $u_n(k)$ to produce the desired $y_n(k+1)$ for n = 0, 1, …, p-1. For some systems, with negligible computation time one can modify the equations to use the $e(k)$ and to produce $u(k)$.

Once one obtains $r_n(k)$ and $\tau_n(k)$, one can compute the needed input by updating the frequency content of the input in the previous time step by using the LAW1 in Reference [5]

$$\underline{U}(k) = S_I U(k-1) + \Lambda L \underline{E}(k)$$
$$u(kT) = (V^*)_N \underline{U}(k) \tag{3-13}$$

where gain matrix Λ is a diagonal matrix with diagonal elements as $\lambda_0, \lambda_1, \lambda_2, \dots, \lambda_{p-1}$, $(V^*)_N$ is the last row of the matrix V^* and S_I is a time shifting matrix

$$S_I = diag[z_o^0, z_o^1, \dots, z_o^{(p-1)}] \tag{3-14}$$

and *diag* indicate a diagonal matrix of the given elements. Given the frequency response of the system computed from the moving window for fundamental and each harmonic

$$M(k) = diag\left[r_0(k)e^0, r_1(k)e^{i\tau_1(k)}, r_2(k)e^{i\tau_2(k)}, \dots, r_{p-1}(k)e^{i\tau_{p-1}(k)}\right] \tag{3-15}$$

the repetitive control matrix representing the inverse of the frequency response is given as

$$L(k) = diag\left[1/r_0(k), \left(1/r_1(k)\right)e^{-i\tau_1(k)}, \left(1/r_2(k)\right)e^{-i\tau_2(k)}, \dots, \left(1/r_{p-1}(k)\right)e^{-i\tau_{p-1}(k)}\right] \tag{3-16}$$

From one time step to the next, after obtaining error $e\big((k+1)T\big)$ one updates the error frequency components from the previous time step window to the time step *k*+1 window according to

$$E_n(k+1) = z_o^n \left\{E_n(kT) - \frac{1}{N} e\big((k-p+1)T\big)\right\} + z_o^{-n(p-1)} \frac{1}{N} e\big((k+1)T\big) \tag{3-17}$$

Above it is assumed that once the measurement of $e(kT)$ is available, $u(kT)$ can be computed fast without losing a time step. If the system needs one step for computation, one would be required to compute $u(kT)$ based on $\underline{E}(k-1)$. However, one can still use S_I to propagate one step forward

$$\underline{U}(k) = S_I U(k-1) + \Lambda L S_I \underline{E}(k-1) \tag{3-18}$$

If one is using the moving window algorithm alone, without using the FIR compensator, as detailed here, it is obvious how to apply a cutoff of the update process. One simply does not compute the frequency components that are not being addressed in the RC law. Mathematically, one can write RC Law (3-18) with a diagonal matrix Φ that contains ones on the diagonal up to the cutoff frequency and zeros thereafter

$$\underline{U}(k) = S_I \Phi U(k-1) + \Lambda L \Phi \underline{E}(k) \tag{3-19}$$

3.3.3 RC Design Based on Matched Basis Functions and the Projection Algorithm

Instead of a moving window, one can use a projection algorithm to compute in real time an estimate of the error projected onto the frequencies of interest, DC, fundamental, and harmonics of interest. The projection algorithm data for input and output gives the frequency response information, which defines the matched basis functions, for each frequency. Then one applies the matched basis function repetitive control theory developed in Reference [7] as the repetitive control law using these matched basis function. The output basis functions applied to the error signal to be corrected are sines and cosines at each frequency addressed, and the matched input basis functions contain the reciprocal of the amplitude change when the input goes through the system, and the negative of the phase change. As developed in Reference [7], the mathematical operations just described are equivalent to a time invariant pole-zero compensator for each frequency addressed. The transfer function to compute $u(k)$ from $e(k)$ is

$$\sum_{n=0}^{N} \lambda_n T_n(z) \tag{3-20}$$

where

$$T_n(z) = \left(\frac{a_n}{r_n}\right)\frac{[cos(\phi_n - \tau_n)z^2 - 2\cos(\tau_n)\, z + \cos\,(\phi_n - \tau_n)]z}{[z^2 + (a_n - 2)\cos(\phi_n)\, z + (1 - a_n)][z^2 - 2\cos(\phi_n)\, z + 1]} \tag{3-21}$$

is a compensator for each frequency n=1, 2, 3, …, N. The compensator contains 2 zeros and 4 poles for every frequency addressed, and a_n is gain picked in the projection algorithm. There is a 4th order difference equation needed to be computed in real time. This method allows one to pick which frequencies one wants to correct based on the frequency content of the error. Only r_n and τ_n need to be updated in $T_n(z)$ for each time step.

3.4 Adjusting the Learning Gain vs. Frequency and Producing a Frequency Cutoff

The methods proposed above also provide an option to introduce different learning gain for each frequency. For the moving window method, we can adjust gain α_n for each frequency in a diagonal matrix L. The new form of matrix L becomes

$$L(k) = diag\left[\begin{array}{cccc} \dfrac{\alpha_0}{r_0(k)} & \dfrac{\alpha_1}{r_1(k)}e^{-i\tau_1(k)} & \dfrac{\alpha_2}{r_2(k)}e^{-i\tau_2(k)} & \cdots \quad \dfrac{\alpha_{p-1}}{r_{p-1}(k)}e^{-i\tau_{p-1}(k)} \end{array}\right] \tag{3-22}$$

It is necessary that each α_n gain be the same for both terms in the complex conjugate factors that relate to the same frequency. For the projection algorithm, we can adjust gain similarly.

Three are various choices for these gains. One could learn each frequency at the same rate, in which case one can make all gains equal. And picking them equal to unity means one is applying the inverse of the steady state frequency response or one can use a different constant. The unit circle condition discussed in Reference [1] indicates that decreasing the learning rate makes repetitive control more robust to model error, in particular to phase error. As the learning rate approaches zero, the stability tolerance approaches plus or minus 90 degrees. Since model uncertainty gets larger at high frequencies, one can choose to use a small gain as the frequencies

go up. One choice according to this objective, is to make a learning law that only cancels the phase of the system

$$L(k) = diag\begin{bmatrix} 1 & e^{-i\tau_1(k)} & e^{-i\tau_2(k)} & \dots & e^{-i\tau_{p-1}(k)} \end{bmatrix} \tag{3-23}$$

or a constant times this. Because the frequency response of physical systems attenuates as the frequency goes up, this choice is learning more slowly at high frequencies. This choice in ILC corresponds to the partial isometry law. When using the projection algorithm to generate the frequency components, and then applying the pole-zero transfer functions of Equations (3-20) and (3-21), this becomes

$$T_n(z) = a_n \frac{[cos(\phi_n - \tau_n)z^2 - 2\cos(\tau_n)\, z + \cos(\phi_n - \tau_n)]z}{[z^2 + (a_n - 2)\cos(\phi_n)\, z + (1 - a_n)][z^2 - 2\cos(\phi_n)\, z + 1]} \tag{3-24}$$

In normal applications of repetitive control, it is usually necessary to use a cutoff filter to prevent updating frequencies for which the model is too inaccurate. Even one unmodeled parasitic pole at high frequency can be enough to produce instability and error growth at such frequencies. Many approaches require the design and application of a zero-phase low-pass cutoff filter in the form of an FIR filter. Observe that here we can avoid this task. For the moving window algorithm, one simply sets the alpha gains to zero for all frequencies above the desired cutoff. For the projection algorithm, on simply sets the associated λ_n to zeros in Equation (3-20), i.e. one does not compute any pole-zero compensator in Equation (3-21) for such frequencies.

3.5 Raising the Cutoff Frequency in a Repetitive Control Systems Using an FIR Compensator Based on Nominal Model

The methods discussed above so far consider that one designs the RC law based on frequency response for all frequencies, or all frequencies up to a cutoff. One can create methods

to observe the frequency content of the input and the output in real time and use this to update the frequency response information for each frequency, or for frequencies that require correction to have error convergence. The expected situation is that the nominal model gives sufficiently good frequency response information for convergence for frequencies up to some value. Often at high frequencies there are parasitic poles or residual modes, i.e. dynamics not present in the model, dynamics that might be hidden in the noise when using data to identify the nominal model. The cutoff frequency required for stability in the real world is then determined by what is wrong with one's model, and hence must be tuned in hardware after applying the RC design to the hardware.

Using the methods discussed above, one can adaptively update the frequency response gain and phase in real time or periodically, watching all frequencies, using either the moving window or the projection algorithm data being collected in real time. One needs decision criteria to know when the data for a frequency is sufficient to warrant use in correcting the model.

Another likely scenario, is to have a repetitive controller running, designed from the nominal model, that uses an FIR filter to mimic its frequency response inverse [8]. One might use frequency response data in real time to redesign the FIR filter, but this perhaps a bit awkward in real time. Then it is natural to ask, can we have an incremental raise in the cutoff, observe some frequency above the original cutoff growing slowly, use the data to design an update by the projection method or moving window method, and simply apply a correction to the exiting repetitive control system? For example, using the moving window to determine frequency response of $F(z)G(z)$. On has data for the input to the real world $G(z)$, observes the output, then processes this output through the filter $F(z)$, and determines what frequency responses need to be corrected. The product $F(z)G(z)$ when using the nominal model, aimed to have zero phase and gain of unity. for frequency of interest. The value of coefficient l_n, the element in diagonal matrix L

corresponding to frequency n, estimates how much the compensator $F(z)$ should to be corrected. For the moving window method, the needed changes in the input, $\Delta u(kT)$, can be computed as

$$\Delta u(kT) = \frac{1}{p} V_{p,n}^* l_n V_n \underline{e}(k) + \frac{1}{p} V_{p,n^*}^* l_{n^*} V_{n^*} \underline{e}(k) \tag{3-25}$$

where $l_n = \frac{1}{|F_n G_n|} e^{-i(\measuredangle F_n - \measuredangle G_n)}$, and n^* is a frequency corresponding to the conjugate frequency of frequency n , and $V_{p,n}^*$ is the element at p^{th} row and n^{th} column of the matrix V^*. For the projection algorithm, one can compute the value of r_n and τ_n to be the magnitude and phase response of $F_n G_n$ at frequency n.

3.6 Some Numerical Experience

Some computations were performed for a system having a residual mode. The actual system is fifth order with a second order underdamped factor with undamped natural frequency of 30Hz and damping ration of 0.5 system that has three poles and a pair of parasitic pole at 30 Hz

$$G(s) = \frac{a}{(s+a)} \left[\frac{\omega_1^2}{s^2 + 2\zeta\omega_1 + \omega_1^2} \right] \left[\frac{\omega_2^2}{s^2 + 2\zeta\omega_2 + \omega_2^2} \right] \tag{3-26}$$

where a=20, ω_1=$2\pi 5.8$ and ω_2=$2\pi 30$. We consider that the model used to design a repetitive controller fails to include this 30Hz mode. The window size p is 100, the sample frequency is 100 Hz, and the desired trajectory is a 10Hz sine wave.

Figure 3-1 shows the discrete time frequency response plots for the system model and for the full system. If one incorporates an overall repetitive control gain times the inverse of the frequency response of the model as the RC law, then as this gain approaches zero, one can obtain zero tracking error in the limit for all frequencies where the difference between the 3rd order model and the 5th order system is less than 90 degrees. Figure 3 shows what happens when one applies the RC designed for the 3rd order system to the real world. The error initial decays substantially,

corresponding to eliminating low frequency error, but after some time, the slow growing and initially smaller high frequency error starts to take over.

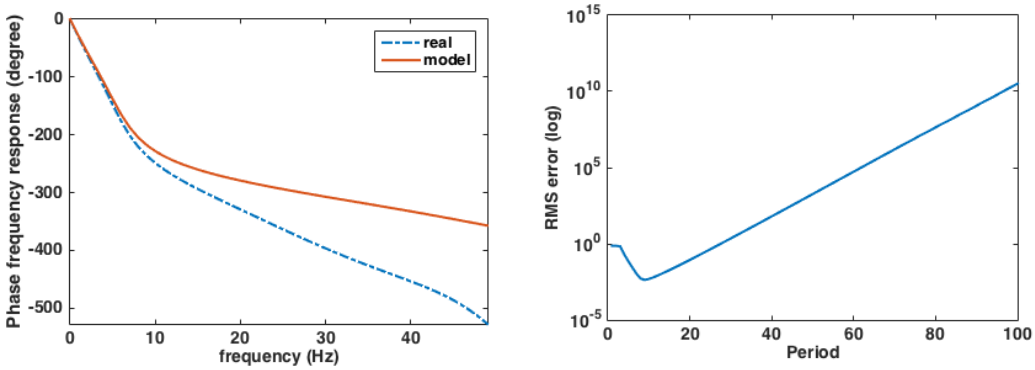

Figure 3-1. Left is the phase plot of the 5th order system, and of the 3rd order model. Right is the RMS error per period versus period when applying the 3rd order RC design to the 5th order system.

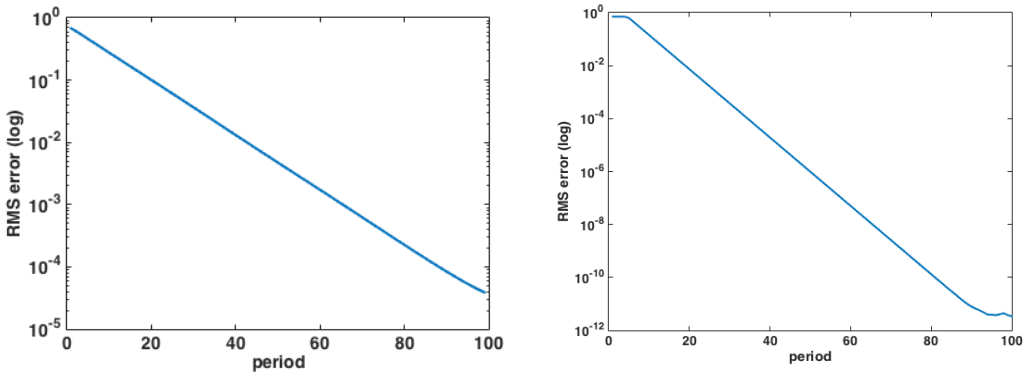

Figure 3-2. Left plot is RMS error per period vs. period when using the moving window method to update phase for all frequencies. Right plot uses the pole-zero compensator for 10Hz applied knowing the matched basis functions for this frequency.

There are various considerations to address when applying the moving window and the projection algorithm methods to update the frequency response model for the chosen frequencies. Concerning the moving window there are extra frequencies that appear that disturb the results. One issue is that the moving window is applied while the RC is running and therefore modifying the frequency content. The moving window with the right period will not have leakage effects on a steady signal, but the time variation introduced what appear to be errors at additional frequencies that are not relevant. Looked at another way, the DFT of a signal that is decaying due to learning,

determines a set of sinusoids to fit the signal in the window. If these sinusoids are extended beyond the present time, there is a step discontinuity (for the continuous time case) at the end of the current window to the start of the next period. Hence, the sinusoids are expecting the start of the next period to be the same as the earlier start, as if no decay of the error has appeared. For continuous time one also expect to see the Gibbs phenomenon. These issues go away as the learning proceeds, but they confuse the issue of whether the error at a given frequency might be growing due to a wrong model, or whether the error is just appeared to grow at the moment due to these effects. As a result, one needs to make a decision of how long to watch the growth of error at a given frequency, and how much growth is needed before one is justified in updating the frequency model.

Concerning the projection algorithm approach, there is no window involved, so the nonstationary nature is not as important, but the algorithm needs to be delivering reasonably current estimates of frequency content. One may need a forgetting factor so that the current estimate of a chosen frequency component of the error is reasonably current, but still allow enough time to be averaging through the noise. In addition, if one makes a periodic update of the repetitive control process, and is modeling the frequency response error in the product of the compensator times real world response, then one likely needs to restart the projection algorithm. Also, experience with the matched basis function RC method developed in Reference [7] indicates that when one is trying to use the approach on many frequencies, the gain may have to small producing slow convergence.

3.7 Conclusion

Repetitive control initially aims for zero error following a periodic command, or in the presence of a periodic disturbance. But this asks for zero error all the way to Nyquist frequency,

and one rarely has a good model at high frequencies. If the model phase error is below 90 degrees at all frequencies, the RC can converge to zero error for sufficiently small learning rates. But one unmodeled high frequency mode can produce exponential growth at the associated frequency. Behavior at this frequency may be buried in the noise level of the data when developing the model for RC design. When the model is wrong enough for error growth, then the RC action can start amplifying the error at these frequencies. Normally, RC must use a zero-phase low-pass filter to stop the learning process at such frequencies. The result is zero error only up to the cutoff needed for stabilization. The cutoff frequency needed is determined by what is wrong with one's model. One does not know in advance of trying to use the RC in hardware what the cutoff must be. The original aim was zero error, and one is disappointed in the final error, and unable to predict it during the design process.

The amplification is slow at high frequencies, because the frequency response of the system is small, but the RC grows the error and eventually it is clearly visible above the noise level, and has produced data ideal for finding out what is wrong with one's model. The objective of this chapter is to investigate methods of using the data developed by this process to modify the model in real time, reducing the model error successively, allowing one to raise the cutoff frequency. This makes what one might call adaptive repetitive control. But normally indirect adaptive control is trying to update a general input-output model. Here we only need to improve the model of the frequency response, and only at whatever frequency or frequencies are causing convergence difficulties. This is a very focused objective, and the slow instabilities at high frequencies in RC are ideally suited to this objective.

We consider trying to update the frequency response model at all frequencies, and present methods to do this. But perhaps the most practical version of the material presented here uses the

following scenario. One designs an RC law based on the reciprocal of the steady state frequency response of one's model. Then apply the law in hardware and analyze what frequencies are decaying and what frequencies are growing as time progresses. One then can make a cutoff filter that stabilizes the learning process, and results in essentially zero error up to the cutoff. Then one can move the cutoff up, going above some harmonic or harmonics that exhibit growth. Fix the phase and maybe amplitude information in the control action, and then repeat, moving the cutoff still higher. Learning all the way to Nyquist all at once is maybe not reasonable in many situations, so this progressively learning at higher and higher frequencies is a natural approach, which can be terminated whenever one wants. Experience with a robot control system illustrates this point. The bandwidth of the feedback controllers was 1.4Hz but Nyquist frequency was 200Hz. There was no way that one could obtain frequency response information anywhere near Nyquist frequency, even if RC were amplifying errors at these frequencies. So one still wants a cutoff to ensure that when RC is operated for a long period it will not be going unstable.

3.8 References

[1] R. W. Longman, "On the Theory and Design of Linear Repetitive Control Systems." *European Journal of Control*, Special Section on Iterative Learning Control, Guest Editor Hyo-Sung Ahn, Vol. 16, No. 5, 2010, pp. 447-496.

[2] B. Panomruttanarug and R. W. Longman, "Frequency Based Optimal Design of FIR Zero-Phase Filters and Compensators for Robust Repetitive." *Advances in the Astronautical Sciences*, Vol. 123, 2007, pp. 1723-1742.

[3] J. Bao and R. W. Longman, "Enhancements of Repetitive Control using Specialized FIR Zero-Phase Filter Designs." *Advances in the Astronautical Sciences*, Vol. 129, 2008, pp. 1413-1432.

[4] R. W. Longman, K. Xu, and B. Panomruttanarug, "Designing Learning Control that is Close to Instability for Improved Paramether Identification." *Modeling, Simulation and Optimization of Complex Processes*, Bock, Kostina, Phu, and Rannacher Editors, Springer-Verlag, Heidelberg, 2008, pp. 359-370.

[5] W. Messner, C. Kempf, M. Tomizuka, R. Horowitz, "A Comparison of Four Discrete Time Repetitive Control Algorithms." *IEEE Control Systems Magazine*, Vol. 13, No. 6, December 1993, pp. 48-54.

[6] H. Yau and R. W. Longman, "Frequeny Response Based Repetitive Control Design for Linear Systems with Periodic Coefficients." *Proceedings of the AIAA/AAS Astrodynamics Specialist Conference*, Vail, Co, August 2015.

[7] Y. Shi, R. W. Longman, and M. Nagashima, "Small Gain Stability Theory for Matched Basis Function Repetitive Control." *Acta Astronautica*, Vol. 95, 2014, pp. 260 -271.

[8] B. Panomruttanarug and R. W. Longman, "Repetitive Controller Design Using Optimization in the Frequency Domain," *Proceedings of the 2004 AIAA/AAS Astrodynamics Specialist Conference*, Providence, RI, August 2004.

Chapter 4

Using Quadratically Constrained Quadratic Programming to Design Repetitive Controllers: Application to Non-Minimum Phase Systems

4.1 Introduction

The previous two chapters develop design methods which are simple and robust to model error at high frequencies. The design methods typically perform well in general RC minimum phase systems, but have difficulty learning at low frequencies in non-minimum phase systems. Chapters 4 and 5 study RC designs for non-minimum phase systems and aim for methods that can address this problem.

Non-minimum phase systems in continuous time have Laplace transfer functions with one or more Laplace transfer function zeros in the right half plane. Such systems have the unusual property that they move the wrong direction first and then the right direction in response to a unit step input. In aerospace engineering, the dynamics of airplane control systems can easily exhibit this behavior. The dynamics of a rocket can do so as well. And in a spacecraft, if one wants to position the tip of an appendage by rotating a spacecraft hub, it can easily be non-minimum phase. Hence, a sensor at the end of a flexible element whose pointing is controlled by rotating the base of the element can be a non-minimum phase problem. The sensitivity transfer function from command to error, or from output disturbance to error is non-minimum phase when one is controlling an unstable system.

Iterative learning control (ILC) is a sister field within control, that aims for zero tracking error of a finite time command that is repeated, each time starting from the same initial condition. The most important difference is that ILC asks for zero tracking error during the transients of a finite time trajectory, while RC asks for zero tracking error asymptotically for a periodic trajectory or disturbance. Considerable research in ILC has studied the application to non-minimum phase systems, References [1-6]. These references characterize the difficulty of the problem, which results from the fact that the inverse of the continuous time transfer function is unstable. Very slow learning at DC (zero frequency) and low frequencies is observed. Generally, one makes a compromise with the requirement of converging to zero tracking error as the iterations progress.

The purpose of this chapter is to investigate the application of RC to non-minimum phase systems, determining to what extent there is a problem related to that encountered by ILC, studying how the various RC design approaches perform on such problems, and then determining how to address the difficulties that arise. Reference [7] also considers RC of non-minimum phase systems. In RC there can be two kinds of non-minimum phase systems: (1) Ones that have a zero outside the unit circle in the z-plane produced as the image of a zero in the right half of the Laplace transform plane, (2) And zeros on the negative real axis of the z-plane that are introduced by the process of converting from continuous time system fed by a zero order hold, to discrete time equivalent. Standard RC approaches handle the second type of zeros outside the unit circle. It is the purpose of chapters 4 and 5 to address problems of the first kind, i.e. ones which are non-minimum phase in continuous time before discretizing.

Six general RC design approaches are studied and evaluated for their usefulness in handling these non-minimum phase systems. In the process, an improved design method is created that generalizes the existing Taylor series approach. An important contribution is the development of

effective methods of using a Min-Max cost function for design optimization. The resulting design approach is evaluated, establishing that while the approach is not to be preferred for minimum phase systems, it is likely the preferred method to obtain effective RC performance in non-minimum phase systems.

4.2 Non-Minimum Phase Systems

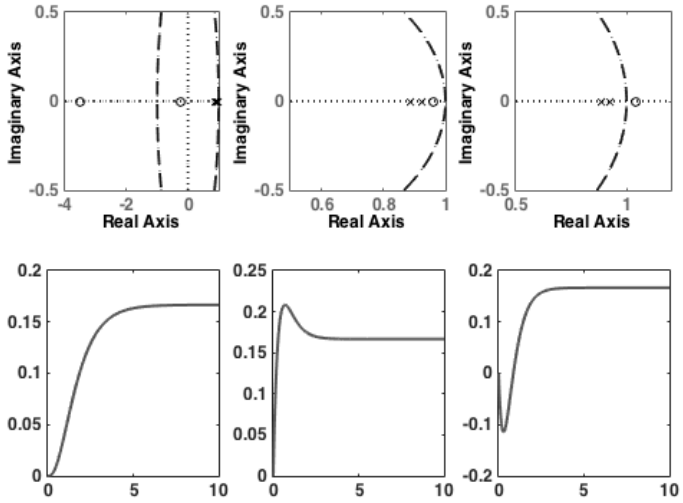

Figure 4-1. Pole-zero locations and resulting unit step responses of minimum and non-minimum phase systems.

Figure 4-1 presents some of the properties of non-minimum phase systems. Each system is given initially as a Laplace transfer function, and it is fed by a zero order hold running at 25Hz, which allows one to convert to a discrete time difference equation model represented as a z-transfer function. Then the unit step response of each system is shown.

The first system has 3 poles at $s = -1, -2, -3$. These map exactly to poles 0.96, 0.92, and 0.88 in the z-plane at $\exp(-T), \exp(-2T), \exp(-3T)$, and two zeros are introduced by the discretization, one inside the unit circle at -0.25, and one outside at -3.51 on the negative real axis.

Hence, this system is minimum phase in continuous time, but some might want to call it non-minimum phase in discrete time. In spite of a zero outside the unit circle in the *z*-plane, the unit step response does not first go negative and then decide to head towards the commanded unity value, i.e. the behavior is that of a minimum phase system. Reference [8] gives the asymptotic locations of the zeros outside the unit circle introduced by discretization, in the limit as the sample time interval T tends to zero. When the pole excess of the Laplace transfer function is 3 and 4 (number of poles minus the number of zeros), then one zero is introduced asymptotically located at -3.732 and at -9.899, respectively. For a pole excess of 5, two zeros are introduced outside at -23.322 and -2.322. For reasonable sample rates, these zeros have not reached these asymptotic values, but they are nevertheless rather far outside the unit circle.

The second system in Figure 4-1 removes the pole at $s = -1$ and instead places a zero there. Hence it is minimum phase both in continuous time and discrete time, and the unit step response is the expected response, and the zero near the origin has introduced some overshoot. The third system moves the zero from $s = -1$ to $s = +1$ making it non-minimum phase. The magnitude frequency response is unaltered, but the phase response is changed. And the unit step response shown, has the unsual property of going negative before going positive in response to a positive constant command. Note that for small T, the zero at $s = +1$ will map approximately to $z \approx \exp(+T)$.

We comment that the "true" non-minimum phase zeros in the *z*-plane, that are images of a non-minimum phase zero in the *s*-plane, will generally be very much closer to the unit circle than the zeros introduced by discretization described above. This property is one source of difficulty in the design of RC for non-minimum phase systems. It is the purpose of this chapter to study the

ability of RC design methods to handle non-minimum phase systems like this third system in the figure.

As an interesting phenomenon, one can analytically compute the zero location for a system such as the 3rd one in the figure, $G(s) = (s - a_1)/[(s - b_1)(s - b_2)]$, using the standard formula for zero order hold conversion $G(z) = (1 - z^{-1})Z\{G(s)/s\}$. The resulting image of the zero in the z-plane is given as

$$z_1 = \frac{Ae^{b_2T} + Be^{b_1T} + Ce^{(b_1+b_2)T}}{A(1 - e^{b_2T}) + B(1 - e^{b_1T}) + C(e^{b_1T} + e^{b_2T})} \tag{4-1}$$

$$A = \frac{(-b_1 + a_1)}{(-b_2 + b_1)(-b_1)} \quad B = \frac{(-b_2 + a_1)}{(-b_1 + b_2)(-b_2)} \quad C = \frac{-a_1}{b_1 b_2} \tag{4-2}$$

For fast sample rate the zero location is $z_1 \approx \exp(a_1 T)$, but for slow sample rates the location is a function of the pole locations. Figure 4-2 shows the zero location as the sample rate is varied, and we observe that as the sample rate is reduced, the zero location reaches positive infinity for a finite value, between 1.157 and 1.158Hz, and then the image of the zero starts coming in toward the origin from negative infinity on the negative real axis.

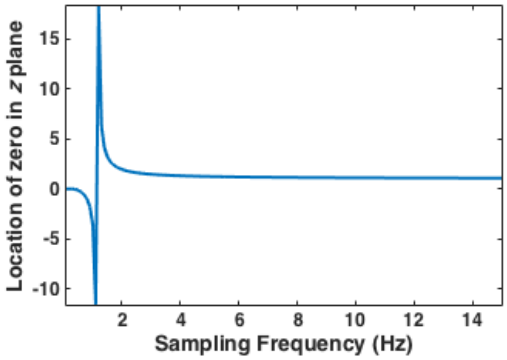

Figure 4-2. Discrete time zero location vs. sample frequency.

4.3 Approaches to Creating RC Compensators

Initially it appears as if picking $F(z) = G^{-1}(z)$ would be an ideal choice. It makes the left hand side of the stability condition zero. Also, the closed loop transfer function from the block diagram in Figure 2-1 is $1/z^p$ making the error difference equation deadbeat, converging to zero tracking error in one period. However, as noted previously, the process of converting a continuous time system fed by a zero order hold into an equivalent discrete time system introduces zeros outside the unit circle for systems with an original pole excess of 3 or more and a reasonable sample rate. In such cases, this choice of $F(z)$ produces an internal instability and cannot be used.

A set of different approaches to develop compensators $F(z)$ that satisfy the stability condition are treated here, producing a compensator in the form of a finite impulse response filter (FIR) that does not introduce internal stability problems. (1) One approach continues to aim for the inverse by using Taylor series expansion of the inverse transfer function, $G^{-1}(z)$. (2) A second approach makes the FIR filter mimic the inverse of the steady state frequency response of the system $F(\exp(i\omega T)) \approx G^{-1}(\exp(i\omega T))$. T is the sample time interval. If this is achieved the left hand side of the stability condition (8) is zero, and instead of achieving zero error in p time steps, zero error is achieved asymptotically as the response approaches steady state. (3) A third approach tries to make $F(\exp(i\omega T))$ minimize the left hand side of the stability condition Eq. (3-10) over frequencies from zero to Nyquist, previously presented in section 3.3.1. Based on References [9, 10], this objective is trying to maximize the decrease in error from one period to the next, i.e. the optimization criterion is the learning speed. (4) A fourth approach is less ambitious, and simply asks that the $F(z)$ cancel the phase of $G(z)$ at all frequencies.

This chapter investigates the performance of 6 different design approaches, with some of the above having two different mathematical formulations. These approaches will be labeled

Approach 1 through Approach 6, and the mathematics for each is discussed in a separate section below. Note that the numbering for the Approaches is in a different order than the order in the listing above.

Generally, the FIR filter has the form

$$F(z) = a_1 z^{m-1} + a_2 z^{m-2} + \cdots + a_m z^0 + \cdots + a_{n-1} z^{-(n-m-1)} + a_n z^{-(n-m)}$$
$$= (a_1 z^{n-1} + a_2 z^{n-2} + \cdots + a_m z^{n-m} + \cdots + a_{n-1} z^1 + a_n z^0)/z^{(n-m)} \quad (4\text{-}3)$$

which in practice simply asks to compute a linear combination of the errors possibly both forward and backward from the time step that is *p* steps in the past. The coefficients are computed based on one of the above objectives. The choice of *n* decides that there will be $n-1$ zeros in the FIR filter, and the choice of m dictates that there will be $n-m$ poles at the origin. When $n-m=0$ there are no poles. We comment that if the design is dealing with poles and zeros inside the unit circle, designs will likely want to put zeros more or less on top of the poles, and one should allow enough zeros for this. And then the total number of zeros inside the unit circle must equal the total number of poles in order to satisfy Eq. (3-10), and this dictates the number of poles to request at the origin by choice of m.

Often it is helpful to make an infinite impulse response filter (IIR), denoted $F_2(z)$, that cancels all of the poles and zeros inside the unit circle, and then design an FIR compensator $F_1(z)$ of the above form using one of the above objectives, addressing only the zeros that remain after applying the IIR filter, the zeros outside the unit circle that could not be cancelled. To illustrate, if

$$G(z) = \frac{K_G(z-z_1)(z-z_o)}{(z-p_1)(z-p_2)(z-p_3)} \quad (4\text{-}4)$$

and z_1 is a zero inside the unit circle, and z_o is a zero that cannot be cancelled outside the unit circle, then the IIR applied to data from the previous period has the form

$$F_2(z) = \frac{(z-p_1)(z-p_2)(z-p_3)}{K_G(z-z_1)} \tag{4-5}$$

Hence, the most general form of the compensator $F(z)$ considered here is

$$F(z) = \phi F_1(z) F_2(z) \tag{4-6}$$

The $F_2(z)$, if it is used, cancels poles and zeros inside the unit circle. The $F_1(z)$ takes the FIR form given above. Applying the IIR filter to the system in Eq. (4-4), makes the product $F_2(z)G(z) = (z - z_o)$, and then one needs to design $F_1(z)$ to satisfy the stability condition with the effective $G(z)$ composed of only the zero outside the unit circle

$$|1 - \phi(z - z_o)F_1(z)| < 1 \qquad \forall e^{i\omega T} \tag{4-7}$$

After designing $F_1(z)$ and $F_2(z)$, one can multiply the result by a gain ϕ. A reduced gain can be used to slow down the learning for improved robustness if desired. But a new use of this gain is presented, which improves the performance near DC of some RC designs for minimum phase systems without serious penalty at higher frequencies.

4.4 Several System Models

In order to investigate the effectiveness of the various approaches to designing repetitive control systems, and in particular, to evaluate how best to design repetitive controllers for non-minimum phase systems, we consider a set of different models for the feedback control systems whose command inputs are adjusted by repetitive control. These feedback systems are given as continuous time system transfer functions that are fed by a zero order hold on the repetitive control command input, with the output sampled synchronously. Five systems are considered, which are later referred to as Systems 1 to 5:

$$G(s) = \frac{(-s+3)}{(s+5)(s+1)} \quad \text{(System 1)}$$

$$G(s) = \left(\frac{a}{s+a}\right)\left(\frac{\omega_o^2}{s^2 + 2\zeta_o\omega_o s + \omega_o^2}\right) \quad a = 8.8,\ \zeta_o = 0.5, \omega_o = 37 \quad \text{(System 2)}$$

$$G(s) = \frac{(-s+3)(-s+4)}{(s+2)(s+3+4i)(s+3-4i)} \quad \text{(System 3)}$$

$$G(s) = \left(\frac{a}{s+a}\right)\left(\frac{\omega_o^2}{s^2 + 2\zeta_o\omega_o s + \omega_o^2}\right)\left(\frac{\omega_1^2}{s^2 + 2\zeta_1\omega_1 s + \omega_1^2}\right) \quad \zeta_1 = 0.5, \omega_1 = 37 \quad \text{(System 4)}$$

$$G(s) = \left(\frac{s+a}{a}\right)\left(\frac{\omega_o^2}{s^2 + 2\zeta_o\omega_o s + \omega_o^2}\right)\left(\frac{\omega_1^2}{s^2 + 2\zeta_1\omega_1 s + \omega_1^2}\right) \quad \text{(System 5)}$$

As discussed above, we consider designing $F(z)$ based on the full resulting discrete time transfer function, and we also consider use of an IIR filter $F_2(z)$ to cancel all poles and zeros inside the unit circle in the *z*-plane. In the latter case the design of the compensator is based on satisfying the stability condition with what remains of the above $G(z)$ after cancellation, which is only its zeros outside the unit circle. Recall that we define a system to be non-minimum phase based on whether the continuous time transfer function has one or more zeros in the right half plane, which results in one or more zeros on the positive real axis beyond +1 in the *z*-plane (we consider real zeros only). When a minimum phase system fed by a zero order hold is discretized, zeros are introduced outside the unit circle in the *z*-plane if the original pole excess is 3 or more and the sample rate is not very slow, but they are on the negative real axis. We study the above 5 systems to illustrate the possibilities.

System 1: Non-minimum phase with a zero at +1 in the *s*-plane. A slow sample rate of 50Hz is chosen to help make distinctions between different approaches more obvious in figures.

System 2: A minimum phase system. But when discretized there are two zeros on the negative real axis, one at -3.52 outside the unit circle. The other is inside at -0.253, and it might be cancelled by use of $F_2(z)$ if desired. The sample rate is chosen as 200Hz partly to illustrate that repetitive control can easily handle systems that are minimum phase in continuous time, but have zeros outside the unit circle introduced by discretization. RC of this system is then to be compared to RC for non-minumum phase systems in continuous time, in order to see the difficulties introduced.

System 3: A non-minimum phase system with two zeros outside the unit circle on the positive real axis, located at +1.18 and +1.12 in the *z*-plane. A sample rate of 25 Hz is used. As the sample rate increases the zeros get closer to +1, and the number of gains needed for good performance by various RC laws increases. 100Hz is also studied.

System 4: A minimum phase system with two zeros introduced by discretization outside the unit circle on the negative real axis at -14.23 and 1.46 (and two introduced inside at -0.2997 and -0.322). Sample rate is 50Hz.

System 5: A non-minimum phase system with one zero on the positive real axis at $z = 1.09$, and one zero introduced by discretization outside on the negative real axis at -2.68 (also one introduced inside at -0.205). Sample rate is 100Hz. As previously suggested, the non-minimum phase zero from continuous time is near +1, while the introduced zero is far from the unit circle.

4.5 Approaches Considered

4.5.1 Approach 1: Optimizing the Learning Rate over Frequencies Using a Quadratic Penalty Function

This is the approach to designing repetitive controllers that Reference [9] has advocated as the best approach. Previously discussed in section 3.3.1, the objective is to pick the gains of $F(z)$

to make the learning fast, i.e. to minimize the magnitude of $\left[1 - G(e^{i\omega_j T})F(e^{i\omega_j T})\right]$ over all frequencies from zero to Nyquist. To do this we pick a quadratic penalty over a suitably dense set of frequencies ω_j in this range

$$J_1 = \sum_{j=0}^{N} \left[1 - G(e^{i\omega_j T})F(e^{i\omega_j T})\right] W_j \left[1 - G(e^{i\omega_j T})F(e^{i\omega_j T})\right]^* \tag{4-8}$$

The computations made here divide the interval from zero frequency to Nyquist frequency into 180 discrete values, $N = 179$. The asterisk indicates complex conjugate, and W_j is a weighting function usually set to unity for all frequencies, but available to be adjusted to emphasize chosen frequencies. If one employs $F_2(z)$ to cancel everything inside the unit circle, then $G(e^{i\omega_j T})F(e^{i\omega_j T})$ is replaced in the optimization criterion by

$$\left[F_2\left(e^{i\omega_j T}\right)G\left(e^{i\omega_j T}\right)\right]F_1\left(e^{i\omega_j T}\right) = \bar{G}(e^{i\omega_j T})F_1(e^{i\omega_j T}) \tag{4-9}$$

and $F_1(z)$ is the FIR filer whose gains are being optimized.

For the usual minimum phase systems, the magnitude of $G(e^{i\omega_j T})$ is large at low frequencies, and decays very substantially as the frequency gets high. It should be small approaching Nyquist frequency when using an appropriately chosen sample rate that has small aliasing. Hence, this cost function exhibits some extra emphasis on producing fast learning at lower frequencies. This is in fact quite desirable for most minimum phase systems, making this a good design approach.

However, non-minimum phase systems with a zero z_o outside the unit circle near +1, have small frequency response at DC and low frequency produced by the small factor in the numerator $|z - z_o| = |exp(i\omega T)\text{-}z_o|$, and the optimization de-emphasizes the learning rate near zero frequency, resulting in poor performance for non-minimum phase systems. One can adjust the weights W_j aiming to improve the performance at low frequency, but examples shown below

suggest this is not very effective. Note that this thinking about what frequencies are emphasized in the cost function changes when one cancels all dynamics inside the unit circle and designs $F_1(z)$ for the zero(s) outside the unit circle.

The optimal design for the gains in $F(z)$ (or in $F_1(z)$) are easily obtained for such a quadratic criterion. Let *x* be the column vector of the gains a_1 to a_n in the FIR filter equation (4-3). And write the system transfer function frequency response as $G(e^{i\omega_j T}) = M_G(\omega_j)\exp(i\varphi_G(\omega_j))$. Then the cost function can be rewritten in the following form

$$J_1 = \frac{1}{2}x^T Ax + b^T x + d$$

$$A = \sum_{j=0}^{N} A_j \quad \bar{b} = \sum_{j=0}^{N} b_j \quad d = \sum_{j=0}^{N} d_j \qquad d_j = W_j \tag{4-10}$$

$$A_j = W_j M_G^2(\omega_j)\begin{bmatrix} 1 & \cos(\omega_j T) & \cdots & \cos\left((n-1)\omega_j T\right) \\ \cos(\omega_j T) & 1 & \cdots & \cos\left((n-2)\omega_j T\right) \\ \vdots & \vdots & \ddots & \vdots \\ \cos\left((n-1)\omega_j T\right) & \cos\left((n-2)\omega_j T\right) & \cdots & 1 \end{bmatrix}$$

$$b_j = W_j M_G(\omega_j)\begin{bmatrix} \cos\left((m-1)\omega_j T + \varphi_G(\omega_j)\right) \\ \cos\left((m-2)\omega_j T + \varphi_G(\omega_j)\right) \\ \vdots \\ \cos\left((m-n)\omega_j T + \varphi_G(\omega_j)\right) \end{bmatrix}$$

The optimal gains are the solution of the linear equation in the *n* unknown gains $Ax = b$.

4.5.2 Approach 2: Matching the Inverse of the Frequency Response Using a Quadratic Penalty Function

Reference [10] also examined another cost function aiming to make $F(z)$ match the inverse of the frequency response of the system

$$J_2 = \sum_{j=0}^{\infty} \left[F\left(e^{i\omega_j T}\right)\text{-}G^{-1}\left(e^{i\omega_j T}\right)\right] \overline{W}_j \left[F\left(e^{i\omega_j T}\right)\text{-}G^{-1}\left(e^{i\omega_j T}\right)\right]^* \quad (4\text{-}11)$$

In that reference, the cost function of Approach 1 was preferred for the minimum phase systems considered there. For minimum phase systems, the frequency response gets very small approaching Nyquist frequency, and hence $G^{-1}\left(e^{i\omega_j T}\right)$ can be very large. Then the cost function will pay particular attention to producing a good FIR design for high frequencies, perhaps at the expense of poor performance at low frequencies.

Note that this cost function, and the cost function of Approach 1 are equivalent. If one picks the weights W_j in Approach 1 to equal $W_j = G^{-1}(e^{i\omega_j T})\overline{W}_j G^{-1}(e^{-i\omega_j T})$, then the cost function becomes the cost function of Approach 2. In order to find the optimal FIR gains *x* for the Approach 2 cost function, one could do a parallel development to that above, or simply use the weight function given here to obtain the solution from the above $Ax = b$ equation using (4-10).

4.5.3 Approach 3: Min-Max Optimization of the Learning Rate over Frequencies

Approach 1 aims to minimize $|1 - G(\exp(i\omega T))F(\exp(i\omega T))|$ in the stability condition, minimizing the sum of squares for each frequency. This aims for a good learning rate at all frequencies in a least squares sense. If there is difficulty satisfying the stability condition, i.e. making this less than unity for all frequencies from zero to Nyquist, it is very natural to think of picking the gains in *x* to minimize the maximum value over all frequencies, trying to get it less that unity everywhere. The cost used in Approach 1 is quadratic and finding the minimum is easy, one simply solves a linear equation $Ax = b$. Minimizing the maximum value generally is a harder optimization problem and require more computation. This section also shows how one can solve this Min-Max problem. Now that an efficient method is available to minimize this logical cost

function, we evaluate whether this is a better design approach than Approach 1 – examining its performance for a range of systems. We discover that the answer is no, for minimum phase systems this new approach is not preferred, but yes for non-minimum phase systems it can address the important issues.

The desired optimization is

$$J_3 = \min_x \max_j \left| \left[1 - F(e^{i\omega_j T}) G(e^{i\omega_j T}) \right] V_j \right| \tag{4-12}$$

where V_j is a weight function, and x is again the column vector of the gains $a_0, a_1, \ldots, a_N$ in the FIR filter. This optimization problem is equivalent to the following

$$\begin{aligned} &\min_x \max_j \quad \left| 1 - F\left(e^{i\omega_j T}\right) G\left(e^{i\omega_j T}\right) \right|^2 W_j \\ &= \min_x \max_j \quad \left[1 - F(e^{i\omega_j T}) G(e^{i\omega_j T}) \right] W_j \left[1 - F(e^{i\omega_j T}) G(e^{i\omega_j T}) \right]^* \\ &= \min_x \max_j \quad \frac{1}{2} x^T A_j x + b_j^T x + d_j \end{aligned} \tag{4-13}$$

where $W_j = V_j^2$. Again we convert to another equivalent formulation

Minimize: $\min_x \bar{M}$ (4-14)

Subject to: $\frac{1}{2} x^T A_j x + b_j^T x + d_j \leq \bar{M} \quad j = 0,1,2, \ldots, N$ (4-15)

And finally another equivalent reformulation

Minimize: $\min_x f^T \bar{x}$ (4-16)

Subject to: $\frac{1}{2} \bar{x}^T \bar{A}_j \bar{x} + \bar{b}_j^T \bar{x} + d_j \leq 0 \qquad j = 0,1,2, \ldots, N$ (4-17)

where

$$\bar{x} = \begin{bmatrix} x \\ \bar{M} \end{bmatrix} \quad f = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \quad \bar{A}_j = \begin{bmatrix} A_j & 0 \\ 0 & 0 \end{bmatrix} \quad \bar{b}_j = \begin{bmatrix} b_j \\ -1 \end{bmatrix} \qquad \bar{d}_j = \frac{1}{2}$$

This final problem formulation is a special case of a Quadratically Constrained Quadratic Programming problem (QCQP) for which there are efficient algorithms. It is actually a quadratically constrained linear programming problem. Solutions are obtained here using the CVX programming package, References [11-13].

Of course one can apply Approach 3 when one uses $F_2(z)$ to cancel all poles and zeros inside the unit circle. However, just as in Approach 1, one very attractive feature of this approach is that one can perform the optimization based on experimental frequency response data, without having to generate the transfer function model with the poles and zeros needed to design the IIR filter.

This Approach 3 is seen to have some good properties for repetitive control of non-minimum phase systems. The small feedback control system magnitude frequency response near DC produced by the zero near +1, creates slow learning, often impractically slow learning, at low frequencies in the cost function of Approach 1, corresponding to a maximum of the left hand side of (3-10). The Min-Max criterion pushes the maximum value down, improving the originally poor performance at low frequency. Of course, this is done at the expense of slower learning at higher frequencies. Some other methods can compete, but the other methods need to use $F_2(z)$, and hence preclude designing directly from frequency response data.

4.5.4 Approaches 4 and 5: Compensator Design Based on Taylor Series Approximation of Inverse Transfer Function

The original approach to designing RC compensators by Taylor series, References [9, 14], do the Taylor expansions separately for the reciprocal of each zero outside the unit circle. It is not necessary to use an $F_2(z)$ to cancel the poles and zeros inside the unit circle, but here we consider

that this is done, and one only needs to consider the factors that are the reciprocals of zeros outside the unit circle when designing $F_1(z)$. Consider the Taylor series expansion

$$\frac{1}{1+\hat{z}} = 1 - \hat{z} + \hat{z}^2 - \hat{z}^3 + \ldots = \sum_{k=0}^{\infty} (\hat{z})^k \approx 1 - \hat{z} + \hat{z}^2 - \hat{z}^3 + \ldots + \hat{z}^r \tag{4-18}$$

The series is convergent if $|\hat{z}| < 1$. Consider the reciprocal of one zero outside the unit circle $1/(z-a)$ using $\hat{z} = -z/a$, then

$$\frac{1}{z-a} = \left(\frac{1}{a}\right)\left(\frac{1}{1+\hat{z}}\right) \tag{4-19}$$

and one can now apply the above expansion. If $r = 1$, the design will place one compensator zero at the mirror image location to the existing zero. For higher values of r, more zeros are place evenly spaced around a circle centered at the origin and going through the original zero. This behavior gave insight into the observed placement of zeros when applying Approach 1 to design compensators. Reference [14] suggests using the remainder term in the Taylor series expansion to help decide what value of r to use. When there are two zeros outside, each one is treated independently with its own truncated Taylor series expansion, and the compensator is then obtained by multiplying the two expansions together. One can pick the value of r independently for each zero to produce similar error levels for each, and this is in fact rather important. Note that the closer the zero is to the unit circle, the slower is the convergence of the above series, and hence, one needs more zeros introduced in the compensator to achieve the same level of performance. This observation is consistent with the observation that one needs more zeros in the compensators for non-minimum phase systems with zeros near +1 than for systems where zeros are introduced on the negative real axis by the discretization process, with locations usually much further from -1. Hence, this is the source of some of the difficulty in achieving good RC performance for non-minimum phase systems.

One of the contributions of this work is to improve on the above Taylor series expansion method when there are two or more zeros outside the unit circle, as in Systems 3, 4, and 5 above. Consider two zeros outside at a_1 and a_2, and everything else is cancelled by $F_2(z)$. Then

$$\begin{aligned}[F_2(z)G(z)]^{-1} &= \frac{1}{(1-a_1)(1-a_2)} = \frac{1}{(-a_1)(-a_2)}\frac{1}{\left(1-\frac{z}{a_1}\right)}\frac{1}{\left(1-\frac{z}{a_2}\right)} \\ &= \frac{1}{a_1 a_2}\left[1+\frac{z}{a_1}+\left(\frac{z}{a_1}\right)^2+\cdots\right]\left[1+\frac{z}{a_2}+\left(\frac{z}{a_2}\right)^2+\cdots\right] \\ &= (a_1 a_2)^{-1}[1+({a_1}^{-1}+{a_2}^{-1})z+({a_1}^{-2}+{a_1}^{-1}{a_2}^{-1}+{a_2}^{-2})z+\ldots\end{aligned} \tag{4-20}$$

Then $F_1(z)$ is produced by truncating the last expression at the chosen number of terms. The roots of the resulting polynomial are not the same as the set of roots from the truncated series in each of the bracketed terms in the middle line of the equation, i.e. the zeros are not the same as those obtained by the original Taylor method above, treating each square bracket term separately. Treating them separately required choice of how many terms to use in each expansion, and there is guidance concerning how small the truncation error is for each choice. When using the combined method one does not need to make two separate choices.

The term Approach 5 will be used for this improved Taylor method, while Approach 4 is used for the original Taylor method described above. Note that if there is only one zero outside the unit circle, Approaches 4 and 5 are identical. Approach 5 is an improvement over Approach 4 when there are multiple zeros outside.

4.5.5 Approach 6: Compensator Design Based on Phase Cancellation (Tomizuka)

This design approach is perhaps the best known design method for repetitive control, Reference [15]. It requires use of an IIR filter $F_2(z)$ to cancel all poles and zeros inside the unit

circle. The design only attempts to cancel the phase produced by the zero or zeros outside the unit circle, and does not attempt to cancel the magnitude change. Each zero z_o outside the unit circle is handled independently, by introducing a zero at the reciprocal location and a pole at the origin, followed by normalization. For a factor $(z - z_o)$ one makes $F_1(z)$ create the term

$$C(z - z_o)(z_o - z^{-1}) = Cz_o[(z - z_o)(z - {z_o}^{-1})]/z \tag{4-21}$$

Set $z = \exp(i\theta)$ to find the frequency response

$$C(z - z_o)(z^{-1} - z_o) = C[1 - 2\cos\theta\, z_o + z_o^2] \tag{4-22}$$

When $\theta = 0$ this equals $C(z_o - 1)^2$, and when $\theta = \pi$ or Nyquist frequency, it equals $C(z_o + 1)^2$. Both values are positive, and the derivative is monotonic betweens since the derivative with respect to θ is equal to $2C \sin\theta\, z_o$. If the zero is on the negative real axis outside the unit circle, then the maximum value is at $\theta = 0$ or DC, and one picks the normalization C to give DC gain of unity which produces fast learning at DC, but slow learning at high frequencies. However, if the zero is from a non-minimum phase continuous time system, and on the positive real axis in the z-plane, then the maximum is at Nyquist frequency and one picks C to make the gain unity at this frequency. This results in negligible learning near DC which makes this approach not work for typical non-minimum phase systems. System (5) has a zero on the positive real axis and a zero on the negative real axis, and this situation requires one to use the product of one of the above factors normalized at DC and another normalized at Nyquist.

4.6 Numerical Studies for Systems 1 Through 5

This section numerically studies the performance of the 6 Approaches as applied to the 5 Systems. Each is studied without cancelling the poles and zeros inside the unit circle, and then is studied with this cancellation with the IIR filter. In the first case, we consider only Approaches 1,

2, and 3. Approach 6 cannot be used without cancelling. And the Taylor series Approaches 4 and 5 will at least cancel the poles inside the unit circle. They do offer the option of cancelling the effect of a zero inside by including more zeros inside instead of placing a pole underneath, but we do not consider this option. If the zeros and poles are cancelled, the titles of the figures given below indicates this, and then the plots present the performance for all 6 Approaches.

System (1): Figure 4-3 shows Approaches 1, 2, 3 performance without canceling poles and zeros inside the unit circle. Approach 1 has rather fast learning at almost all frequencies, but learning at DC is very slow. Approaches 2 and 3 fix this with rather uniform learning rate at all frequencies. Figure 4-4 gives the associated phase of $F(z)G(z)$ which has appreciable oscillations except for Approach 1, and Fig. 4-5 gives the corresponding polar frequency response. Figure 4-6 examines the possible benefit of turning down the overall gain ϕ to 0.75. It does not help Approach 1, but on the average Approaches 2 and 3 improve. This can help somewhat with robustness to model error also. Note the number of gains being used in these designs. Figures 4-7 and 4-8 give results when one cancels the two poles inside the unit circle, and uses either 11 or 19 zeros instead of 29. All Approaches are plotted. We observe that Approach 6 cannot be used for non-minimum phase systems, and also Approach 2 is not working for 11 zeros. It appears that with enough gains, one can make Approach 1 work, with fast learning at nearly all frequencies, but slow near DC. Approach 3 minimizes the maximum error and improve performance at DC but one pays for it with slower learning everywhere except near DC. Approaches 3, 4, and 5 give similar results. Decreasing ϕ in Fig. 4-9 helps Approaches 2, 3, and 4-5. Figure 4-10 gives the corresponding zero locations, with all but Approach 1 being nearly the same, and Approach 1 having the zeros further from the origin. The cost function for Approach 1 has weight factor W_j that can give extra emphasis to minimizing at chosen frequencies. Figure 4-11 increases the weight by a factor of 100,

starting from zero up to 10%, 20%, and 50% Nyquist. We observe that this seems remarkably ineffective at reducing the plot near DC, and can cause trouble at higher frequencies. We conclude that it is not obvious how one can employ this design parameter to get good performance near DC.

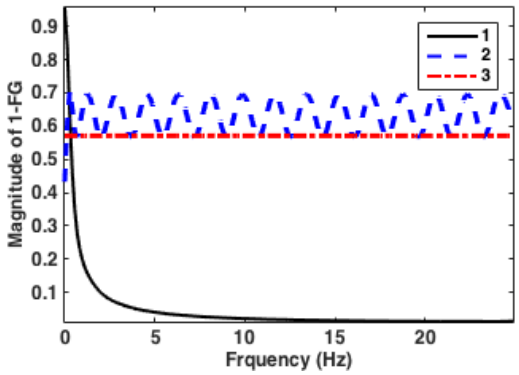

Figure 4-3. Magnitude frequency response of 1-$F(z)G(z)$ of System (1) (n=30, m=30).

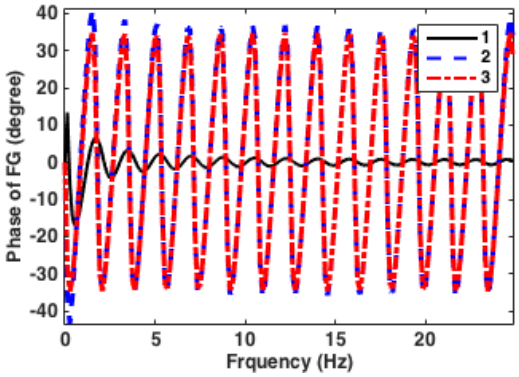

Figure 4-4. Phase frequency response of $F(z)G(z)$ System (1) (n=30, m=30).

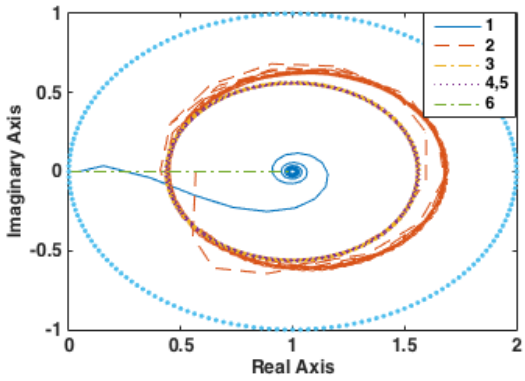

Figure 4-5. Polar plot corresponding to Figs. 4-3 and 4-4.

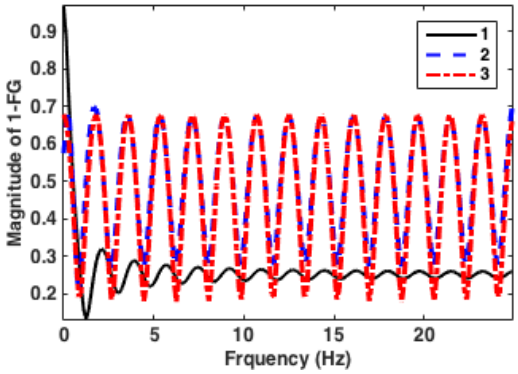

Figure 4-6. Fig. 4-3 with overall gain ϕ = 0.75.

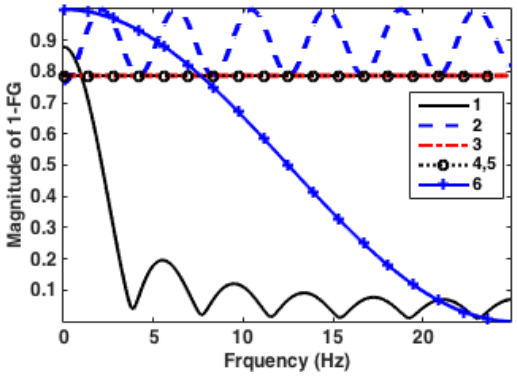

Figure 4-7. Fig. 4-3 with cancelation poles and zero inside the unit circle (n=12, m=12).

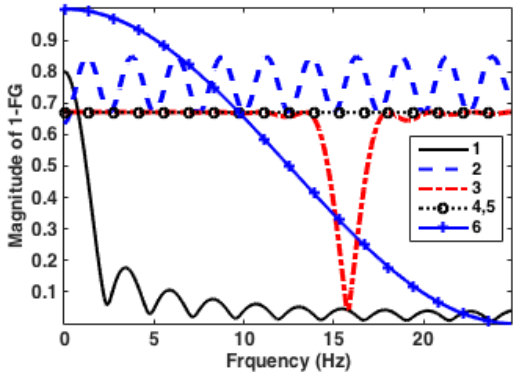

Figure 4-8. Fig. 4-7 with *n*=20 and *m*=20.

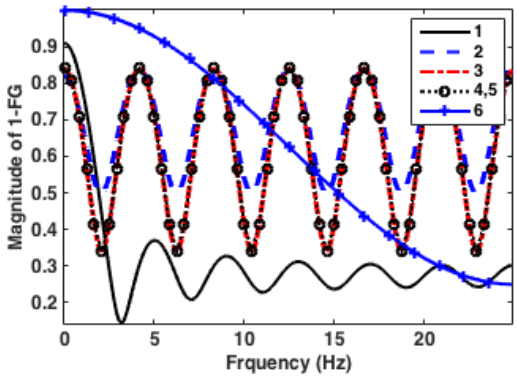

Figure 4-9. Fig. 4-7 with overall gain ϕ= 0.75.

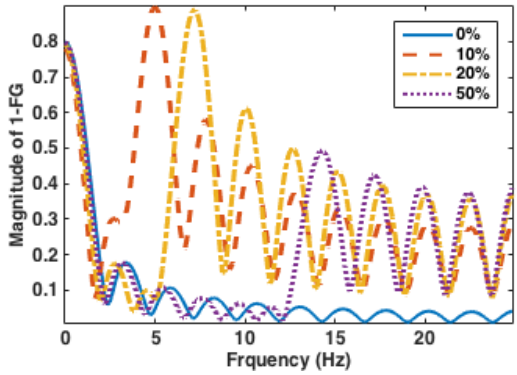

Figure 4-11. Magnitude frequency response of 1-*F*(z)*G*(*z*) of System (1) using Approach 1 with different weight.

System (2): Figures 4-12 to 4-16 consider System 2 which is a minimum phase system included to show the difference in ease of designing a good RC for minimum phase compared to the non-minimum phase experience with System (1) above. Figure 4-12 uses $n = 5$ allowing 4 zeros in the compensator. There are 3 poles inside, so the optimization might do something similar to cancelling these with 3 of the zeros, and this requires using one pole at the origin $n - m = 1$ so that the number of poles inside equals the number of zeros inside. Then there is one zero left over to try to handle the effect of the zero outside the unit circle at -3.52. Hence, this is a very simple compensator that requires just adding up 5 gains times 5 errors stored from the previous period,

and its only action to handle the zero outside the unit circle is to place one zero outside to compensate. Figure 4-12 shows that Approaches 1, 2, and 3 all work well (note vertical axis scale), and Approach 1 works the best up until high frequencies. One usually employs a zero-phase filter cutoff that stops the learning at high frequencies because of model inaccuracy there. Also, slower learning usually makes convergence more robust to model error. Hence, Approach 1 is to be preferred.

Figure 4-13 introduces the IIR filter to cancel poles and zeros inside. A fair comparison uses $n = m = 2$ corresponding to two gains producing one zero outside, and the error levels are similar, but this time one can use Approach 6 that is better up to about 20 Hz, and then becomes much slower as the frequency goes up. Figure 4-14 investigates a new option. If one particularly wants to have fast learning at DC, we can normalize the gain ϕ to produce zero error there, just as Approach 6 does. The result for Approach 1 comes rather near to matching Approach 6 for frequencies near DC, and is dramatically better than Approach 6 at all higher frequencies.

Figure 4-15 considers the same problem as Figure 4-12, without cancelation inside, but allows the compensator to have 8 zeros. The plots no longer wiggle up and down, but the learning rate is roughly the same over the full frequency range, and the Min-Max criterion starts to look like a straight line. On the other hand, Fig. 4-16 presents the results using the IIR, with the appropriate number of gains for a fair comparison, $n = m = 6$. It is clear that with just 5 zeros, and this minimum phase system, repetitive control using all Approaches 1 through 5 can give nearly perfect results, getting close to zero error in one iteration, while Approach 6 has much slower learning.

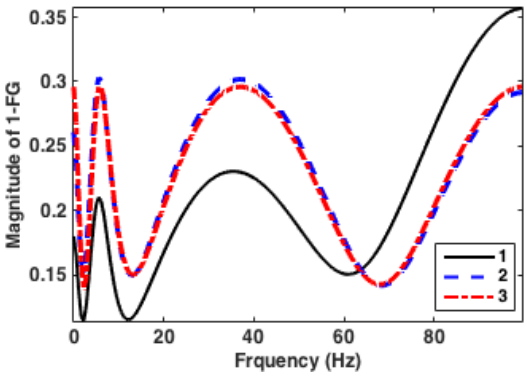

Figure 4-12. Magnitude frequency response of 1-*F*(z)*G*(*z*) of System (2) when *n*=5 and *m*=4.

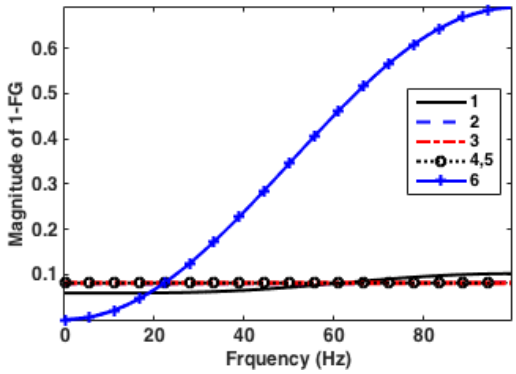

Figure 4-13. Magnitude frequency response of 1-*F*(z)*G*(*z*) of System (2) when poles and zeros inside are canceled (*n*=2, *m*=2).

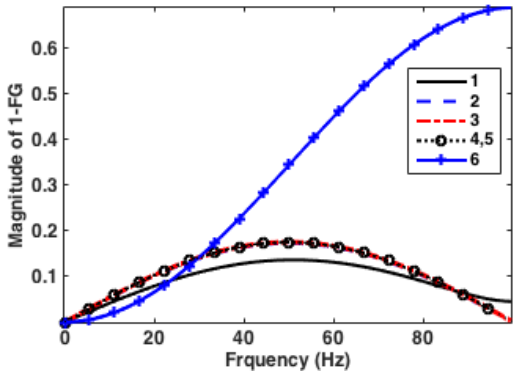

Figure 4-14. Fig. 4-13 but overall gain at DC is normalized.

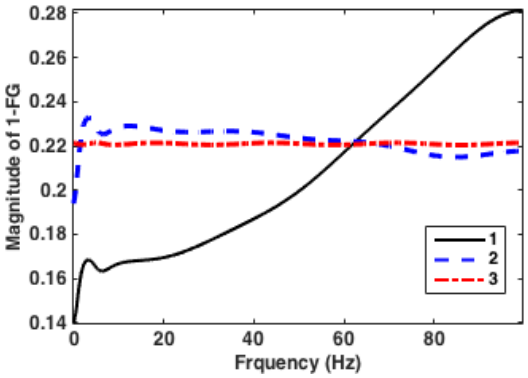

Figure 4-15. Fig.12 with more gain (*n*=9, *m*=8).

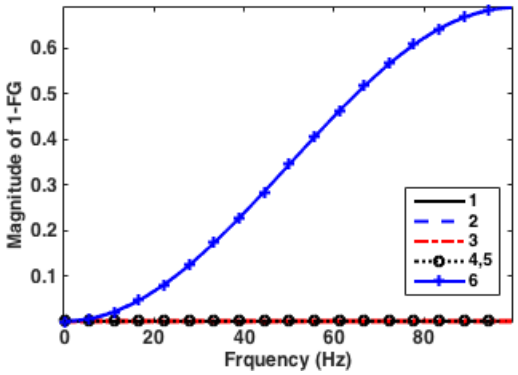

Figure 4-16. Fig. 4-13 with more gains (*n*=6, *m*=6).

System (3): Figure 4-17 considers System (3) with 2 zeros on the positive real axis, and with a 25Hz sample rate. Approach 2 does not make a stable system when using 19 zeros, Approaches 1 and 3 do, with 3 substantially better near DC, and substantially worse at higher frequencies. Figure 4-18 cancels the 3 poles inside the unit circle with an IIR, and then uses 16 zeros outside for the FIR design. Performance is similar for Approaches 1 and 3, and Approaches 2, 4 and 5 (which now give different results), and 6 produce unstable designs. Figure 4-19 increases the sample rate which one expects will accentuate the problem, making the zeros outside much closer to the unit circle. Only Min-Max works at all frequencies when using 30 gains.

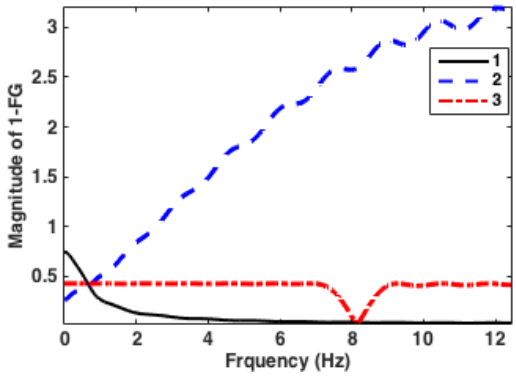

Figure 4-17. Magnitude frequency response of 1-$F(z)G(z)$ of System (3) (n=20, m=20).

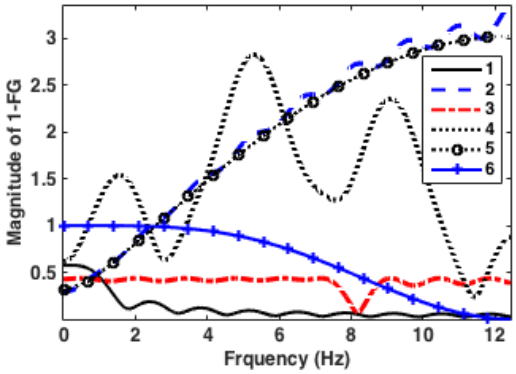

Figure 4-18. Magnitude frequency response of 1-$F(z)G(z)$ of System (3) when cancel inside unit circle and using (n=17, m=17).

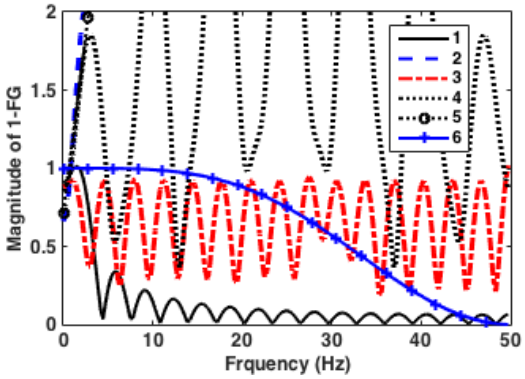

Figure 4-19. Same as Fig. 4-18 with high sample rate at 100 Hz and more gains (n=30, m=30).

System (4): Figures 4-20 through 4-23 treat minimum phase System (4) with two zeros introduced on the negative real axis. Figure 4-20 shows the learning rate for Approaches 1, 2, and 3. There are 4 poles inside the unit circle and 2 zeros, so we let $n - m = 2$ to allow 4 zeros to approach cancelling the poles inside, and add two compensator poles inside to make the number of zeros and poles inside match. One observes that all approaches work, note the vertical axis scale. But the approach advocated by the authors, Approach 1, works best except very near Nyquist frequency.

Figure 4-21 compares learning rates for all Approaches. Approach 6 gives the fastest learning rate near DC, but all other approaches learning faster at somewhat higher frequencies. Again, among the other approaches, Approach 1 is the best for nearly all frequencies until near Nyquist. Figure 4-22 again investigates the new option for improving low frequency learning in minimum phase systems, normalizing the gain ϕ to zero DC error, just as in Approach 6. The Approach 1 result nearly matches Approach 6 near DC, and is dramatically better than Approach 6 at all higher frequencies. Figure 4-23 shows the locations of the 6 zeros of the compensator outside the unit circle, showing that the locations are nearly the same for Approaches 2, 3, and 5, while the zeros for Approach 1 are at a larger radial distance and not on a circle.

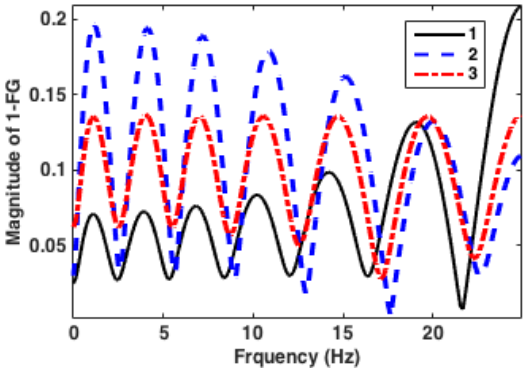

Figure 4-20. Magnitude frequency response of 1-*F*(z)*G*(*z*) of System (4) (*n*=12, *m*=10).

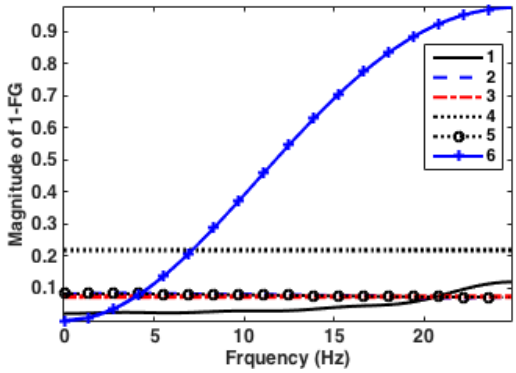

Figure 4-21. Magnitude frequency response of 1-F(z)G(z) System (4) when cancel inside unit circle (n=7, m=7).

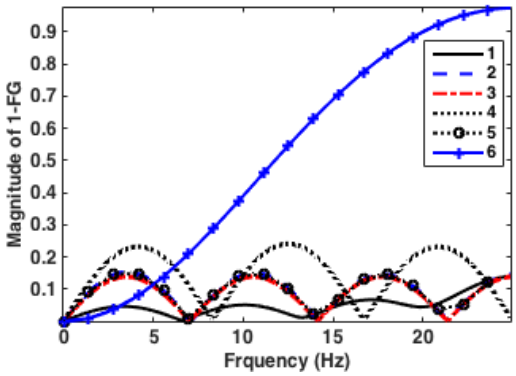

Figure 4-22. Same as Fig. 4-21 but overall gain is normalized at DC.

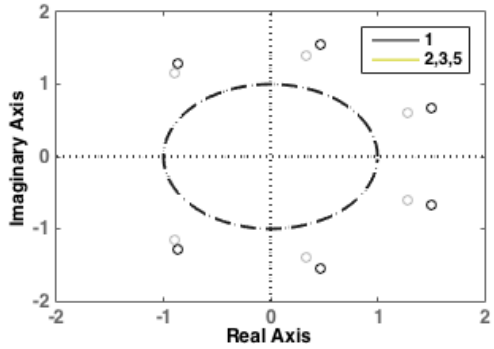

Figure 4-23. Zero location of compensators corresponding to Fig. 4-21 using Approaches 1, 2, 3, and 5.

System (5): This system has a zero outside the unit circle on both the positive and the negative real axis. Without cancelling, Fig. 4-24, one can use Approach 2 or the Min-Max of Approach 3 with 6 zeros in the FIR. When the IIR filter is used, and the appropriate number of gains matched, the

results are shown in Fig. 4-25. Approach 6 of course does not learn at DC, but this time, learning is slow at high frequencies because of the 2 normalizations needed, one at Nyquist and one at DC. The two Taylor approaches are distinct, Approach 4 fails, and Approaches 5 and 2 are nearly equivalent and give relatively good designs. Note that these two are similar but outperform the Min-Max design at nearly all frequencies and only pay for this with slightly slower learning at DC. So this example shows that while the Min-Max Approach 3 seems generally effective for non-minimum phase systems, allowing reasonably good learning rate at DC, the modified Taylor Approach 5 as well as the Approach 2 can be competitive, provided one uses the IIR to cancel. Figure 4-26 shows the compensator for Approach 5 simply does a mirror reflection of the 2 system zeros outside the unit circle. Figure 4-27 shows where the 2 zeros are placed using the other design approaches, which are totally different. Again Approach 1 places the zeros furthest from the origin. Figure 4-28 shows that with enough gains all approaches work except for Tomizuka, with similar comments to Figure 4-27. Figure 4-29 shows that the new Taylor method chooses to use 7 out of the 8 zeros to address the zero near +1, and only one zeros to address the zero at -2.68. Figure 4-30 shows the corresponding zero locations of Figure 4-28.

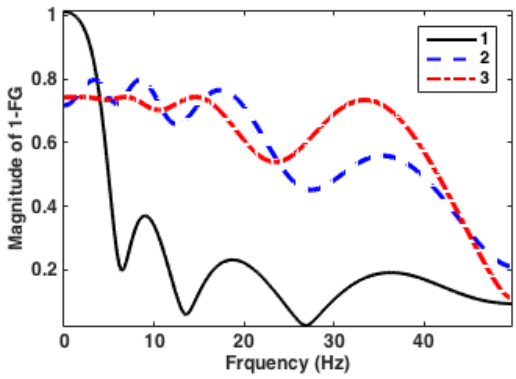

Figure 4-24. Magnitude frequency response of 1-F(z)G(z) of System (5) (n=7, m=6).

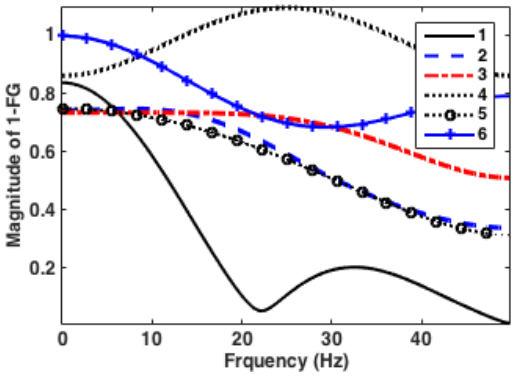

Figure 4-25. Magnitude frequency response of 1-$F(z)G(z)$ of System (5) when cancel inside unit circle and using low number of gains (n=3, m=3).

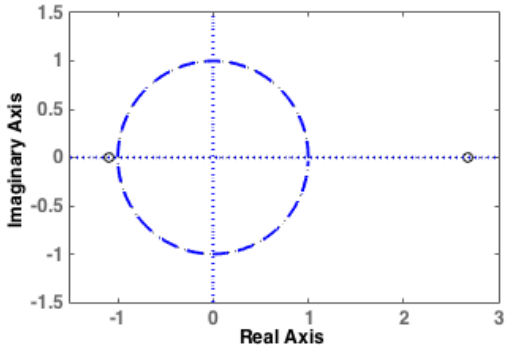

Figure 4-26. Zero location of compensator corresponding to Fig. 4-25 using Approach 4.

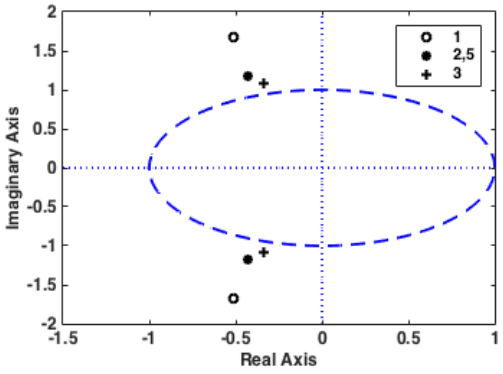

Figure 4-27. Zero location of compensators corresponding to Fig. 4-25 using Approaches 1, 2, 3, and 5.

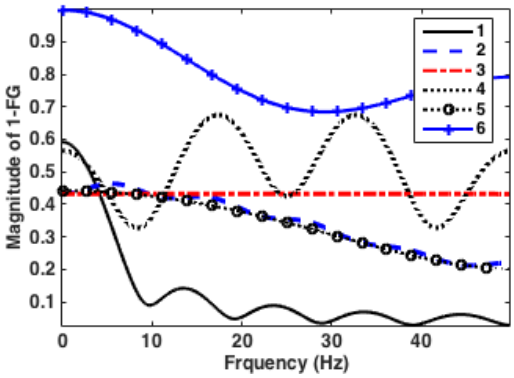

Figure 4-28. Magnitude frequency response of 1-$F(z)G(z)$ of System (5) when cancel inside unit circle and using larger number of gains (n=9, m=9).

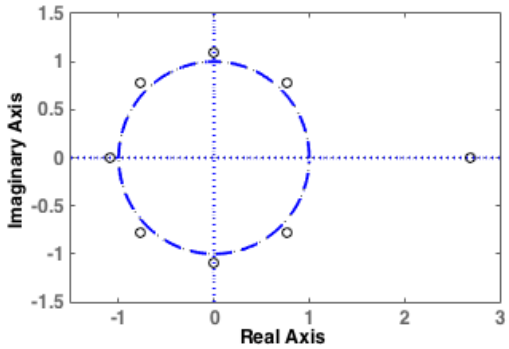

Figure 4-29. Zero location of compensator corresponding to Fig. 4-28 using Approach 4.

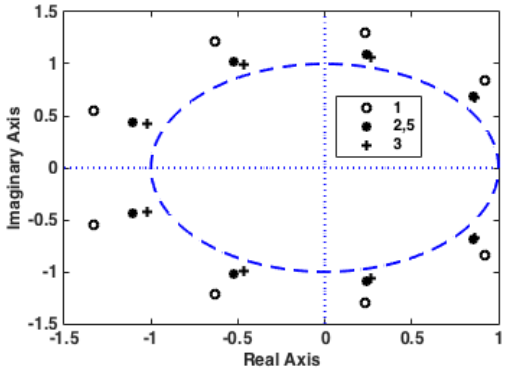

Figure 4-30. Zero location of compensators corresponding to Fig. 4-28 using Approaches 1, 2, 3, and 5.

4.7 Conclusions

In repetitive control, non-minimum phase systems usually have a problem of slow learning at low frequencies due to a near pole zero cancellation at +1. In this chapter, five RC design methods are investigated to determine how well the methods perform in minimum phase and non-minimum phase systems that contain different locations of poles and zeros. Approach 1, optimizing the learning rate over frequencies using a quadratic penalty function on the difference from unity of the product of compensator times the system transfer function, performs well in minimum phase systems - learns fast at low frequencies and learns slowly at high frequencies, but has a problem of slow learning at frequencies near DC when applied to non-minimum phase systems. Approach 2, matching compensator with the inverse of the frequency response using a quadratic penalty function, can address some non-minimum phase problems but can fail in others. Approach 3, using a Min-Max optimization over all frequencies, normally provides a rather uniform learning rate through all frequencies. Thus, compared to Approach 1, the Min-Max design learns slower at low frequencies for minimum phase systems but improves performance of learning rate at DC and low frequencies for non-minimum phase systems. Approaches 4 and 5, consider compensator designs based on Taylor series approximations of the inverse transfer function, and can compete with the approach 3 in some cases. Approach 5 also behaves similar to Approach 2 with learning gain adjusted. However, approaches 4 and 5 require extra IIR part of the compensator that requires knowledge of the zero locations inside the unit circle. Approach 6, the compensator design based on phase cancellation presented by Tomizuka, can learn well for minimum phase systems, but because of the normalization needed, there is almost no learning at DC for non-minimum phase systems.

Among the design methods considered in this dissertation, the Min-Max method is the most effective method for non-minimum phase systems. The problem can be formulated in QCQP form and easily solved by using the CVX package. This method is further investigated in Chapter 5 to address practical problems in applications.

4.8 References

[1] C. T. Freeman, P. L. Lewin, and E. Rogers, "Experimental Evaluation of Iterative Learning Control Algorithms for Non-Minimum Phase Plants," *international Journal of Control,* Vol. 78, 2005, pp. 826-846.

[2] Z. Cai, C. Freeman, E. Rogers, and P. Lewin, "Reference Shift Iterative Learning Control for a Non-minimum Phase Plant," *Proceeding of the 2007 American Control Conference*, 2007, New York City, NY, 2007, pp. 558-563.

[3] D. H. Owens and B. Chu, "Modelling of Non-Minimum-Phase Effects in Discrete-Time Norm Optimal Iterative Learning Control," *International Journal of Control*, Vol. 83, 2010, pp. 2012-2027.

[4] D. H. Owens, B. Chu, E, Rogers, C. T. Freeman, and P. L. Lewin, "Influence of Nonminimum Phase Zeros on the Performance of Optimal Continuous-Time Iterative Learning Control," *IEEE Transactions on Control Systems*, Vol. 22, 2014, pp. 1151-1158.

[5] S. Gunnarsson and M. Norrlof, "On the Design of ILC Algorithms Using Optimization," *Automatica*, Vol. 37, 2001, pp. 2011-2016.

[6] J. Ghosh and B. Paden, "Iterative Learning Control for Nonlinear Nonminimum Phase Plants with Input Disturbances," *Proceedings of the American Control Conference*, San Diego, California, 1999, pp. 2584-2589.

[7] L. Wang, C. T. Freeman, E. Rogers, and D. H. Owens, "Experimentally Validated Continuous-Time Repetitive Control of Non-Minimum Phase Plants with a Prescribed Degree of Stability," *Control Engineering Practice,* Vol. 18, 2010, pp.1158-1182.

[8] K. Astrom, P. Hagander, and J. Strenby, "Zeros of Sampled Systems," *Proceedings of the 19th IEEE Conference on Decision and Control*, 1980, pp. 1077-1081.

[9] R. W. Longman, "On the Theory and Design of Linear Repetitive Control Systems," *European Journal of Control*, Special Section on Iterative Learning Control, Guest Editor Hyo-Sung Ahn, Vol. 16, No. 5, 2010, pp. 447-496.

[10] B. Panomruttanarug and R. W. Longman, "Repetitive Controller Design Using Optimization in the Frequency Domain," *Proceedings of the 2004 AIAA/AAS Astrodynamics Specialist Conference*, Providence, RI, August 2004.

[11] M. Grant and S. Boyd, "CVX: Matlab Software for Disciplined Convex Programming, version 2.0 beta," http://cvxr.com/cvx, September 2013.

[12] M. Grant and S. Boyd, "Graph Implementations for Nonsmooth Convex Programs," *Recent Advances in Learning and Control* (a tribute to M. Vidyasagar), V. Blondel, S. Boyd, and H. Kimura, editors, pages 95-110, *Lecture Notes in Control and Information Sciences*, Springer, 2008. http://stanford.edu/~boyd/graph_dcp.html.

[13] S. Boyd and L. Vandenberghe. *Convex Optimization*, Cambridge University Press, Cambridge, 2004.

[14] K. Xu and R. W. Longman, "Use of Taylor Expansions of the Inverse Model to Design FIR Repetitive Controllers," *Advances in the Astronautical Sciences*, Vol. 134, 2009, pp. 1073-1088.

[15] K-K. Chew and M. Tomizuka, "Steady-State and Stochastic Performance of a Modified Discrete-Time Prototype Repetitive Controller," 1988 ASME Winter Annual Meeting, December 1988, also in ASME *Journal of Dynamic Systems, Measurement and Control*, March 1990, pp. 35-41.

Chapter 5

Min-Max Merged with Quadratic Cost for Repetitive Control of Non-Minimum Phase Systems

5.1 Introduction

The previous chapter developed a method to directly address the main difficulty encountered in designing RC for non-minimum phase systems, namely the slow learning at low frequencies that is produced by a near pole-zero cancellation at DC. The amount of decrease of the error each period is approximately represented by $\left|1 - F\left(e^{i\omega T}\right)G\left(e^{i\omega T}\right)\right|$ at each frequency ω, and the optimization criterion is converted to minimizing the maximum value of this function over frequencies from zero to Nyquist, Reference [1]. The approach is reasonably effective at improving the learning near DC, but is perhaps not the ideal optimization criterion at higher frequencies. The purpose of the investigation in this chapter is to obtain the benefits of the Min-Max optimization criterion near DC, but improve the performance at higher frequencies, especially aiming for improved robustness. In order to give a detailed and complete understanding of the two approached developed in this chapter, the next section starts from the most basic considerations in the design of repetitive controllers, and develops the logic that leads to the approaches investigated here. This is done in the form of a series of Remarks which culminate in the candidate approaches.

5.2 The Big Picture and the Problems Addressed

REMARK 1: Consider the general repetitive control system in the lower part of Figure 2-1 and consider using a repetitive control law $R(z) = \emptyset G^{-1}(z)/(z^p - 1)$ which uses the inverse of the discrete time closed loop system transfer function. The transform of the error $E(z)$ satisfies $[(z^p - 1) + \emptyset G^{-1} G]E = [z^p - 1][Y_D - V]$. This represents a homogeneous difference equation since the functions on the right hand side are periodic with p time steps. The associated characteristic polynomial is $z^p = (1 - \phi)$. The root locus starts with p roots evenly spaced on the unit circle when $\phi = 0$, and they move radially inward as ϕ increases, all reaching the origin when $\phi = 1$. We conclude that this RC law converges for unity gain had deadbeat convergence to zero tracking error in a finite number of time steps. Use of a repetitive control compensator $F(z) = G^{-1}(z)$ appears to be the ideal design. This RC law performs as intended for some systems, but for pole excess of 3 or more in $G(s)$ and reasonable sample rates, $G(z)$ will have zeros outside the unit circle, making the repetitive control law unstable.

REMARK 2: To address this stability problem Reference [2] adjusts the RC law to aim for the inverse of the steady state frequency response of $G^{-1}(e^{-i\omega T})$ where T is the sample time interval, instead of the inverse of the transfer function. The cost function is the J_2

of Chapter 4

$$J_2 = \sum_{j=0}^{N} \left[G^{-1}\left(e^{i\omega_j T}\right) - F(e^{i\omega_j T})\right]\left[G^{-1}\left(e^{i\omega_j T}\right) - F(e^{i\omega_j T})\right]^* \qquad (5\text{-}1)$$

In order to create an implementable compensator $F(z)$, it is chosen in the form of an FIR filter in Equation (3-11) that sums a set of errors from the previous period each multiplied by a gain

$$F(z) = a_1 z^{m-1} + a_2 z^{m-2} + \cdots + a_m z^0 + \cdots + a_{n-1} z^{-(n-m-1)} + a_n z^{-(n-m)}$$
$$= (a_1 z^{n-1} + a_2 z^{n-2} + \cdots + a_m z^{n-m} + \cdots + a_{n-1} z^1 + a_n z^0)/z^{(n-m)} \quad (5\text{-}2)$$

chosen by the optimization. Since all poles are at the origin, this creates a stable RC compensator. As seen in the Chapter 4 this approach can work, but it has an implicit weighting on frequencies that might not be desirable. Consider the 3rd order minimum phase Laplace transfer function of Equation (2-13). The gain at DC is unity, but at 100 Hz it is 6×10^{-5}. The $F(z)$ tries to match the reciprocal of this at 100 Hz asking for a very large number. It is clear that for such systems, the compensator must put much more emphasis on matching the high frequency behavior than the low frequency behavior when minimizing the cost.

REMARK 3: Reference [2] converts the cost function J_2 to cost function J_1 in Equation (4-11) in Chapter 4 and observes better behavior in various problems

$$J_1 = \sum_{j=0}^{N} \left[1 - G(e^{i\omega_j T}) F(e^{i\omega_j T})\right]\left[1 - G(e^{i\omega_j T}) F(e^{i\omega_j T})\right]^* \quad (5\text{-}3)$$

Of course, both cost functions produce the same minimum of the inverse of the steady state frequency response if there is full freedom to make $F(e^{-i\omega_j T})$ do whatever is needed at each frequency. But one would like to have a relatively small number of gains in the FIR filter in order to limit the number of computations made each time step. And this means that the cost functions produce different results. Experience with this cost function and the 3rd order system with pole excess of 3, indicates that using only 3 terms in $F(z)$ is sufficient to produce stability of the RC design. This relationship between number of terms needed and pole excess is fairly general. When 12 gains are used the square bracket terms in Equation (5-3) are zero to a few decimal places, so the compensator can be very effective.

Observation 1: The optimization criterion J_1 can be rewritten as

$$J_1 = \sum_{j=0}^{N} \left[G^{-1}\left(e^{i\omega_j T}\right) - F(e^{i\omega_j T})\right]\left[G(e^{i\omega_j T})G(e^{i\omega_j T})^*\right]\left[G^{-1}\left(e^{i\omega_j T}\right) - F(e^{i\omega_j T})\right]^* \qquad (5\text{-}4)$$

By comparison to J_2, the weight function $G\left(e^{i\omega_j T}\right)G(e^{i\omega_j T})^*$ has been introduced. For systems with small output at high frequencies, this squares the small number, and makes the optimization put much less emphasis on making $F(e^{-i\omega_j T})$ look like $G^{-1}\left(e^{i\omega_j T}\right)$ at high frequencies.

Observation 2: Following Remark 1, but with the $G^{-1}(z)$ replaced by $F(z)$, produced Equation (2-7)

$$z^p E(z) = [1 - \phi F(z)G(z)]E(z) \qquad (5\text{-}5)$$

If one makes a quasi-static assumption of the convergence process, then substituting $z = e^{-i\omega T}$ into the square bracket creates a frequency transfer function from one period of p time steps to the next. Its magnitude for any ω is the factor by which the amplitude at that frequency is multiplied from one period to the next. If all such magnitudes are less than one, it suggests that all error frequency components decay with periods. Reference [3] shows that this predicted decay for each frequency is in fact quite accurate for very fast error decay, in spite of the quasi-static assumption. This is the result of the usual relatively large value of p in practical problems. The conclusion is that cost function Equation (3) can be interpreted as a quadratic penalty over all frequencies to Nyquist aiming to produce the fastest learning speed, or convergence rate.

Observation 3: Reference [3] proves that

$$\left|1 - \phi F\left(e^{i\omega T}\right)G\left(e^{i\omega T}\right)\right| < 1 \qquad \forall\omega \qquad (5\text{-}6)$$

is a necessary and sufficient condition of stability for all possible periods p of RC using compensator $F(z)$ (see Equation (2-8)). And only for very small and usually not practical values of p, can the RC be stable without satisfying this inequality.

Consider the implications of this condition. The design of F aims to cancel the magnitude change of G, and it aims to cancel the phase change made going through transfer function $G(z)$. If the phase is right, then the magnitude of the product can be anywhere between zero and two, so magnitude errors are not normally very important in achieving stability. On the other hand, phase errors are very important. Suppose the magnitude of the product of F and G is unity at a given frequency, then the maximum phase error allowed before inequality in Equation (5-6) is violated, is $\pm 45°$. If one chooses to converge more slowly at that frequency, then as the gain of the product approaches zero, and the absolute value approaches unity, the maximum phase error approaches $\pm 90°$. Thus, reducing the learning speed at any frequency improves the stability robustness to model error at that frequency. And since model inaccuracy increases when one goes to high frequency, we conclude we would like the design process to have slow learning at high frequency.

REMARK 4: For any frequency ω that violates the inequality in Equation (5-6), the absolute value indicates the increase in the amplitude of any error at that frequency from one period to the next according to Equation (5-5). In order to prevent error growth, we modify the learning law to use a low pass filter $H(z)$ to cut off the learning process at such frequencies. The filter must be zero phase so that it does not disturb the phase of the compensator, as in Equations (2-4) and (2-5). The learning law becomes $U(z) = z^{-p}H(z)[U(z) + F(z)E(z)]$, and the error propagation difference equation becomes

$$[(z^p - H(z)) + \phi H(z)F(z)G(z)]E(z) = (z^p - H(z))[Y_D(z) - V(z)] \quad (5\text{-}7)$$

Frequencies above the cutoff mean that $H(z)$ is not unity and the right hand side is no longer zero. This makes a nonhomogeneous equation and above the cutoff the right hand side is a forcing function producing nonzero error after convergence. Convergence is a property of the solution of the homogeneous equation, and its solution converges at all frequencies provided the cutoff satisfies $\left|H\left(e^{i\omega T}\right)[1 - \phi F\left(e^{i\omega T}\right)G\left(e^{i\omega T}\right)]\right| < 1$ at all frequencies up to Nyquist. One must expect to need to use this cutoff for long term stability robustness. Having a good phase model all the way to Nyquist frequency is often not possible. The cutoff frequency must be tuned in hardware application by observing what frequencies grow, i.e. violate Equation (5-6), because one does not usually know what is wrong with one's model. A somewhat extreme example, the 3rd order model in Equation (2-13) models the behavior of one link of a robot. It has a 1.4 Hz bandwidth to reduce excitation of the lowest frequency vibration mode produced by flexibility in the harmonic drives. Magnitude response at 100 Hz is 6×10^{-5}. It is clear that there is no way one can make a good model that far above the bandwidth of the system, and there will be a number of higher frequency vibration modes if one could identify them. However, if the phase is wrong RC will build the amplitude of the error and eventually bring the instability out of the noise level. Note that if approaching frequencies for which instability will occur, by using a slower learning rate one is able to have a higher frequency cutoff, and eliminate the tracking error up to a higher frequency.

REMARK 5: Consider the typical behavior as a function of frequency of minimum and non-minimum phase systems.

Minimum Phase: One expects the transfer function *G* to have its largest magnitude response at DC, except if there is some resonant peak, or there is a compensator design involved that introduces

a phase lead. And as frequency gets large the magnitude response decays, for the continuous time model decays by 20 dB times the pole excess for every decade. With the weight factor in optimality criterion Equation (5-4), the design of *F* will put emphasis on matching magnitude response at low frequencies, and allow *F* to have slow convergence at high frequencies. From REMARK 3, this is a desirable property for robustness to model error. Figure 5-1 compares the performance of J_1 versus the Min-Max on the minimum phase system in Equation 2-13. The Min-Max method learns more uniformly at all frequencies. The quadratic cost is preferred for robustness to model error at high frequencies and faster learning at low frequencies.

Non-Minimum Phase: A zero z_C in transfer function $G(s)$ fed by a zero order hold, maps to zero z_D in the discrete time transfer function $G(z)$ approximately according to $z_D = e^{-z_C T}$ (the Taylor series expansion matches to the power equal to the number of poles). If the zero is a real zero on the positive real axes in the *s* plane, one expects that it is not too far from the origin, and it will map to the positive real axis in the *z* plane to a point relatively close to +1. The $R(z)$ has *p* poles around the unit circle and the first one is at +1. This indicates an approximate pole zero cancellation near +1, making a small gain from input to output for constant inputs, and for low frequency inputs. This produced slow learning from one period to the next according to Equation (5-5) for low frequencies, unless a large gain is used in *F* localized to low frequencies and the gain decays quickly as the frequency gets beyond this region. Otherwise, it will precipitate instability at higher frequencies. If the number of gains used in *F* is limited, the weight factor in cost function Equation (5-4) will choose to sacrifice the learning speed near DC. One needs a very large number of gains in *F* to fight this trend. We conclude that the near pole-zero cancellation is a challenge for RC design of non-minimum phase systems.

REMARK 6: The maximum value of the left hand side of Equation (5-6) occurs at some frequency, and represents is the slowest decay per period over all frequencies up to Nyquist. The Min-Max optimization criterion minimizes the maximum value over all frequencies, and hence must address this slow learning near DC. Observation of the behavior of Min-Max solutions is that the design attempts to produce approximately the same learning rate at all frequencies. This behavior is undesirable if one is interested in having slow learning at high frequencies for stability robustness.

REMARK 7: One advantage of the RC designs using cost functions based on frequency response, is that it is possible to design RC based only on frequency response test data, without having to create a differential or difference equation model. One applies a rich and long input signal and processes the input output data to determine frequency response directly. However, the results get more and more inaccurate at frequency goes up, and does so in a very noisy way, making the phase model from one frequency to the next jump around. This kind of noisy behavior is a serious difficulty when one is minimizing the maximum value. One needs to stop the Min-Max optimization before the data starts to exhibit such behavior, or one must use some smoothing process. This difficulty is much less serious when using the quadratic optimization criterion of Equation (5-3). One purpose of this chapter is to study how these issues affect the various possible RC designs.

REMARK 8: From the above REMARKS, one concludes that the Min-Max optimization criterion is particularly good at handling non-minimum phase systems that have slow learning near DC (similar thinking applies if the non-minimum phase zeros are complex, and the slow learning

applies to some higher frequency). But the Min-Max will try to learn at similar rates at all frequencies, and this increased learning rate at high frequencies is undesirable for model error robustness. Figure 5-2 shows this behavior for repetitive control of the following 4th order non-minimum phase system

$$G(s) = \frac{165.95(4 - s)}{s^4 + 21.5s^3 + 170.28s^2 + 2\zeta\omega_o s + \omega_o^2} \tag{5-8}$$

This represents a rotary mechanical system of inertias, dampers, torsional springs, a timing belt, pulleys and gears, and comes from Reference [6]. The sample interval T is equal to 0.02, and n and m in the quadratic cost compensator are equal to 20. Examining the learning at DC, the quadratic cost gives 0.8907, and the Min-Max design is able to reduce this to 0.7974. One main purpose of this chapter is to create an RC design method that transitions from Min-Max optimization for low frequencies to quadratic cost of the form of Equation (5-3) at high frequencies. The aim is to obtain the advantages of both designs simultaneously, without the disadvantages.

REMARK 9: An alternative design process is also considered. We note that the difficulty under consideration is that the Min-Max approach is not desirable at high frequencies. Observing that we expect to need to cut off the learning process above some frequency anyway, perhaps one can simply use the Min-Max optimization applied only to frequencies from zero to some chosen frequency. This can improve the performance near DC. Above the chosen frequency the design is free to go wild, and we need to apply an effective zero-phase cutoff filter. Probably this approach is sacrificing performance at higher frequencies, but it represents a simple and perhaps effective design approach. We study the effectiveness of this approach with various numerical studies.

5.3 Formulation of Objective Function to Transition from Min-Max to Quadratic Cost as Frequency Increases

In this section, we develop an optimality criterion that can emphasize the Min-Max at low frequencies and quadratic cost J_1 at high frequencies. The gains in $F(z)$ are chosen to optimize the objective function

$$\min_x \quad \bar{M} + \sum_j \left[1 - F\left(e^{i\omega_j T}\right)G\left(e^{i\omega_j T}\right)\right]V_j\left[1 - F\left(e^{i\omega_j T}\right)G\left(e^{i\omega_j T}\right)\right]^* \quad \forall j \tag{5-9}$$

$$\text{Subject to} \quad \left[1 - F\left(e^{i\omega_j T}\right)G\left(e^{i\omega_j T}\right)\right]W_j\left[1 - F\left(e^{i\omega_j T}\right)G\left(e^{i\omega_j T}\right)\right]^* \leq \bar{M} \quad \forall j$$

where x contains all compensator gains $a_1, a_2, \ldots, a_n$, the asterisk indicates complex conjugate, W_j's are weighting factors for the Min-Max cost function chosen to be non-zero for DC and low frequencies, V_j's are weighting factors for the quadratic penalty chosen to be non-zero for high frequencies.

The cost function of this new method contains two objectives. The first objective is to try to minimize the weighted maximum error on selected frequencies by finding feasible x that satisfies the weighted constraints $\bar{M}$ for every frequency j. This part is simply the Min-Max design from the Chapter 4 but with limited frequency range by using W_j's. The other objective is the second term of the minimization that is the weighted sum of the quadratic penalty over high frequency range, aiming to increase robustness to high frequency.

The objective function can be reformulated in the QCQP form as

$$\min_x \bar{x}^T K \bar{x} + f^T \bar{x} \tag{5-10}$$

$$\text{Subject to } \frac{1}{2}\bar{x}^T \bar{A}_j \bar{x} + \bar{b}_j^T \bar{x} + d_j \leq \bar{M}$$

where

$$\bar{x} = W_j \begin{bmatrix} x \\ \bar{M} \end{bmatrix} \qquad \bar{A}_j = W_j \begin{bmatrix} A_j & 0 \\ 0 & 0 \end{bmatrix} \qquad \bar{b}_j = W_j \begin{bmatrix} b_j \\ -1 \end{bmatrix}$$

$$d_j = \frac{1}{2} W_j \qquad K = \begin{bmatrix} \sum_{j=0}^{N} V_j A_j & 0 \\ 0 & 0 \end{bmatrix} \qquad f = W_j \begin{bmatrix} \sum_{j=0}^{N} 2 V_j b_j \\ 1 \end{bmatrix}$$

which can be solved by using CVX programming package, References [4] and [5].

The MATLAB spline-based curve function that maps

$$V_j = v \cdot \begin{Bmatrix} 0, & \omega_j \le a_v \\ 2\left(\frac{\omega_j - a_v}{b - a_v}\right)^2, & a \le \omega_j \le \frac{a_v + b_v}{2} \\ 1 - 2\left(\frac{\omega_j - a_v}{b_v - a_v}\right)^2, & \left(\frac{a_v + b_v}{2}\right) \le \omega_j \le b_v \\ 1, & \omega_j \ge b_v \end{Bmatrix} \tag{5-11}$$

is used to define V_j's . Similarly, W_j's are mapped according to

$$W_j = \begin{Bmatrix} 1, & \omega_j \le a_w \\ 1 - 2\left(\frac{\omega_j - a_w}{b_w - a_w}\right)^2, & a_w \le \omega_j \le \frac{a_w + b_w}{2} \\ 2\left(\frac{\omega_j - a_w}{b_w - a_w}\right)^2, & \left(\frac{a_w + b_w}{2}\right) \le \omega_j \le b_w \\ 0, & \omega_j \ge b_w \end{Bmatrix} \tag{5-12}$$

For simplicity, in this work a_w and a_v are set to be equal to a, and b_w and b_v are set to be equal to b. The function allows the weighting functions V_j and W_j to smoothly increase and decrease, respectively. There are three parameters that one can adjust in the cost function to tune the repetitive control for desired performance, *a*, *b*, and *v*.

The a_w represents the frequency, as a fraction of Nyquist frequency, at which the weighting function for the Min-Max cost function W_j starts decreasing from unity. The b_w controls how fast

the weight W_j decreases. Then the difference $b_w - a_w$ represents the transition band, or the frequency interval to make the transition. On the other hand, a_v is where V_j starts increasing from zero, i.e. where we start applying the quadratic penalty to the objective function. The b_v is where the penalty V_j reaches its final value of v. By making a_w and a_v both equal to a, and b_w and b_v equal to b, one of the weights starts down at the same frequency that the other weight start up, and they both finish their transition at the same frequency. The value of v is the quadratic penalty coefficient indicating the relative importance of the Min-Max cost in its frequency range versus the quadratic cost minimization in its frequency range. At high frequencies, slower decay from one period to another is desirable, the magnitude of the frequency transfer function from period to period being close to +1 is desirable. Hence, we need to adjust the value of v for this purpose, but ensure that the magnitude remains below unity all the way to Nyquist.

5.4 Evaluating the Performance of the Merged Cost Function – Adjusting the Cost Function Parameters

Figure 5-3 compares the learning rate for Min-Max, for quadratic cost, and for Min-Max merged with quadratic cost when applied to the following 4th order non-minimum phase system of Equation (5-8). The figure demonstrates that the Min-Max merged with quadratic penalty accomplishes the modifications of the learning rate as a function of frequency shifting in the desired direction. There is in fact, faster learning rates at nearly all frequencies except for frequencies near Nyquist. The DC learning rate for Min-Max is 0.7974 while the merged cost function reduces this to 0.6134. Figures 5-4 and 5-5 show weighting factors W_j and V_j used for Figure 5-3, respectively, that are chosen to make a fast transition from the Min-Max cost function to the quadratic cost function. The W_j's are equal to unity until 50 percent of Nyquist and the transition band is 10 percent of Nyquist, and then W_j's are equal to zero from 60 percent of Nyquist

up to Nyquist. The v is equal to 0.007, and a and b are equal to 50 and 60 percent of Nyquist, respectively.

The design parameters to use in tuning the cost function are the frequencies for the start and the end of the transition zones a, b, as well as the transition frequency interval, or transition band, $a - b$, and the relative importance of the quadratic cost minimization versus the Min-Max, v. Figure 5-6 studies the behavior when the transition band is held constant at 10%, but the frequency at which the band starts, a, is varied 30%, 50%, and 70%, and compared to Min-Max alone. The DC values are 0.6134, 0.6481, 0.7002, and 0.7974, respectively. The number of gains $n = m = 20$, and v is equal to 0.005. We conclude that reducing the frequency when the Min-Max cost weight starts its descent to zero, improves the learning rate at DC and low frequencies, but the quadratic penalty takes over at a lower frequency resulting in the magnitude of $1 - F(e^{i\omega_j T})G(e^{i\omega_j T})$ starting to grow at lower frequencies.

Figure 5-7 fixes the value of b at 50% Nyquist, and adjusts the value of a. In other words, the figure examines the effect of varying the transition band width $a - b$ values of 50%, 40%, 30%, 20%, 10%, and 0%. Figures 5-8 and 5-9 give the corresponding W_j and V_j plots to Figure 5-7. The values at DC are 0.5672, 0.5797, 0.5963, 0.6129, 0.6275 which are all very close to the same values. When the transition band is longer, the weighting functions W_j start decreasing earlier. As a result, the magnitude of $1 - F(e^{i\omega_j T})G(e^{i\omega_j T})$ starts growing at lower frequencies, but the lower and higher frequencies are remarkably similar for each transition band width. It appears that the transition band width is not very effective at modifying learning behavior.

Figure 5-10 shows learning rates with different values of v. Figure 5-11 is the corresponding plot of v. The choice of the value of v adjusts how important it is to reduce the value of the quadratic penalty function at high frequencies. The lower the value of v is, the faster

the learning at low frequencies and the slower learning rate at high frequencies. The lowest value of v that keeps the magnitude of $1 - F(e^{i\omega_j T})G(e^{i\omega_j T})$ below unity for all frequencies up to Nyquist is 0.005, the lowest value for which the repetitive control system is asymptotically stable. Aiming to increase robustness at high frequencies, one needs a slower learning rate close to unity. One allows values approaching unity to improve robustness to expected model error at high frequencies. For v equal to 0.004, 0.005, 0.006, 0.007. 0.008, the DC values are 0.5989, 0.6481, 0.6992, 0.7562, 0.8245 respectively.

Figure 5-12 repeats Figure 5-3 using a faster sample rate of 100 Hz instead of 50 Hz, and also examines the influence of changing the number of gains in the compensator $F(z)$. The faster sample rate makes it more difficult to learn fast at DC. With the same number of gains as before in Figure 5-3, Min-Max gave 0.7974 and with double the sample rate it is 0.9856. The corresponding merged cost function comparison goes from 0.6134 to 0.8132. Increasing the number of gains in F(z) improves learning at DC using the faster sample rate, the Min-Max value reduces to 0.8959 and the merged cost function value reduces to 0.7509.

We conclude that the merged cost function can improve the learning rate substantially compared to Min-Max and at the same time produce the slower learning rate desired for robustness at high frequency. The numerical results suggest using a moderate value for a and b, learning slower at high frequency for robustness but adjusted to preventing instability at high frequency, and then this facilitates faster learning in the low frequency range. There appears to be no substantial benefit to have a large transition band a - b, so this value can be a moderate value as well. If one wants faster learning at low frequency, one can lower the frequency a at which the transition starts. And using a smaller value of v also can increase the learning rate at low frequencies, at the expense of slower learning at high frequencies.

5.5 Alternative Approach: Use Min-Max up to Chosen Frequency and Apply Zero-Phase Low-Pass Filter Cutoff

We can consider a simpler alternative to the merged cost function approach. The Min-Max objective function normally would be applied to all frequencies to Nyquist. But perhaps one is really interested in improved learning rate for non-minimum phase systems at low frequencies, and is not concerned about learning at high frequency. Because of model error one might need to cut off the learning at high frequency and forgo the attempt to get enough robustness for convergence using a slow learning rate. This attempt is limited to producing stability for model phase error between plus and minus 90 degrees. When the model error exceeds this value one will still need to use a cutoff filter when applying the merged design, and in this case the benefit of the merged cost design is that the cutoff can be made at a higher frequency. References [7] and [8] present designs for zero phase low pass filters to use to cut off the learning above a chosen cutoff frequency.

To study this alternative, we can start with the Min-Max cost function of Equation (4-12) and set the weights W_j to zero above some value of j corresponding to the chosen frequency. There is an option of making use of Equation (5-11) to phase out the importance of frequencies approaching the zero cutoff, again having a transition band $b - a$.

We investigate how the cutoff and transition band of weighting function W_j's influence the learning rate when the low-pass filter is not applied. Figures 5-13 shows the learning rate using the Min-Max compensator with cutoff of the learning starting at 30%, 50% and 70% Nyquist, and using a 10% transition band, reaching weight zero at 40%, 60%, and 80%. Figure 5-14 presents a detailed view of this figure. Lowering the cutoff frequency produces faster learning rate at low

frequencies compared to using Min-Max to Nyquist. But note that the resulting learning rate at low frequencies seems to be nearly the same for all 3 cutoff values. Also, note that the magnitude of $1 - F(e^{i\omega_j T})G(e^{i\omega_j T})$ sharply increases above unity right after the transition band, i.e. when W_j reache zero, at 40%, 60% and 80% Nyquist. Figures 5-15 and 5-16 present figures corresponding to 5-13 and 5-14, examining the influence of the size of the transition band. The value of b at which the weight W_j reaches zero is 70% for all plots in the figure, but the decay of the weight starts at a, which is 0, 30%, and 70% corresponding to no transition zone. In this setting, 0% transition band performs best since the magnitude of $1 - F(e^{i\omega_j T})G(e^{i\omega_j T})$ is the lowest at almost all frequencies. Note that the learning rate is very substantially improved by limiting the Min-Max to 70% of Nyquist, suggesting that this is an effective design method if one's primary interest is learning at low frequencies, but its usefulness is predicated on one's ability to have an effective cutoff filter to cancel the sharp growth above 70%.

The ideal cutoff filter is equal to one in the passband and equal to zero in the stopband. For this application, the design of cutoff filter needs to be very effective in the stopband. The FIR zero-phase low-pass filter design developed in Reference [8] allows one to put emphasis on the stopband by adjusting the value of γ in Equation (2-5), but that is done at the expense of accuracy in the passband, and the accuracy of the repetitive control can only be as good as the filtered value of the error signal. Inaccuracy in the passband can be amplified when going from the forcing function in the difference Equation (2-6) to the resulting particular solution to the difference equation. The error response to all frequencies is obtained by getting the transfer function from disturbance to error, i.e. the sensitivity transfer function $S(z)$ in Equation (5-13), going from desired output $Y^*(z)$ and disturbance $V(z)$ to the corresponding error $E(z)$

$$S(z) = \frac{E(z)}{Y^*(z) - V(z)} = \frac{z^p - H(z)}{z^p - H(z)[1 - \phi F(z)G(z)]} \tag{5-13}$$

Aiming to accurately fix error in the passband, one should not put the weight of γ to be too high. The cutoff frequency of the FIR low-pass filter should be equal or less than the learning cutoff where the errors start to grow rapidly.

Figure 5-17 shows the resulting magnitude of $H(z)[1 - F(z)G(z)]$ using different cutoff frequencies for weighting function W_j (Figure 5-18) and for FIR low-pass filter $H(z)$ (Figure 5-19). The weighting functions W_j's use $a = b$, with cutoff at 30%, 50% and 70% of Nyquist. The FIR low-pass filters $H(z)$ are designed with the passband starting 10% below the above cutoffs, and ending at the above cutoffs. The cutoff design uses 51 gains, and equal weighting of passband vs. stopband. It is clear that using a low frequency of the cutoff in the Min-Max cost function, makes the cutoff filter unable to produce small values above the cutoff, and the values at the higher frequencies produce a forcing function in the difference equation for the error that could result in large total error. On the other hand, one should not have the cutoff in the forcing function too high, for example, to avoid the noise in the frequency response model discussed in the next section. Again, we comment that the learning rate at the low frequencies seems to be rather independent of the percent Nyquist in the cutoff for 30%, 50%, and 70%.

5.6 Evaluating Performance of Both Approaches when Designing from Noisy Frequency Response Data

A very desirable property of the repetitive control design methods presented here, is that they do not require a mathematical model of the system, but instead can be designed based only on frequency response information that can be obtained experimentally. This section considers

designing from such data and addresses how one should adjust the parameters to address the influence of noise in the frequency response data.

Consider computing the system frequency response from experiments using rich input data. A rich input containing all frequencies is applied to the system, the resulting output is recorded, and the two are used to compute frequency response. We use Welch's average periodogram method, the TFE algorithm in MATLAB, Reference [9].

When the data is noisy, the quality of the identification of magnitude and phase deteriorates as the frequency goes up and the signal to noise ratio gets small. We examine up to what frequency one can trust the computed frequency response and study implications for adjusting frequency range used for weighting function and penalty coefficients.

The system in Equation (5-8) fed by a zero order hold is used as a model with 100 Hz sample rate. The input is generated using normally distributed pseudo-random numbers with unity standard deviation, generated for various data lengths. A pseudo random noise is also generated normally distributed, unity standard deviation, is chosen and then scaled so that when added to the output it produces a signal to noise ratio of 100 producing noisy measurement data.

Figure 5-20 and 5-21 show the estimated phase and true phase response of the system when 1,000,000 and 10,000,000 time steps of data are used, respectively, in the TFE algorithm with default window size, and default overlap. Extra curves are shown for adding and subtracting 360 degrees from the phase of the system. Phase error of ± 90 degree or more in the phase cancellation in $F(z)G(z)$ causes instability. Error in the gain is less critical. As the frequency increases, identified phase plots start to contain substantial influence from noise and no longer reflects the true behavior.

Figure 5-22 investigates how well the Min-Max merged with quadratic penalty works when frequencies response data is noisy using data in Figure 5-21. It is clear that the one does not want to use Min-Max all the way to Nyquist. The DC value is 0.9856. The solid curve for the merged cost uses $a = 40$ and $b = 100$ with $v = 0.006$. This is adjusted to get stability all the way to Nyquist. The third curve uses the merged cost function with $a = 50$, $b = 60$, $v = 0.008$, and introduces a zero-phase low-pass filter cutoff with passband to 50% and transition band of 10%. Having the cutoff means that one is no longer constrained to try to produce stability to Nyquist as in the solid curve, an better performance is obtained below the filter cutoff. The DC values for the merged cost is reduced to 0.8480, and when the cutoff filter is added the DC value is 0.3293, which is dramatically reduced.

Figure 5-23 repeats Figure 5-14 but using the noisy frequency response data. This time the 70% curve is much worse than before, and the learning rates for the other curves are not as similar as before. Figure 5-24 examines the behavior when one introduces the zero-phase low-pass filter to stabilize the curves in Figure 5-23, presenting the magnitude frequency response of $H(z)[1 - F(z)G(z)]$. The values of W_j correspond to cutoff starting at 20%, 40%, and 60% with 10% transition to zero. The FIR low-pass filters are given in Figure 5-25. Due to phase error at high frequencies near Nyquist, the resulting magnitude of $1 - F(z)G(z)$ using the Min-Max approach all the way to Nyquist results in almost no learning approaching Nyquist as well as at DC. The results show that limiting learning cutoff in W_j can improve learning rates at low frequencies. As the learning cutoff decreases, the learning rate becomes faster at low frequencies at the expense of no learning above the cutoff. This learning cutoff should not exceed the frequency range where the phase identification is accurate enough to avoid instability from model error at high frequencies.

Figures 5-26 and 5-27 correspond to Figures 5-16 and 5-15, respectively, which study the influence of transition band of the weighting function for the Min-Max optimization when all weighting functions become zero at 70% of Nyquist frequency. The cutoff of the FIR zero-phase low-pass filter is designed to be lower than 70% Nyquist. The plots show that longer transition band can improve learning rates at DC and low frequencies but might need a lower cutoff frequency for the low-pass filter. Otherwise, the magnitude of $1 - F(z)G(z)$ can grow above unity at low frequency and become unstable. The transition band of 70% Nyquist, which means the weight W_j that starts decreasing from unity at DC and completely equal to zero at 35 Hz, needs a cutoff frequency for low-pass filter before 25 Hz while the transition band of 40% Nyquist needs a cutoff frequency at 33Hz. The one with transition band 0% grows above unity at 39 Hz which is higher than the frequency range that we try to optimize. Figure 5-28 shows the magnitude plot of $H(z)[1\text{-}F(z)G(z)]$ when the corresponding FIR low-pass filters in Figure 5-29 are applied.

In the real world when data is noisy and phase frequency response at high frequencies becomes inaccurate due to noise, excessively long data sets can be required for accurate phase response identification at high frequencies. Min-Max design all the way to Nyquist is very sensitive to the noise at high frequencies. One might want to use some smoothing technique trying to smooth through the noise. The merged cost function allows one to decrease this influence very substantially. But it still may not be able to guarantee convergence to Nyquist because of the phase error in the computed frequency response. One may introduce a zero-phase low-pass filter to cut off the learning when the frequency response model becomes too highly influenced by noise. One can then consider the alternative design which does not try to learn to high frequencies, resulting in much better learning at low frequencies, but this time requiring an effective zero-phase low-pass filter to produce stability. In practice, one might might tune this cutoff in hardware by picking

a desired trajectory that contains all frequencies and run for a number of periods to see which frequencies are growing from one period to the next period. This process is possible since the instability in RC grows slowly. But we comment that when using the alternative design, the slow instability may not apply. One might need to start with a cost function aiming for learning to Nyquist and reduce the frequency range gradually.

5.7 Conclusions

The combined optimization criterion presented here introduces significant design choices in generating repetitive controllers for non-minimum systems. When using the pure Min-Max criterion for all frequencies up to Nyquist, the speed of learning at DC is very much improved over the quadratic cost approach, but is still limited. And the performance at high frequencies can be asking for higher learning rates than one might want based on robustness to model error. When transitioning to a quadratic cost as the frequencies increase, the performance of Min-Max at low frequencies can be improved. If the quadratic cost is further de-emphasized relative to the Min-Max using the weights, then one obtains further improvement in Min-Max low frequency performance, and can still maintain stability for the given model up to high frequencies. One can choose to go still further, and simply ask for Min-Max up to some chosen frequency and use a cutoff filter to ignore everything above this frequency. This can produce particularly good performance near DC for non-minimum phase systems, and is easy to design. But it sacrifices all attempt to learn at high frequencies. Thus, the methods presented represent a spectrum of choices for the designer. We conclude that in practice, instead of using the pure Min-Max approach to address non-minimum phase systems, one should prefer to use one of the two methods developed here when designing repetitive controllers.

5.7 References

[1] P. Prasitmeeboon and R. W. Longman, "Using Quadratically Constrainted Quadratic Programming to Design Repetitive Controllers: Application to Non-Minimum Phase Systems," *Proceeding of the AIAA/AAS Astrodynamics Specialist Conference,* Vail, CO, August 2015.

[2] B. Panomruttanarug and R. W. Longman, "Repetitive Controller Design Using Optimization in the Frequency Domain," *Proceedings of the 2004 AIAA/AAS Astrodynamics Specialist Conference*, Providence, RI, August 2004.

[3] R. W. Longman, "On the Theory and Design of Linear Repetitive Control Systems," *European Journal of Control*, Special Section on Iterative Learning Control, Guest Editor Hyo-Sung Ahn, Vol. 16, No. 5, 2010, pp. 447-496.

[4] M. Grant and S. Boyd, "Graph Implementations for Nonsmooth Convex Programs," *Recent Advances in Learning and Control* (a tribute to M. Vidyasagar), V. Blondel, S. Boyd, and H. Kimura, editors, pages 95-110, *Lecture Notes in Control and Information Sciences*, Springer, 2008. http://stanford.edu/~boyd/graph_dcp.html.

[5] S. Boyd and L. Vandenberghe. *Convex Optimization*, Cambridge University Press, Cambridge, 2004.

[6] L. Wang, C. T. Freeman, E. Rogers, and D. H. Owens, "Experimentally Validated Continuous-Time Repetitive Control of Non-Minimum Phase Plants with a Prescribed Degree of Stability," *Control Engineering Practice,* Vol. 18, 2010, pp.1158-1182.

[7] B. Panomruttanarug and R. W. Longman, "Frequency Based Optimal Design of FIR Zero-Phase Filters and Compensators for Robust Repetitive Control," *Advances in the Astronautical Science*, Vol. 123, 2006, pp" 219-238.

[8] J. Bao and R. W. Longman, "Enhancement of Repetitive Control using Specialized FIR Zero-Phase Filter Designs," *Advances in the Astronautical Sciences*, Vol. 129, 2008, pp. 1413-1432.

[9] P.D. Welch, "The use of FFT for the Estimation of Power Spectra: A Method based on Time Averageing over Short, Modified Periodograms," IEEE Transactions on Audio Electroacoustics, Vol. AU-15, June 1967, pp. 70-73.

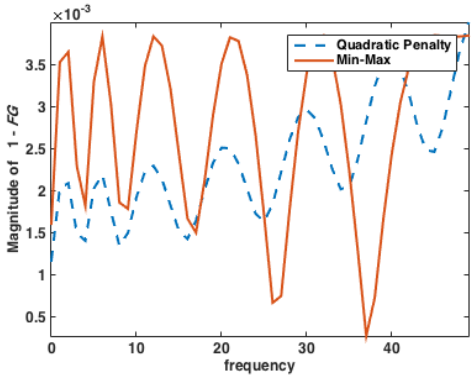

Figure 5-1. $|1 - G(z)F(z)|$ of the 3rd order minimum phase system by using the quadratic penalty and the Min-Max approaches.

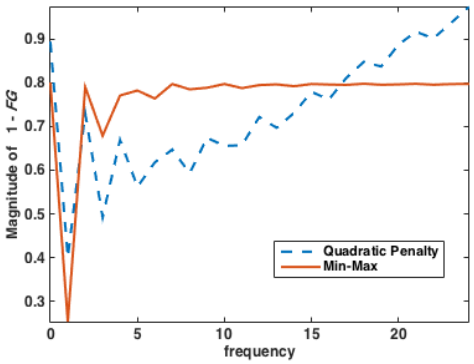

Figure 5-2. $|1 - G(z)F(z)|$ of the 4th order non-minimum phase system by using the quadratic penalty approach and the Min-Max approach.

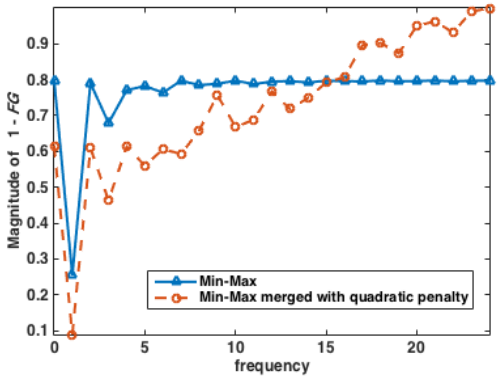

Figure 5-3. $|1 - G(z)F(z)|$ of the 4th order non-minimum phase system by using the Min-Max approach and the Min-Max merged with quadratic penalty.

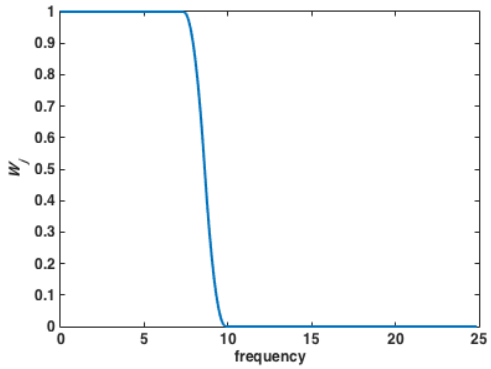

Figure 5-4. Weighting factors for Min-Max cost function W_j corresponding to Fig.5-3.

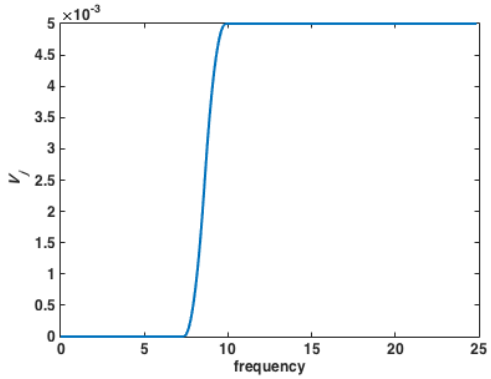

Figure 5-5. Weighting factors for quadratic penalty V_j corresponding to Fig.5-3.

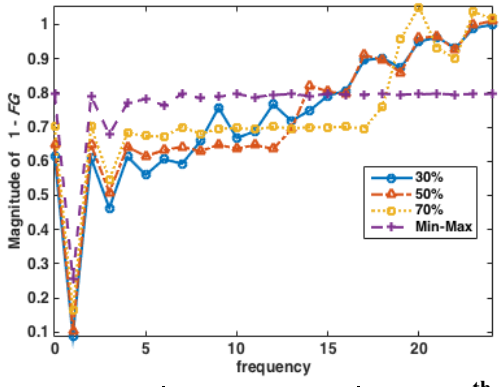

Figure 5-6. $|1 - G(z)F(z)|$ of the 4th order non-minimum phase system by using the Min-Max merged with quadratic penalty with different learning cutoff percentage of Nyquist.

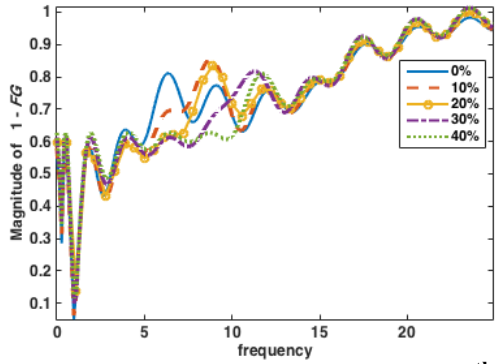

Figure 5-7. $|1 - G(z)F(z)|$ of the 4th order non-minimum phase system by using the Min-Max merged with quadratic penalty with different transition band for weighting function W_j.

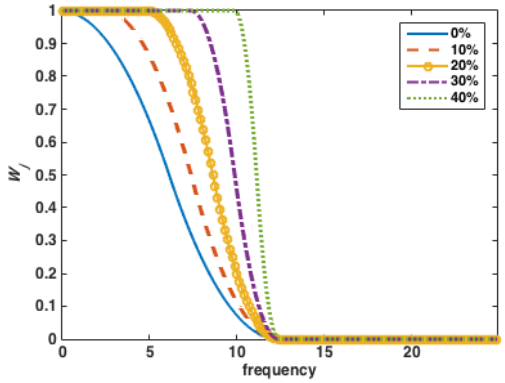

Figure 5-8. Weighting factors for Min-Max cost function W_j corresponding to Fig.5-6.

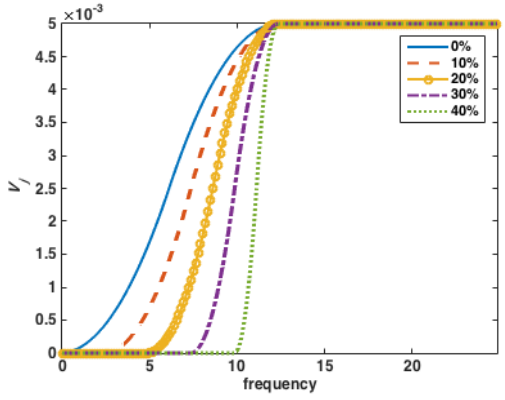

Figure 5-9. Weighting factors for quadratic penalty V_j corresponding to Fig.5-6.

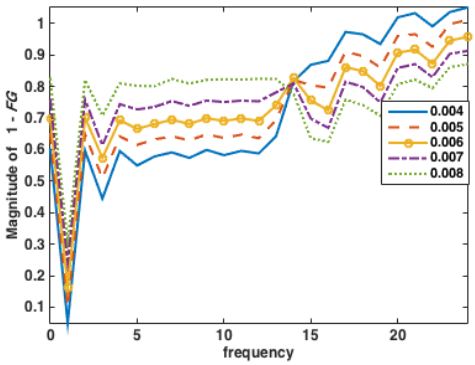

Figure 5-10. $|1 - G(z)F(z)|$ of the 4th order non-minimum phase system by using the Min-Max merged with quadratic penalty with different weighting function for quadratic penalty V_j.

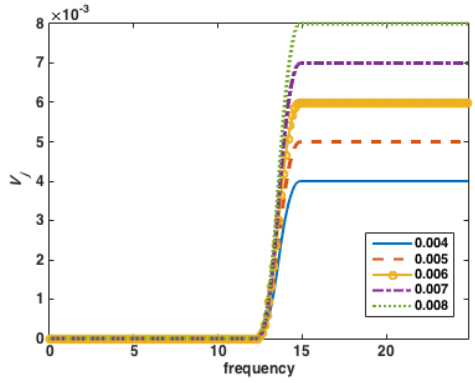

Figure 5-11. Weighting $V_j$on quadratic penalty corresponding to Fig. 5-6.

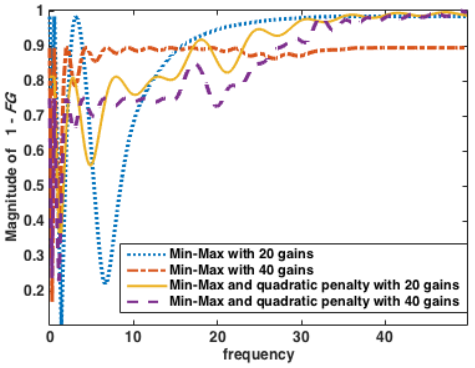

Figure 5-12. Compared learning rate between Min-Max method and the Min-Max merged with quadratic penalty with different sample rate and number of gains.

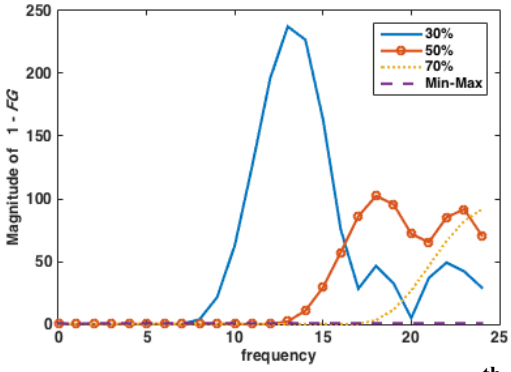

Figure 5-13. $|1 - G(z)F(z)|$ of the 4th order non-minimum phase system by using the Min-Max with limited range of non-zero for W_j without low-pass cutoff filter.

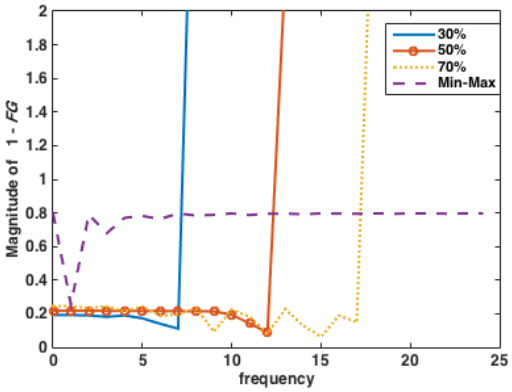

Figure 5-14. Detail of Fig. 5-13.

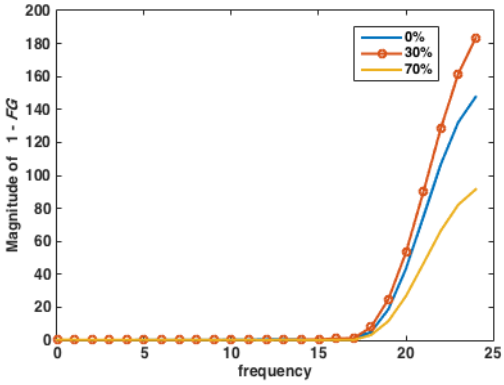

Figure 5-15. $|1 - G(z)F(z)|$ when W_j is completely zero at 70% Nyquist with different frequency range of transition band.

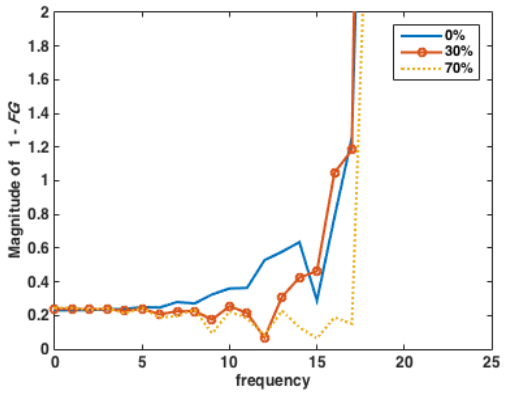

Figure 5-16. Detail of Fig. 5-15.

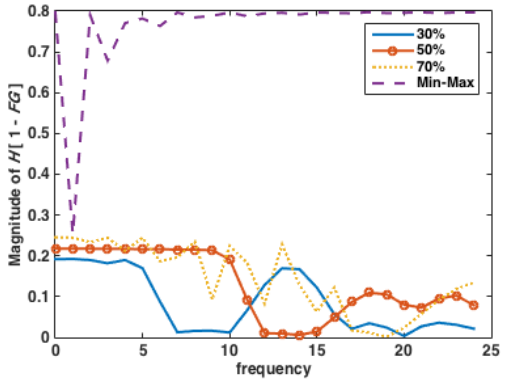

Figure 5-17. $H(z)|1 - G(z)F(z)|$ of the 4th order non-minimum phase system by using the Min-Max with limited range of non-zero for W_j.

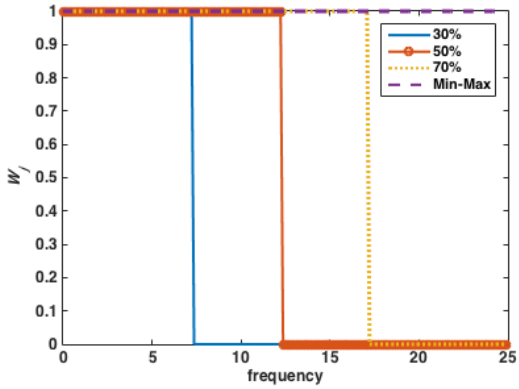

Figure 5-18. W_j corresponding to Fig. 5-17.

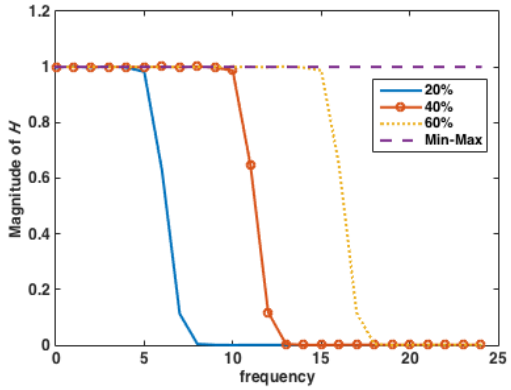

Figure 5-19. $|H(z)|$ corresponding to Fig. 5-17.

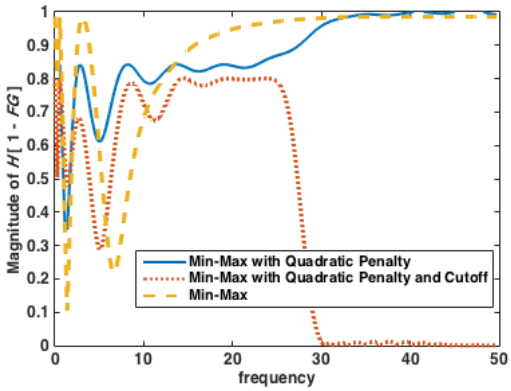

Figure 5-22. Comparing learning rate using $|H(z)[1 - G(z)F(z)]|$

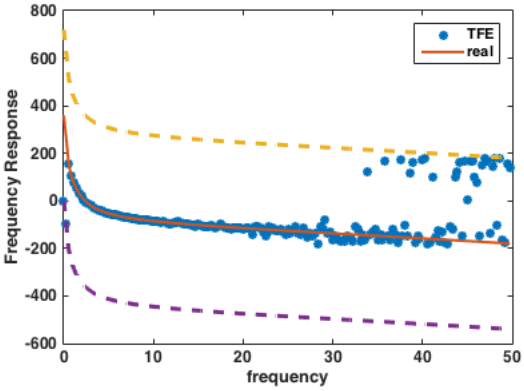

Figure 5-20. Phase plot, SNR of 100, 1,000,000 points, sample rate of 100 Hz.

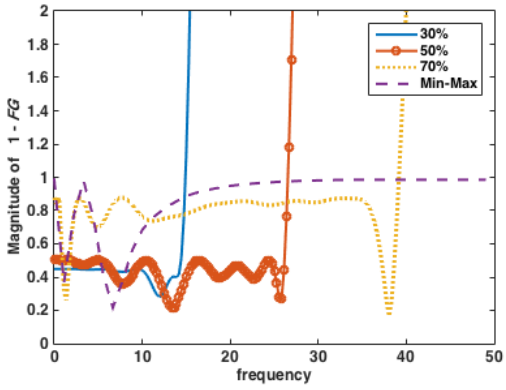

Figure 5-23. $|1 - G(z)F(z)|$ with different learning cutoff.

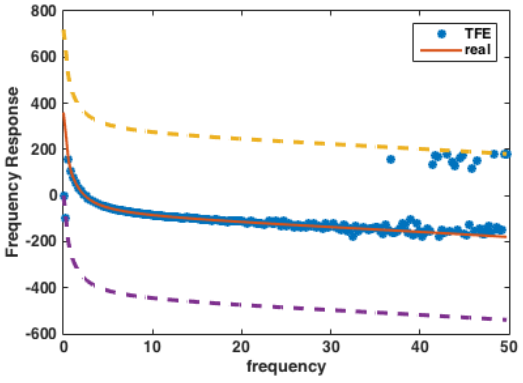

Figure 5-21. Phase plot, SNR of 100, 10,000,000 points, sample rate of 100 Hz.

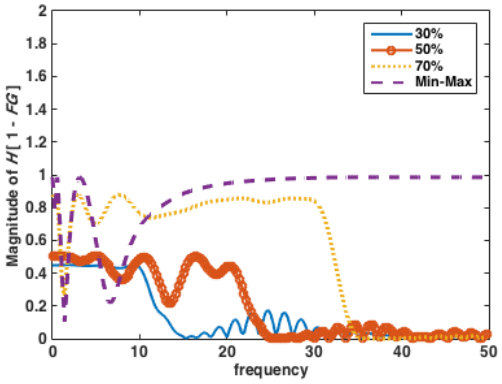

Figure 5-24. $|H(z)[1 - G(z)F(z)]|$ when W_j is completely zero at 70% Nyquist with different frequency range of transition band.

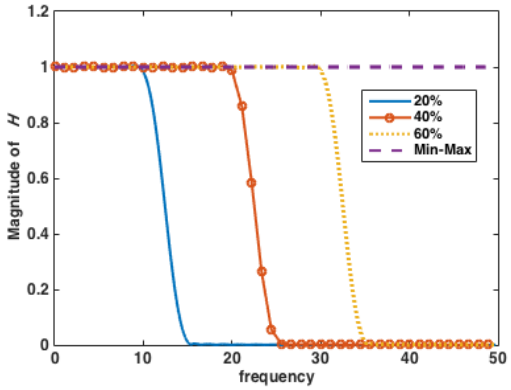

Figure 5-25. Magnitude frequency response for FIR low-pass filters corresponding to Fig. 5-26.

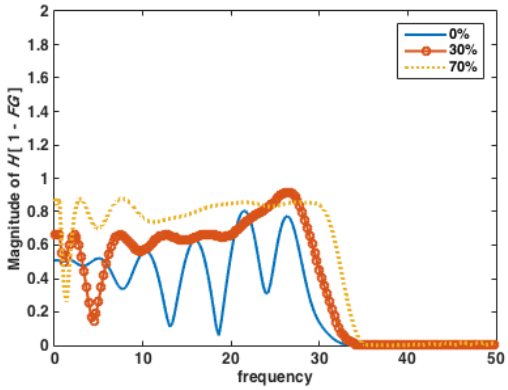

Figure 5-28. $H(z)|1 - G(z)F(z)|$ when applying low-pass FIR filter.

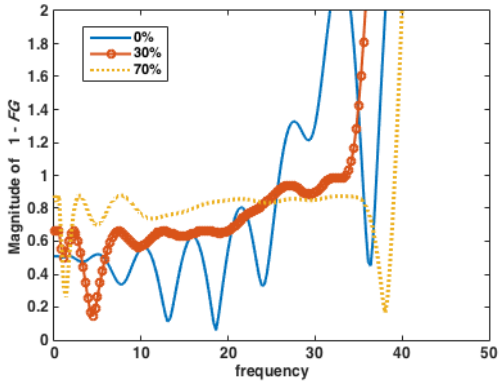

Figure 5-26. $|1 - G(z)F(z)|$ when W_j is completely zero at 70% Nyquist with different frequency range of transition band.

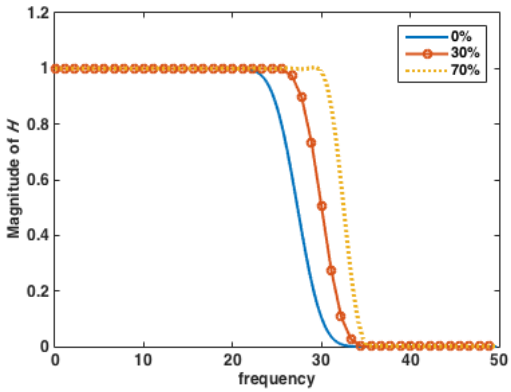

Figure 5-29. Corresponding frequency magnitude respond of FIR low-pass filter to Fig. 5-28.

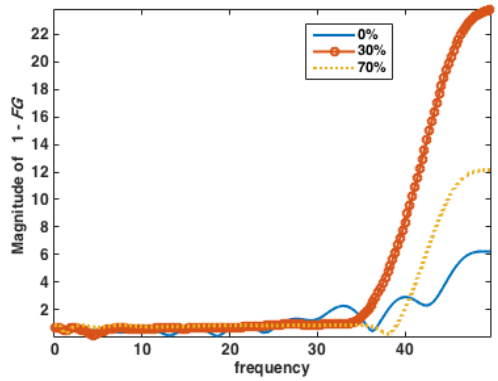

Figure 5-27. Same as Fig. 5-26.

Chapter 6

Conclusion

Several methods have been developed in this dissertation to create repetitive control laws with improved performance and increased robustness for minimum phase systems and non-minimum phase systems. In repetitive control, the command to be executed and the disturbance are periodic functions of time. Typical feedback control has limited performances because it does not make use of knowledge of the periodic nature of the command and disturbance. Repetitive controllers use error from the previous period to adjust the command in the present period. In theory, repetitive control can completely eliminate the effects of a periodic disturbance. This dissertation is separated into four major topics. Chapter 2 and 3 present solutions to general RC systems. Chapter 4 and 5 present optimization methods that specifically work well for non-minimum phase systems but might not be preferred for minimum phase systems.

The first topic in this dissertation studies to what extend the emulation methods can be used to implement RC compensators. The pole-zero mapping method is modified to create an effective RC approach using only the continuous time models. The new method uses the exact images of the pole locations, approximate images of any zero locations, and supplies poles at the origin to address the phase influence of the asymptotic zeros introduce by the discretization process. This method is simple and provides insight into stability robustness of RC compensator designs. The results of the methods will almost always be stable.

The ideal performance for a repetitive control system is to have zero tracking error. Because of model error at high frequencies from parasitic poles or residual modes, one normally requires a zero phase filter to cutoff the process above some frequency. This stabilizes the RC system at the expense of no longer addressing the error above the cutoff. The instability just above the needed cutoff frequency is normally slow, and this allows one to use real time data to correct the model error for such frequencies. This allows one to raise the cutoff, and perhaps eliminate the need for a cutoff entirely. Chapter 3 develops real time model update methods that adaptively update the frequency response model for the chosen frequencies allowing one to successively raise the cutoff frequency in real time. The methods considered to identify frequency response are the moving window approach and the projection algorithm. These methods allow one to compute and use the magnitude and phase response adjustments needed to the input command for each frequency independently while the systems are running.

The later part of this dissertation focuses on designing RC compensators for non-minimum phase systems, having a zero or zeros in the right half of the complex plane for the continuous time system. Such systems exhibit unusual behavior in response to unit step inputs, with the output initially going in the opposite direction from the step before moving in the commanded direction. The zero is usually mapped to discrete time on positive real axis close to +1 creating behavior similar to pole-zero cancellation at DC and low frequencies. The behavior causes slow learning at DC and low frequencies for existing RC design methods. Chapter 4 investigates the performance of existing approaches on non-minimum phase systems and proposes the optimization criterion to minimize the maximum of the error over frequencies in order to push the maximum value down. The Min-Max design approach improves the originally poor performance at low frequencies at the expense of slower learning at higher frequencies. A Taylor series expansion design method can

sometimes compete, but it needs an extra IIR part to exactly cancel poles and zeros inside the unit circle. The Min-Max can be designed directly from frequency response data. The Min-Max method is not to be preferred for minimum phase systems, but it is effective in most non-minimum phase systems because it directly addresses the basic pole-zero cancellation difficulty. To design RC controllers for this cost function, the problem statement is reformulated as a QCQP problem and solved by using CVX package.

The use of the Min-Max cost design method for non-minimum phase systems is furthered studied in Chapter 5. Although the method can effectively push down the maximum error, the method requires model accuracy up to high frequencies. In real world applications, signals at high frequencies can be decaying into the noise level due to their low magnitude frequency response. When one designs using frequency response obtained directly from long data runs with a rich input, the influence of noise on the model at high frequency, creates unwanted sensitivity to noise at high frequencies with the Min-Max design. Normally, the quadratic cost criterion results in slow learning at high frequencies, and this improves the robustness to model error at high frequencies. This chapter develops two methods to address the poor behavior of Min-Max at high frequencies. The first uses the Min-Max objective at low frequencies and transitions to the quadratic cost at high frequencies. This combination has the advantage of Min-Max that it addresses poor learning at low frequencies, and the advantage of the quadratic penalty function for improved robustness at high frequencies. A zero-phase low-pass filter to cutoff the learning at high frequencies can still be needed due to high frequency model error or actuator limitation. The other approach considered in Chapter 5 is to use the Min-Max objective up to some frequencies and apply a learning cutoff to frequencies above this frequency. Such an approach will not be able to address as large a frequency range, but it is seen to be simple and effective. The methods studied in Chapters 4 and

5 aimed specifically for good performance for non-minimum phase systems, are usually not to be preferred when designing RC for minimum phase systems.