Eliminating the Internal Instability in Iterative Learning Control for Non-minimum Phase Systems

Te Li

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

Eliminating the Internal Instability in Iterative Learning Control for Non-minimum Phase Systems

Te Li

Iterative Learning Control (ILC) iterates with a real world control system repeatedly performing the same task. It adjusts the control action based on error history from the previous iteration, aiming to converge to zero tracking error. ILC has been widely used in various applications due to its high precision in trajectory tracking, e.g. semiconductor manufacturing sensors that repeatedly perform scanning maneuvers.

Designing effective feedback controllers for non-minimum phase (NMP) systems can be challenging. Applying Iterative Learning Control (ILC) to NMP systems is particularly problematic. Asking for zero error at sample times usually involves inverting the control system. However, the inverse process is unstable when the system has NMP zeros. The control action will grow exponentially every time step, and the error between time steps also grows exponentially. If there are NMP zeros on the negative real axis, the control action will alternate its sign every time step.

ILC must be digital to use previous run data to improve the tracking error in the current run. There are two kinds of NMP digital systems, ones having intrinsic NMP zeros as images of continuous time NMP zeros, and NMP sampling zeros introduced by discretization. Two ILC design methods have been investigated in this thesis to handle NMP sampling zeros, producing zero tracking error at addressed sample times: (1) One can simply start asking for zero error after

a few initial time steps, like using multiple zero order holds for the first addressed time step only (2) Or increase the sample rate, ask for zero error at the original rate, making two or more zero order holds per addressed time step.

The internal instability can be manifested by the singular value decomposition of the input-output matrix. Non-minimum phase systems have particularly small singular values which are related to the NMP zeros. The aim is to eliminate these anomalous singular values. However, when applying the second approach, there are cases that the original anomalous singular values are gone, but some new anomalous singular values appear in the system matrix that cause difficulties to the inverse problem. Not asking for zero error for a small number of initial addressed time steps is shown to eliminate all anomalous singular values. This suggests that a more accurate statement of the second approach is: using multiple zero order holds per addressed time step, and eliminating a few initial addressed time steps if there are new anomalous singular values.

We also extend the use of these methods to systems having intrinsic NMP zeros. By modifying ILC laws to perform pole-zero cancellation inside the unit circle, we observe that all of the rules for sampling zeros are effective for intrinsic zeros. Hence, one can now achieve convergence to zero tracking error at addressed time steps in ILC of NMP systems with a well behaved control action.

In addition, this thesis studies the robustness of the two approaches along with several other candidate approaches with respect to model parameter uncertainty. Three classes of ILC laws are used. Both approaches show great robustness. Quadratic cost ILC is seen to have substantially better robustness to parameter uncertainty than the other laws.

Table of Contents

Acknowledgements

I want to take this opportunity to reflect on all those who have helped me during my PhD study. It is their help and support that encourage me to face all the challenges and go through all the difficulties.

Firstly, I would like to express my sincere gratitude to my advisor Prof. Richard Longman for his continuous support and trust, for his patience and immense knowledge. He is one of the smartest and most easy-going persons I have ever met. And I feel extremely blessed to work with him, and most importantly to learn from him, not only about his way of conducting research but also his philosophy of life.

I also want to thank my parents. They always support me and believe in me no matter what. It is their selfless love that motivates me even when I become hesitated and afraid.

My gratitude extends to my lab members: Jianzhong, Ae, Xiaoqiang, Bing, Francesco and Henry. I still remember how we help each other preparing for the conference presentations and of course all the fun that we had during our group trip. Their support and company made this long and difficult journey more fun and easier.

Finally, I would like to thank Yunde and Yao. They helped me a lot during the very beginning, and really sped up the learning curve so I could start my own research as soon as possible.

Chapter 1

Introduction

1.1 Background of ILC

Iterative learning control (ILC) is a relatively new method of control that aims to achieve zero tracking error of a finite time tracking maneuver that is repeated. The original ILC idea dates back to the late 1970s when Uchiyama introduced the concept on high-speed motion control of a robot arm following a desired trajectory through iterative trials [1]. However, the learning control concept was not widely recognized until the initial explicit formulation of ILC in English was given by Arimoto [2]. In mid 1980s, ILC starts to flourish and receive broad interest due to further development of applications in robots [3-4]. Since then, the literature on ILC has increased significantly. In addition to a considerable amount of journal and conference papers, there are also major surveys [5-9], books [10-13], and special issues [14-16].

ILC has been applied in a wide range of areas. According to the survey performed in 2007 [9], the top three application fields are robots, rotary systems and process control including batch/factory/chemical process. Take computer disk drives for example. The data are written while the disk is rotating so there is vibration which results in some high frequency wiggles in the tracks. In order to read data on a disk, a control system follows these tracks on the disk. At the factory, ILC is used on each track to improve the accuracy when the control system follows the track, so that the storage can be increased. There are also applications to spacecraft operations for repeated scanning maneuvers of fine pointing equipment. The learning process can learn to compensate for both repeating effects from structural flexibility, and deterministic control system error is response

to time varying tracking commands. There is the potential for high precision pointing control achieved through a learning process.

1.2 Instability of ILC Inverse Problem

In general, ILC considers situations in which a control system is to perform the same trajectory repeatedly. The system is returned to the same initial conditions before the start of each repetition. Based on the tracking error observed in the previous run or repetition, the command for the next repetition is adjusted, aiming to make the tracking error converge to zero. Such ILC systems are necessarily digital since data must be stored from each repetition for use in the control updates for the next repetition. The ILC problem seeks the input history needed to produce a desired output history. This is an inverse problem. To find the desired input, one can plug the desired output into the left hand side of the governing difference equation and then that side of the equation becomes a forcing function. The task is then to solve for the general solution which contains a particular solution which is associated with the desired output and the homogeneous solution [17].

An important property of digital control systems is that when a zero order hold is used to feed a continuous time plant, the resulting discrete time z-transfer function is then obtained, but it usually has the property that the discretization has introduced zeros outside the unit circle. Take a third order system with no continuous-time zeros for example, two zeros have been introduced during the discretization. This must happen because when the input to the original differential equation is changed at a time step, the output should change at the next sample time, and this requires a one-time step difference between the most recent input on the right of the equation and the most recent input on the left.

Reference [18] develops the asymptotic locations of the roots introduced in the discretization as the sample time interval tends to zero, as a function of the pole excess in the original Laplace

transfer function, i.e. the number of poles minus the number of zeros. For any Laplace transfer function with pole excess of 3 or more, there will be at least one root introduced that is outside the unit circle for fast enough sample rate.

If the original continuous time system has zeros on the right Laplace plane, there will be another type of non-minimum phase zeros in the corresponding discrete time system. Following the definition in Reference [19], we name the non-minimum phase zeros introduced during the discretization process as sampling zeros, and the images of the original continuous time non-minimum phase zeros as intrinsic zeros.

Therefore, the unique inverse solution for the necessary control action to produce zero tracking error will almost always be growing exponentially with time. For example, a system with pole excess of 3 has a zero at -3.7321 asymptotically as sample rate tends to infinity, then the solution to the homogeneous difference equation consist a constant determined by initial conditions times -3.7321 (using the asymptotic value) to the k^{th} power, where k is the time step number. This solution achieves zero tracking error at every sample time, but the control action required grows exponentially and alternates in sign every time step. It indicates that the actuator will hit saturation after not that many time steps. In addition, the error between sample times also grows exponentially. Thus, the ILC problem is an ill-posed problem whose solution fails to address the intended problem, for any digital system with non-minimum phase zeros.

The ILC problem uses a matrix model of the convolution sum solution of the state space equations, then the inverse of a matrix determines the input history needed to produce the desired output at all time steps. This matrix is analytically guaranteed to have full rank, i.e. it is guaranteed to have an inverse. Reference [20] shows that the singular value decomposition of this finite time matrix is related to the frequency response of the discrete time system. This gives a very attractive

connection between z-transfer function models and time domain state space models. As the dimension of the matrix gets larger, the singular values approach the magnitude frequency response of the system at each discrete frequency that can be observed in the number of time steps in the finite data length. The input and output singular vectors approach sinusoidal functions, and the phase change going through the system is obtained by the phase difference of the corresponding input and output singular vectors.

However, if a digital system contains NMP zeros, there will be one or more particularly small singular values, and both singular values and associated singular vectors do not contain the information of frequency response. For example, a system with one NMP zero has one particularly small singular value. Incidentally, except for this singular value, the other singular values will match the magnitude frequency response of the system as the number of time steps goes to infinity, and is already close at relatively slow sample rate. Moreover, this anomalous singular value decreases as the size of the matrix is increased. Looking at the magnitude of the reciprocal of the unstable zero location taken to a power equal to the size of the matrix, we observe that this slope matches the slope of the smallest singular value. This demonstrates the relationship between the singular value producing the ill-conditioning, and the location of the root of the characteristic polynomial for the right hand side of the difference equation that is outside the unit circle. Reference [21] shows that the number of particularly small singular values is equal to the number of non-minimum phase zeros considering systems with only sampling zeros.

When applying ILC in NMP systems, the error associated with normal singular vectors will decrease and approach zero in a relatively reasonable amount of iterations. However, an astronomical number of iterations can be needed to accomplish any significant learning in the error part associated with the anomalous singular vectors.

In practice, it looks like the iterative process has finished its convergence, but one is perhaps disappointed in the final error level because nothing significant has been accomplished in eliminating error in the part of the error space associated with anomalous singular vectors. One reason is that no one ever does that many of iterations (e.g. 10^{100} for a 3rd order system sampled at desired trajectory p steps long) in hardware. Even if one could perform 10^{100} iterations, the corrective update in error history on any given iteration can be extremely small. If it is so small that the update is beyond the last digit in the digital to analog and analog to digital converters in the hardware, then no accumulation of corrective signal is possible as the iterations progress. The control action in this part of the space is then never updated.

These comments indicate that in practice, the application of ILC laws are mathematically very often subject to this internal instability, but that one never observes the phenomenon. And what one does observe is that the error in all other parts of the space goes to zero, and one is left with the error one had initially in the part of the space associated with the small singular value.

Previous literatures have developed various ways that attempt to solve this problem. Reference [22] proposed an approach based on advanced output data, this approach can stabilize the mapping between input and output for a certain number of systems with non-minimum phase zeros that satisfy a predefined assumptions. Instead of using zero order hold, [23] uses an interpolating hold to solve the problem, and it does for a small class of problems, but it does not solve the problem in general. References [24] proposed a learning strategy based on a reference shift algorithm. The experimental results have shown that this algorithm achieves faster convergence and lower final error in a reasonable time. Stable inversion based approaches can also been found in references [25]-[26]. These approaches can successfully decrease the final error level that converge but fail to converge to zero tracking error if using normal ILC learning laws.

Reference [27]-[30] suggests that leaving a few time steps at the beginning unaddressed can produce extra freedom, doing this can make the system have the correct initial state so that the control action won't grow with time. In practice, one can append several time steps before the execution of desired trajectory, and not asking for zero error at these time steps. The right number of steps need to be added is equal to the number of non-minimum phase zeros. Applying this approach, one can successfully obtain zero tracking error at all the time steps except for the beginning. We call this approach as initial deletion in later chapters.

We know that using a single zero order hold throughout each time step will have the instability problem. Suppose we allow the zero order control action to be updated a number of times within each time step. One can think of this extra zero order hold value as a kind of generalized hold replacing a simple zero order hold. This study shows that this can eliminate the particularly small singular values and vectors which fail to follow the frequency response.

However, while examining the singular values and the singular vectors when using this skip step approach, we observed yet another kind of anomalous singular value in some cases. Compared to the original anomalous singular values, the deviation between the new anomalous singular values and standard frequency response based ones are not as significant. The original ones had the property that the associated input and output singular vectors contained one or more input singular vectors that grow exponentially with time step, and the associated output singular vectors that decay exponentially. However, the new kind of anomalous singular value has different behavior. They are not as small, and hence might be tolerable. And the number of these new anomalous singular values is smaller. When there are a reasonable number of time steps in the trajectory, these singular values become unaffected by increases in the total number of steps in the trajectory, rather than getting progressively worse. Moreover, both input and output singular

vectors decay with time step. This decay with time step suggests that there may be a relationship between these singular values and the decaying of transient in the system, but no connection is established yet.

It is shown that one can modify the problem further so that there are no anomalous singular values. By not asking for zero error at a number of initial time steps equal to the number of new anomalous singular values in the skip step matrix, all singular values will relate to the system frequency response. This then can form the basis of a well posed inverse problem that can be used to design ILC systems.

Based on this structure, a new quadratic cost control law was designed to ask for zero error at each time step, and simultaneously minimizes the tracking error and the control action for each of the introduced intermediate steps. If the penalty is chosen well, the exponential growth of magnitude of the control is eliminated, and one can also aim for improved tracking at the intermediate points. The simulation results show that the error at addressed time steps is numerically zero and the error in between addressed time steps can approach 10^{-10} and even smaller.

Robustness to model parameter uncertainty is particularly important in ILC because it aims to converge to zero error in the real world instead of in our model of the world. Thus, we examine the robustness of the two approaches (initial deletion, multiple zero order holds) when incorporated in three main ILC control laws, namely the Euclidean norm contraction mapping law, the partial isometry law, and a quadratic cost ILC law. Both approaches have similar robustness towards model error, and the quadratic cost law is shown to have the best robustness properties. This result is based on Monte Carlo style evaluation of models based on probability distributions of the coefficients in the model.

Multiple zero order holds and initial deletion approaches have been successfully applied when system only contains the sampling zeros. However, our results suggest that the rules of the two approaches no longer applies when system contains intrinsic zeros. We proposed a new ILC design to incorporate a filter that cancels poles and zeros inside the unit circle. The simulation results show that the design makes the original rules work again.

1.3 Thesis Layout

Chapter 2 gives the mathematical formulas for the ILC problem and the detailed description of the internal instability issue. We first examine the instability problem by looking at poles and zeros of the transfer function. And then we relate the instability to the ill-conditioning of the system matrix. The properties of the singular values and vectors of this matrix are also investigated in this Chapter.

Chapter 3 introduces the multiple zero order holds approach. A new quadratic cost law design is developed utilizing the extra freedom, suggesting one can use this learning law to achieve zero tracking error at addressed time steps and maintain a well-behaved intersample error and control action at the same time. In addition, we show that initial deletion is another effective approach.

Chapter 4 demonstrates the difficulties when extending the two approaches (multiple zero order holds and initial deletion) to systems with intrinsic zeros. Then a new design that incorporates a filter to perform pole-zero cancelation is developed. Simulation shows that it makes the rules of the two approaches effective for more general non-minimum phase system, meaning systems that also contain intrinsic zeros.

Chapter 5 examines the robustness of the two approaches with respect to system parameters uncertainty. Several other approaches are also investigated for comparison purposes. Three basic learning laws are used to perform the robustness test.

Chapter 6 gives a summary of the study, and also a list of potential future topics we may keep exploring.

1.4 References

[1] M. UCHIYAMA, "Formulation of High-Speed Motion Pattern of a Mechanical Arm by Trial," *Transactions of the Society for Instrumentation and Control Engineers*, Vol. 14, 1978, pp. 706-712.

[2] S. ARIMOTO, S. KAWAMURA, AND F. MIYAZAKI, "Bettering Operation of Robots by Learning," *Journal of Robotic Systems*, Vol. 1, No. 2, 1984, pp. 123-140.

[3] G. CASALINO AND G. BARTOLINI, "A Learning Procedure for the Control of Movements of Robotic Manipulators," *Proceedings of the IASTED Symposium on Robotics and Automation*, May 1984, pp. 108-111.

[4] J.J. CRAIG, "Adaptive Control of Manipulators Through Repeated Trials," *Proceedings of the American Control Conference*, June 1984, pp. 1566-1573.

[5] K.L. MOORE, "Iterative learning control - an expository overview," *Applied and Computational Controls, Signal Processing, and Circuits*, Vol. 1, No. 1, 1999, pp. 151-241.

[6] J.-X. XU, "The frontiers of iterative learning control - part I," *Journal of Systems, Control and Information*, Vol. 46, No. 2, 2002, pp. 63-73.

[7] J.-X. XU, "The frontiers of iterative learning control - part II," *Journal of Systems, Control and Information*, Vol. 46, No. 5, 2002, pp. 233-243.

[8] D.A. BRISTOW, M. THARAYIL AND A.G. ALLEYNE, "A survey of iterative learning control," *IEEE Control Systems Magazine*, Vol. 26, No. 3, 2006, pp. 96-114.

[9] H.-S. AHN, Y.Q. CHEN, AND K.L. MOORE, "Iterative learning control: brief survey and categorization," *IEEE Transactions on Systems, Man, and Cybernetics-Part C: Applications and Reviews*, Vol. 37, No. 6, 2007, pp. 1099-1122.

[10] K.L. MOORE, *Iterative Learning Control for Deterministic Systems*, Springer-Verlag, London, U.K., Advances in Industrial Control, 1993.

[11] Z. BIEN AND J.-X. XU, *Iterative Learning Control: Analysis, Design, Integration and Applications*, Kluwer, Boston, 1998.

[12] Y. CHEN AND C. WEN, *Iterative Learning Control: Convergence, Robustness, and Applications*, Springer, London, 1999.

[13] J.-X. XU AND Y. TAN, *Linear and Nonlinear Iterative Learning Control*, Springer, Berlin, 2003.

[14] K.L. MOORE AND J.-X. XU, “Editorial: Iterative Learning Control,” *International Journal of Control*, Vol. 73, No. 10, 2000.

[15] “Iterative Learning Control,” *Asian Journal of Control*, Vol. 4, No. 1, 2002.

[16] H.-S. AHN AND D.A. BRISTOW, “Special Issue on Iterative Learning Control,” *Asian Journal of Control*, Vol. 13, No. 1, 2011.

[17] R.W. LONGMAN, “Iterative Learning Control and Repetitive Control for Engineering Practice,” *International Journal of Control*, Vol. 73, No. 10, 2000, pp. 930-954.

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

[19] M. J. Blachuta, “On zeros of sampled systems,” *American Control Conference*, Albuquerque, NM, 1997, Vol. 5, pp. 3205-3209.

[20] K. Chen and R.W. Longman, "Creating Short Time Equivalents of Frequency Cutoff for Robustness in Learning Control." *Advances in the Astronautical Sciences*, Vol. 114, 2003, pp. 95-114.

[21] Y. Li and R. W. Longman, “Addressing Problems of Instability in Intersample Error in Iterative Learning Control,” Advances in the Astronautical Sciences, Vol. 129, 2008, pp. 1571-1591.

[22] Gu-Min Jeong, Chong-Ho Choi, Iterative learning control for linear discrete time nonminimum phase systems, Automatica, Volume 38, Issue 2, February 2002, Pages 287-291.

[23] 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.

[24] Z. Cai, C. Freeman, E. Rogers and P. Lewin, “Reference Shift Iterative Learning Control for a Non-minimum Phase Plant,” American Control Conference, New York, NY, 2007, pp. 558-563.

[25] S. Devasia, D. Chen, and B. Paden. Nonlinear inversion-based output tracking. IEEE Transactions on Automatic Control, 41(7):930-942, 1996.

[26] T. Sogo, "Stable inversion for nonminimum phase sampled-data systems and its relation with the continuous-time counterpart," Decision and Control, 2002, Proceedings of the 41st IEEE Conference on, 2002, pp. 3730-3735 vol.4.

[27] P. A. LeVoci and R. W. Longman, “Intersample Error in Discrete Time Learning and Repetitive Control,” Proceedings of the 2004 AIAA/AAS Astrodynamics Specialist Conference, providence, RI, Aug. 2004.

[28] R. W. Longman, T. J. Kwon, and P. A. LeVoci, "Making the Learning Control Problem Well Posed – Stabilizing Intersample Error." Advances in the Astronautical Sciences. Vol. 123, 2006, pp. 1143-1162.

[29] Y. Li and R. W. Longman, "Using Underspecification to Eliminate the Usual Instability of Digital System Inverse Models," Advances in the Astronautical Sciences, Vol. 135, 2010, pp. 127-146.

[30] Y. Li and R. W. Longman, "Characterizing and Addressing the Instability of the Control Action in Iterative Learning Control." *Advances in the Astronautical Sciences*. Vol. 136, 2010, pp. 1967-1985.

Chapter 2

The Internal Instability in ILC for NMP Systems

2.1 Introduction

ILC aims to achieve zero tracking error, thus it is trying to converge to the inverse solution. The controllers can be obtained by using the input and output data to create a model, and then invert this model to find the amount of change that the command should produce to eliminate output errors. Non-minimum phase digital systems have zeros outside the unit circle in their discrete time z-transfer functions. Thus, one is likely to encounter difficulties since the inverse transfer function is unstable. In practice, the error often seems to stop decreasing after a number of iterations. If the process is continued until zero error is reached, the control action grows exponentially with time steps. This also makes the intersample error grow exponentially, and its sign alternates every time step when an NMP zero is introduced on the negative real axis.

The learning process requires the use of data from previous runs to update the command used in the next run, and hence it must be digital using sampled data. In digital systems, there are two types of non-minimum phase zeros. The first type is introduced during discretization for systems with pole excess of 3 or more using a zero order hold input. The second is mapping of non-minimum phase zeros of the original continuous time system to their discrete time image. Following Ref. 1, we name the first type as sampling zeros and the second type as intrinsic zeros.

Reference 2 developed an understanding of how the discretization process introduces sampling zeros in the discrete time control system transfer function, when the original continuous time system is fed by a zero order hold input. Unfortunately, when there is a pole excess of three or

more in the continuous time system, the *z*-transfer function discrete time equivalent will have at least one zero introduced outside the unit circle for reasonable sample rates.

References 1 and 2 study the intrinsic zeros. When sampling really fast, these zeros approach 1. And if considering extreme cases when sampling frequency become slower and slower, the intrinsic zeros go to positive infinity first and then suddenly change from positive to negative infinity. It is not easy to connect the location of intrinsic zeros with the locations of continuous time NMP zeros. However, when sampling time is fast, the discrete zero is approximately equal to the exponential of the continuous time zero.

In ILC one usually looks at the finite time problem using a matrix model that includes all time steps of a given run. Reference 3 shows that the singular value decomposition of this finite time matrix is related to the frequency response of the discrete time system. As the dimension of the matrix gets large, the singular values match the magnitude frequency response of the system, and the phase change going through the system is obtained by the phase difference of the corresponding input and output singular vectors.

To invert the system is equivalent to the inverse of the matrix. But it is normally very badly ill-conditioned. And this ill-conditioning is the manifestation of the unstable inverse transfer function. References 4, 5, and 6 study this behavior, among other things extracting frequency response information from the SVD of this matrix. The frequency response result is an asymptotic result, and for any NMP system, there will be one or more anomalous singular values. One can show that these are associated with the zeros outside the unit circle. These unstable poles of the inverted system make the matrix very badly ill conditioned, in spite of the fact that the matrix is analytically guaranteed to be full rank since all eigenvalues are nonzero for a given time delay through the system. The result is that the control action required to obtain zero tracking error at each time step

requires control action that grows exponentially with time. The inverse model produces zero error at every sample time, but the exponentially growing control action magnitudes produce error between sample times that grow exponentially with time step.

This chapter investigates the internal instability of ILC when applied to digital systems with non-minimum phase zeros. We look at the two types of NMP zeros separately. For each type, we look at the instability problem from the perspective of poles and zeros, and then from the analysis of the singular values and vectors.

2.2 Statement of the Learning Control Problem

Consider a single-input, single-output (SISO) system of the general form

$$\begin{aligned} x(k+1) &= Ax(k) + Bu(k) \quad k = 0,1,2,\cdots, p-1 \\ y(k) &= Cx(k) + v(k) \quad k = 1,2,\cdots, p \end{aligned} \tag{2-1}$$

which can be the dynamics of closed loop control system. The $v(k)$ represents any disturbances which occur every trial run appearing anywhere in the control loop, and written in terms of the equivalent disturbance on the output. The $u(k)$ is the input to the control system which is to be adjusted in the learning process, aiming to converge to the input that produces the desired output. The desired trajectory $y_d(k)$ is p steps long. Each iteration, also called run or repetition, is considered to start from the same initial condition $x(0)$. We define vectors giving the whole history of the input and output for the j^{th} repetition as

$$\underline{u}_j = [u_j(0) \quad u_j(1) \quad \cdots \quad u_j(p-1)]^T \tag{2-2}$$

$$\underline{y}_j = [y_j(1) \quad y_j(2) \quad \cdots \quad y_j(p)]^T \tag{2-3}$$

And we also define the desired output history $\underline{y}_d$, the error history $\underline{e}_j = \underline{y}_d - \underline{y}_j$, and the disturbance history $\underline{v}$ analogous to Eq. (2-3). One can write the convolution sum solution of Eq. (2-1) for any time step, and the solution history, and the error history, for any run *j* as follows

$$y(k) = CA^k x(0) + \sum_{i=0}^{k-1} CA^{k-i-1} Bu(i) + v(k) \tag{2-4}$$

$$\begin{aligned} \underline{y}_j &= P\underline{u}_j + \overline{A}x(0) + \underline{v} \\ \underline{e}_j &= -P\underline{u}_j + (\underline{y}_d - \overline{A}x(0) - \underline{v}) = -P\underline{u}_j + \underline{f} \end{aligned} \tag{2-5}$$

where

$$P = \begin{bmatrix} CB & 0 & 0 & \cdots & 0 \\ CAB & CB & 0 & \cdots & 0 \\ CA^2B & CAB & CB & \ddots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ CA^{p-1}B & CA^{p-2}B & CA^{p-3}B & \cdots & CB \end{bmatrix} \quad ; \quad \overline{A} = \begin{bmatrix} CA \\ CA^2 \\ CA^3 \\ \vdots \\ CA^p \end{bmatrix} \tag{2-6}$$

Define a backward difference operator in iterations for any variable $z(k)$ by $\delta_j z(k) = z_j(k) - z_{j-1}(k)$. Then we can write relationships for the change in input history to the resulting change in output history, and to the resulting change in the error history

$$\delta_j \underline{y} = P\delta_j \underline{u} \quad ; \quad \delta_j \underline{e} = -P\delta_j \underline{u} \tag{2-7}$$

The initial conditions and the disturbances are assumed to be the same in every repetition, making $\delta_j x(0) = 0$, $\delta_j v(k) = 0$. We assume that there is a one time step delay going from input to output in Eq. (2-1), which implies that the scalar $CB \neq 0$. Therefore, all eigenvalues are equal to *CB* and matrix *P* is guaranteed to be nonsingular. Simple modifications handle the case when *CB* is zero.

A general linear ILC law takes the form

$$\underline{u}_{j+1} = \underline{u}_j + L\underline{e}_j \quad \text{or} \quad \delta_{j+1}\underline{u} = L\underline{e}_j \tag{2-8}$$

where L is the learning gain matrix. Combining this with the second of Eq. (2-7) produces an equation for the error from iteration to iteration

$$\underline{e}_{j+1} = (I - PL)\underline{e}_j \tag{2-9}$$

One seeks learning gain matrices L that make the magnitude of the largest eigenvalue of $I - PL$ less than one, for convergence to zero error, and one would like to have the largest singular value less than one for monotonic convergence of the Euclidean norm of the error.

Three basic learning laws are often used in our simulation

(1) Contraction Mapping Law (Reference 7): This is a contraction mapping of the Euclidean norm of the tracking error from iteration to iteration. It is also referred as P transpose law. The matrix L in Eq. (2-8) is given as

$$L = \phi P^T \tag{2-10}$$

(2) Partial Isometry Law (Reference 8): High frequency components of the error can converge slowly when using the contraction mapping law. Faster convergence can be obtained using the partial isometry law. Denote the singular value decomposition (SVD) of P by $P = USV^T$, where columns and rows of U and V are orthonormal unit vectors of dimension p. Thus the inverse of U and V are equal to their transpose. The partial isometry law is

$$L = \phi V U^T \tag{2-11}$$

(3) Quadratic Cost Law (Reference 9): This control law is designed by minimizing a quadratic cost function

$$J_{j+1} = \underline{e}^T{}_{j+1}\underline{e}_{j+1} + r\delta_{j+1}\underline{u}^T\delta_{j+1}\underline{u} \tag{2-12}$$

The r is a scalar gain which controls the speed of convergence from iteration to iteration. Differentiating J with respect to u, gives

$$L = (P^T P + rI)^{-1} P^T \tag{2-13}$$

2.3 Internal Instability Due to Sampling Zeros

First, we consider NMP system with only sampling zeros. These zeros are introduced during discretization using a zero order hold. They move away from the unit circle towards asymptotical locations when sampling faster, and approach -1 when sampling slower. If sampling really slow, all extra zeros will go inside the unit circle. For reasonable sample rates, the pole excess determines the number of unstable poles in the discrete time inverse model. We study the behavior of the singular values of the Toeplitz matrix P as a function of pole excess, and also the associated singular vectors.

2.3.1 Models Used to Investigate Behavior of Different Pole Excess

The following continuous time systems are used to generate the discrete-time systems which contain NMP sampling zeros. These transfer functions have no zeros in continuous time, and the subscript is the system order and the pole excess.

$$\begin{gathered}
G_1(s) = \frac{a}{s+a};\ G_2(s) = \frac{\omega_1^{\,2}}{s^2 + 2\zeta_1\omega_1 s + \omega_1^{\,2}};\ G_3(s) = G_1(s)G_2(s) \\
G_4(s) = \frac{G_2(s)\cdot\omega_2^{\,2}}{s^2 + 2\zeta_2\omega_2 s + \omega_2^{\,2}};\ G_5(s) = G_1(s)G_4(s);\ G_6(s) = \frac{G_4(s)\cdot\omega_3^{\,2}}{s^2 + 2\zeta_3\omega_3 s + \omega_3^{\,2}} \\
G_7(s) = G_1(s)G_6(s);\ G_8(s) = \frac{G_6(s)\cdot\omega_4^{\,2}}{s^2 + 2\zeta_4\omega_4 s + \omega_4^{\,2}};\ G_9(s) = G_1(s)G_8(s) \\
G_{10}(s) = \frac{G_8(s)\cdot\omega_5^{\,2}}{s^2 + 2\zeta_5\omega_5 s + \omega_5^{\,2}};\ G_{11}(s) = G_1(s)G_8(s)
\end{gathered} \tag{2-14}$$

The constants are $a = 8.8$, $\zeta_1 = \zeta_2 = \zeta_3 = \zeta_4 = \zeta_5 = 0.5$, ω_1, ω_2, ω_3, ω_4, ω_5 equal 37, 125.7, 62.8, 94.2 and 157.1 rad/sec, approximately 6, 20, 10, 15, and 25 Hz. Note that these frequencies increase monotonically except for ω_2.

2.3.2 The Ill-posedness of the ILC Inverse Problem (Poles and Zeros)

To illustrate the instability of the inverse problem addressed by ILC, consider the third order model presented in previous section. Then there is an equivalent discrete time z-transfer function $G(z)$

$$G(s) = \left(\frac{a}{s+a}\right)\left(\frac{\omega_0^2}{s^2 + 2\zeta\omega_0 s + \omega_0^2}\right) \; ; \; G(z) = \frac{b_2 z^2 + b_1 z + b_0}{z^3 + a_2 z^2 + a_1 z + a_0} \tag{2-15}$$

where $a = 8.8$, $\zeta = 0.5$, $\omega_0 = 37$. The discrete time equivalent difference equation relating input to output has the form

$$y(k+3) + a_2 y(k+2) + a_1 y(k+1) + a_0 y(k) = b_2 u(k+2) + b_1 u(k+1) + b_0 u(k) \tag{2-16}$$

On the right hand side, two zeros have been introduced from the discretization process. When a differential equation is fed by a zero order hold, a change in the input at one time step will produce a resulting change seen for the first time in the output at the next sample time step, i.e. there is a one time-step delay in the discrete time model. This implies that the most recent time step on the right hand side of Eq. (2-16) must be one step behind the most recent time step on the left hand side. Hence, generically an n^{th} order differential equation converted to discrete time will have $n-1$ zeros. These contain images of any zeros in the original differential equation, and the rest are introduced by the discretization. The number of original poles minus original zeros is termed the pole excess of the transfer function of the differential equation, and generically the pole excess of

the discrete time transfer function is unity. (The one time step delay applies unless the differential equation has the same number of zeros as poles which is close to being non-physical, or if the sample time is so slow that the unit pulse response output happens to be going through zero at the next sample time, i.e. $CB = 0$.)

An iterative learning control law aims to converge to the input history needed to produce the desired output history. Once it has converged, looking at Eq. (2-16), the solution can be viewed as follows. One substitutes the desired output into the left hand side, and then one need to solve for the input sequence $u(k)$ that will satisfy the equation. The left hand side has now become a forcing function, and we need to solve the resulting nonhomogeneous difference equation. The solution contains a particular solution, and the solution of the associated homogeneous equation whose characteristic polynomial is $b_2 z^2 + b_1 z + b_0 = 0$, the polynomial of the zeros of the *z*-transfer function. If the zeros are r_1, r_2, then the solution for $u(k)$ contains $C_1(r_1)^k + C_2(r_2)^k$, where the arbitrary constants are determined by the initial conditions.

Reference 2 develops the asymptotic locations of the roots introduced in the discretization as the sample time interval *T* tends to zero, as a function of the pole excess in the original Laplace transfer function, i.e. the number of poles minus the number of zeros. These are presented in Table 1-1. They are all on the negative real axis (corresponding to Nyquist frequency). Odd pole excesses introduce an even number of zeros, half of which are inside the unit circle and the other half are outside the unit circle, located at the reciprocals of those inside. Even pole excesses introduce an odd number of zeros, and this extra one is asymptotically located at -1.

For pole excess of 3 as in Eq. (2-15) the asymptotic locations for r_1, r_2 are -3.7321 and -1/3.7321 (the sample rate has to be below 10 Hz to have the outside zero move inside the unit circle, too

slow for reasonable control of the system). The $u(k)$ is a solution of an unstable difference equation with one solution growing exponentially with time step, but alternating sign every time step, and it will dominate the solution after a few time steps. For dramatic impact, imagine the solution for pole excess of 7 sampled at 1000 Hz, that would contain an arbitrary constant based on initial conditions times -109.305 to the 1000th power at the end of one second. Note also, for every zero outside the unit circle, the asymptotic locations contain corresponding zeros inside the unit circle at the reciprocal location.

Table 1-1. Asymptotic zero locations outside and on the unit circle.

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

There is a mathematical solution that achieves zero tracking error at every sample time, but the control action grows exponentially and alternates in sign every time step (because the dominant solution is a negative number to the k^{th} power). The fact that the control action grows exponentially at the above rate means that the solution is impractical in the sense that the actuator will hit saturation after not that many steps. But it fails to be a solution of the intended problem in a more

serious way. The exponential growth with alternating sign makes the system move in opposite directions between one pair of time steps and the next pair, at an accelerating rate. At every sample time, the error is going through zero, but between sample times the tracking error is growing exponentially. Thus, the ILC problem is an ill-posed problem whose solution fails to address the intended problem, for any system with a pole excess of 3 or more, fed by a zero order hold, and with a fast enough sampling rate.

Considering the same third order system sampled at 100 Hz, the left plot in Figure 2-1 shows the control action for the first 20 time steps, on a logarithmic plot. One can see that the control action grows to around 10^{10} and changes sign every time step after only 20 time steps. The right plot shows the zero error for time step 11 to 20 marked with asterisk, and shows the growth of the error between these time steps, which reaches 10^6 in magnitude.

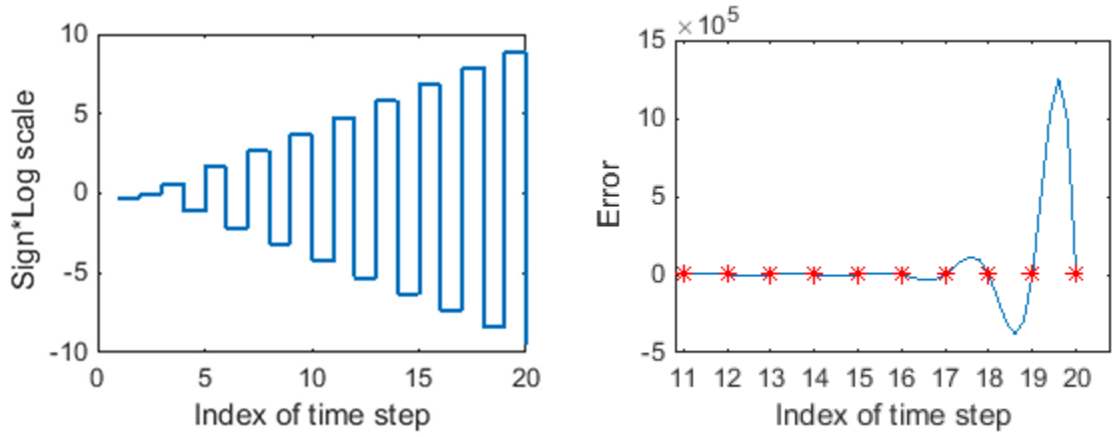

Figure 2-1. The intersample error vs. time step and the control action on a logarithmic scale using the inverse solution with only one zero order hold between addressed time steps.

2.3.3 Comments on the Singular Values and Singular Vectors of *P* Matrix

ILC uses a matrix model to represent the system dynamics as shown in Eq. (2-6). Reference 3 provides some insight concerning the meaning of the singular values and the singular vectors of the Toeplitz matrix *P*. We sketch the development given in that reference. Consider Eq. (2-5) with

the initial condition set to zero and the disturbance set to zero, $\underline{y} = P\underline{u}$. This represents the convolution sum solution for each time step from Eq. (2-4), which is also the output obtained from the transfer function times the input, which automatically assumes zero initial conditions and no disturbance. We can take the discrete Fourier transform (DFT) of the input and output history as $\underline{U} = H\underline{u}$ and $\underline{Y} = H\underline{y}$ where

$$H = \begin{bmatrix} (z_o^0)^0 & (z_o^0)^{-1} & \cdots & (z_o^0)^{-(p-1)} \\ (z_o^1)^0 & (z_o^1)^{-1} & \cdots & (z_o^1)^{-(p-1)} \\ \vdots & \vdots & \ddots & \vdots \\ (z_o^{p-1})^0 & (z_o^{p-1})^{-1} & \cdots & (z_o^{p-1})^{-(p-1)} \end{bmatrix} \quad H^{-1} = (1/p)(H^*)^T \quad z_o = e^{i\omega_o} \quad \omega_o = (2\pi/p) \quad (2\text{-}17)$$

The p is the number of entries in the column vector histories, and superscript asterisk indicates complex conjugate. Then we apply this to $\underline{y} = P\underline{u}$ obtaining $\underline{Y} = (HPH^{-1})\underline{U} = (\hat{H}P\hat{H}^{-1})\underline{U} = E\underline{U}$ where $\hat{H} = (1/\sqrt{p})H$ and $\hat{H}^{-1} = (\hat{H}^*)^T$ contains normalized columns and rows. Therefore, the frequency components of the input and output histories are related by

$$\underline{Y} = E\underline{U} \qquad E = \hat{H}P(\hat{H}^*)^T \tag{2-18}$$

$$P = (\hat{H}^*)^T E\hat{H} \tag{2-19}$$

In the limit as p tends to infinity, Eq. (2-18) tends to the steady state frequency response of the system. The DFT frequency response $Y_n e^{i(\omega_o n)k} = G(e^{i(\omega_o n)})U_n e^{i(\omega_o n)k} = M_n e^{i\theta_n} U_n e^{i(\omega_o n)k}$ is given in terms of complex exponential inputs, and to get the sine and cosine response one adds or subtracts the complex conjugate responses. The G is the discrete time z-transfer function $G(z)$ with magnitude and phase responses as shown for the discrete frequencies observed in the finite number

p of data points. In the limit, a single frequency input results in a single frequency output with the amplitude and phase change of $G(z)$, so we can know the value of E

$$E = diag(M_0 e^{i\theta_0}, M_1 e^{i\theta_1}, ..., M_{p-1} e^{i\theta_{p-1}}) \tag{2-20}$$

One can now eliminate complex functions since P is a real matrix, with the result that the singular value decomposition $P = USV^T$ becomes the frequency response of the system, with S containing the magnitude response M_n, the singular vectors converging to sinusoids, with the phase change θ_n through the system being the result of the phase difference between the input singular vectors in V and the output singular vectors in U.

For large p, this development suggests that each singular value will be related to a specific frequency, and which frequency can be determined by doing a DFT of the associated singular vectors. If the frequency response of the system is monotonic, then the monotonic descent of the singular values produced by a singular value decomposition routine exhibit the same order as the discrete set of frequencies visible in p steps. Otherwise some reordering can be needed. Note that in the limit there will be two singular values for each frequency, since the space associated with one frequency needs two functions like sine and cosine to span the space.

This thinking can apply for large p when essentially everything considered is steady state. However, our P is a finite time and must also represent finite time phenomena. Viewed in terms of transfer functions, the inverse system is governed by the reciprocal of the transfer function whose poles are the original zeros and hence unstable. Viewed in terms of Eq. (2-16) whose solution to get zero error in the next run requires the inverse of matrix P, this matrix must be ill conditioned. We study this below.

2.3.4 Anomalous Singular Values and Singular Vectors of Full Toeplitz Matrix *P*

The analysis given above indicates that the singular values of matrix *P* match the steady state magnitude frequency response of the system as the number of time steps in the trajectory *p* tends to infinity, i.e. the dimension of *P* goes to infinity. We will see that this relationship remains close for rather short trajectories as well (Reference 4). However, for pole excesses of 3 or more there are one or more singular values that do not lie near the frequency response curve, and we call these anomalous or 'bad' singular values of *P*. These singular values make *P* ill conditioned, and make the zero tracking error solution at every time step not usable. This section examines these anomalous singular values and associated singular vectors for the full matrix *P*.

The Number of Anomalous Singular Values

Figures 2-2 and 2-3 consider systems with pole excess of 3 up to 10 using the models in Eq. (2-14) fed by a zero order hold sampling at 100 Hz, Nyquist frequency is 50 Hz, and the *P* matrices are square 50 by 50 matrices. There are 50 singular values most of which correspond to discrete frequencies that can be observed in 50 time steps of data. With this even number of time steps, the DFT computation in Eqs. (2-17) to (2-20) will produce a single singular value that must map to DC as the matrix gets large, 2 singular values that become associated with each individual frequency between zero and Nyquist (48 such evenly spaced frequencies in this case) as the matrix size increases, and one singular value related to Nyquist frequency. Provided the magnitude frequency response is monotonically decreasing (it is not completely true for the systems in Eq. (2-14), but nearly so), the singular values follow the frequency response curve. This gives the mapping of singular values to frequencies in Hz that becomes more exact as the matrix size increases. If it is not monotonic, Reference 3 tells one to do a DFT of the associated singular vectors to know the appropriate frequency. The singular vectors may show an isolated peak for

one singular value, alternating with a peak having two discrete frequencies in it, but in the limit there must be two identical singular values for each frequency, and the mixed case must disappear. Figures 2-2 and 2-3 show the frequency response curve as a solid line, and asterisks are used for each singular value. We see that the singular values lie essentially on top of the magnitude response curve, except for the few anomalous smallest singular values. For systems of pole excess of three or more, at least one anomalous singular value appears, and the number of such particularly small singular values for various pole excesses is shown in the third column of Table 1-2, the numbers are equal to the number of zeros outside the unit circle.

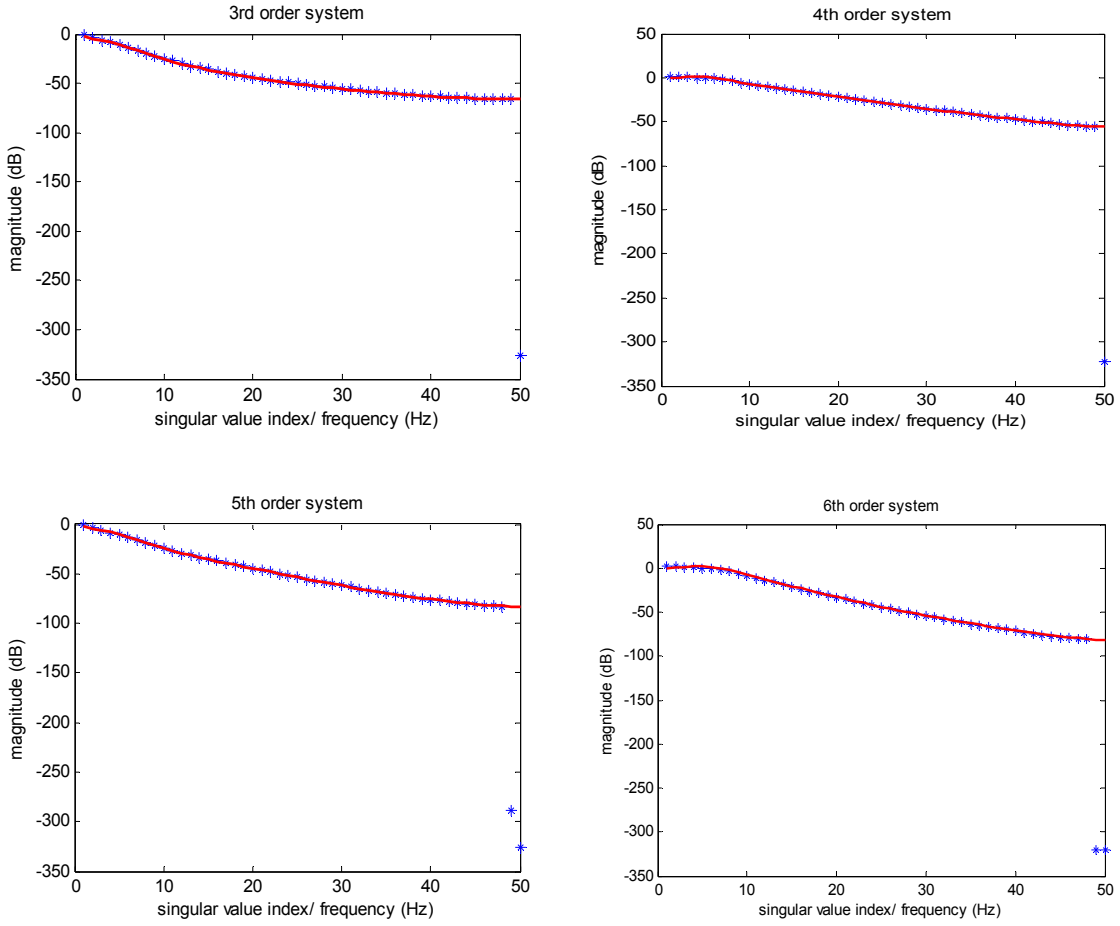

Figure 2-2. Singular Values and Magnitude Frequency Response (pole excesses of 3, 4, 5, and 6)

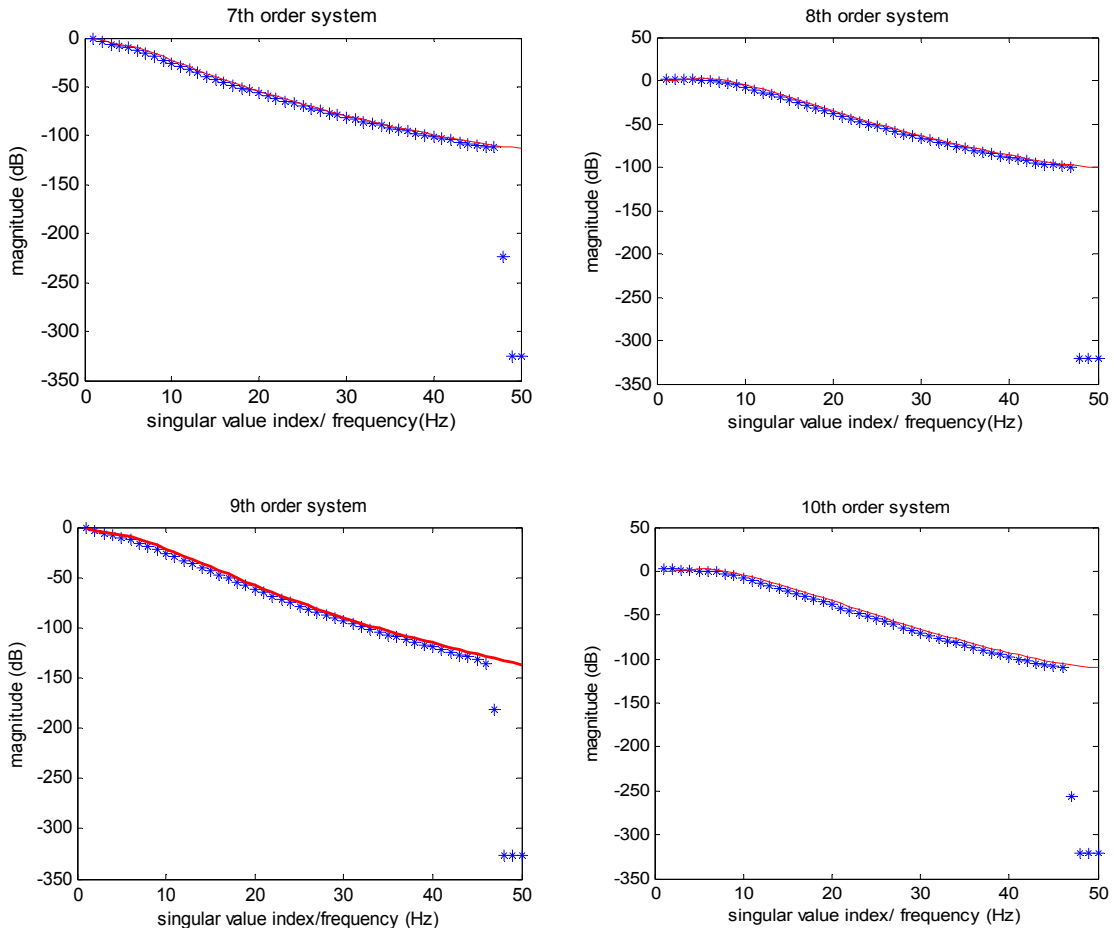

Figure 2-3. Singular Values and Magnitude Frequency Response (pole excesses of 7, 8, 9, and 10)

Table 1-2. Number of Particularly Small Singular Values

Pole excess	Number of zeros outside the unit circle	Number of anomalous singular values
2	0	0
3	1	1
4	1	1
5	2	2
6	2	2
7	3	3
8	3	3
9	4	4
10	4	4
11	5	5

The Anomalous Singular Values as a Function of Dimension of P

We comment that ILC often pushes the limits of numerical computation, and the MATLAB algorithm used to compute the singular values given above is often unable to give correct values. For example, looking at the bottom of Figure 2-3 it appears that there are 3 equal singular values, but this is most likely not the case. Figure 2-4 looks at the anomalous singular value for the 3rd order model as a function of matrix size p using sample time $T = 0.02$. After about $p = 35$ the singular value stops decaying and becomes numerical noise somewhere between -300 and -400 dB. One can extrapolate this curve to estimate the true value. For a 50 time step problem at 50 Hz sample rate, extrapolating to $p = 50$, produces a smallest singular value roughly equal to -500 in dB corresponding to 10 to the -25 power. Of course, this indicates a very ill conditioned matrix, but nevertheless the lower triangular matrix is analytically guaranteed to be full rank since all eigenvalues are $CB \neq 0$.

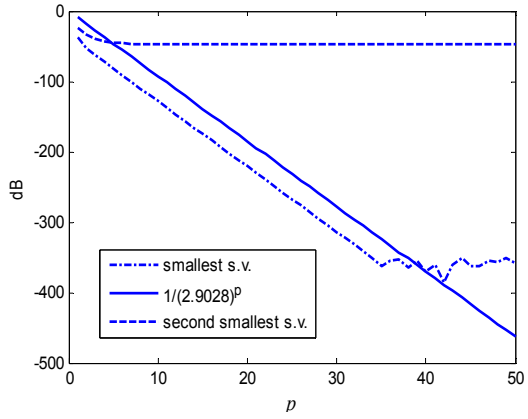

Figure 2-4. Smallest and Next to Smallest Singular Value vs. Size of *P*.

Also shown in Figure 2-4 is a solid line corresponding to the value of the magnitude of the reciprocal of the "unstable" zero location for this system, taken to a power equal to the size of the matrix P. This is the root that asymptotically reaches -3.7321 as sample time interval T tends to

zero. It equals –2.9028 for the 50Hz sample rate. We see that the slope matches the slope of the smallest singular value. This demonstrates that the anomalous singular values are associated with the zeros outside the unit circle in the discrete time transfer function introduced using a zero order hold input, in this case the right hand side of Eq. (2-16).

The Input and Output Anomalous Singular Vectors

If one examines the right and left singular vectors associated with singular values related to the magnitude frequency response curve, they resemble sinusoids. Now we examine the singular vectors associated with the anomalous singular values. Figure 2-5 considers the same third order system, left plot for the 50 components of the output singular vector in U associated with the one anomalous singular value, and the right plot shows the components of the corresponding input singular vector in V.

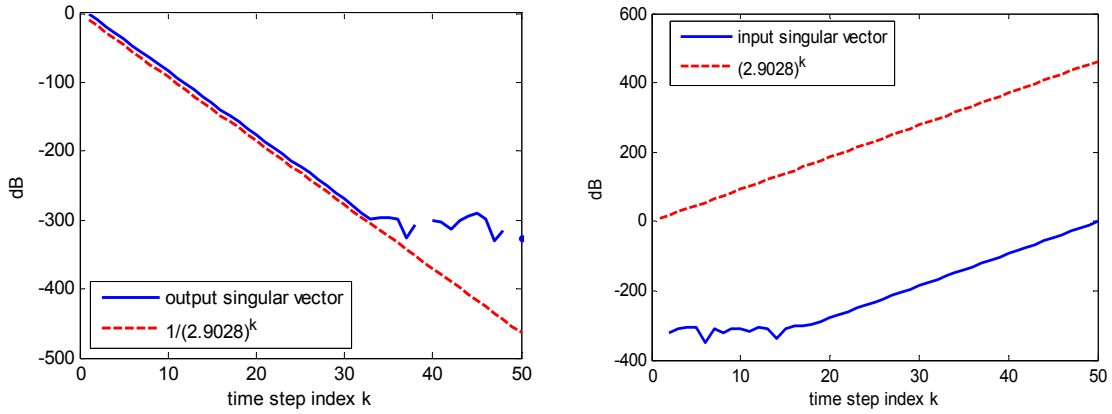

Figure 2-5. Magnitudes of the Components of Last Output and Input Singular Vectors. Also shown, Magnitudes of Reciprocal of Zero Location and Zero Location to the k^{th} Power (pole excess of 3)

The dashed line represents the absolute value of the zero location outside the unit circle and its reciprocal each taken to the power k equal to the time step, or row number. The slope of the output singular vector equals the absolute value of $(-1/2.9028)^k$, and the slope of the input singular vector

equals the absolute value of $(-2.9028)^k$. These produce straight line slopes on the dB scales. Again we see that the numerical computation is unable to compute all of the components of these vectors accurately when these values are too small. It appears counterintuitive that a big error at the beginning of a trajectory equal to the output singular vector, requires an input of the form of the corresponding input singular vector. In other words, to correct an error at the beginning requires control action that is very large at the final step. But this is a result of the instability of the inverse system.

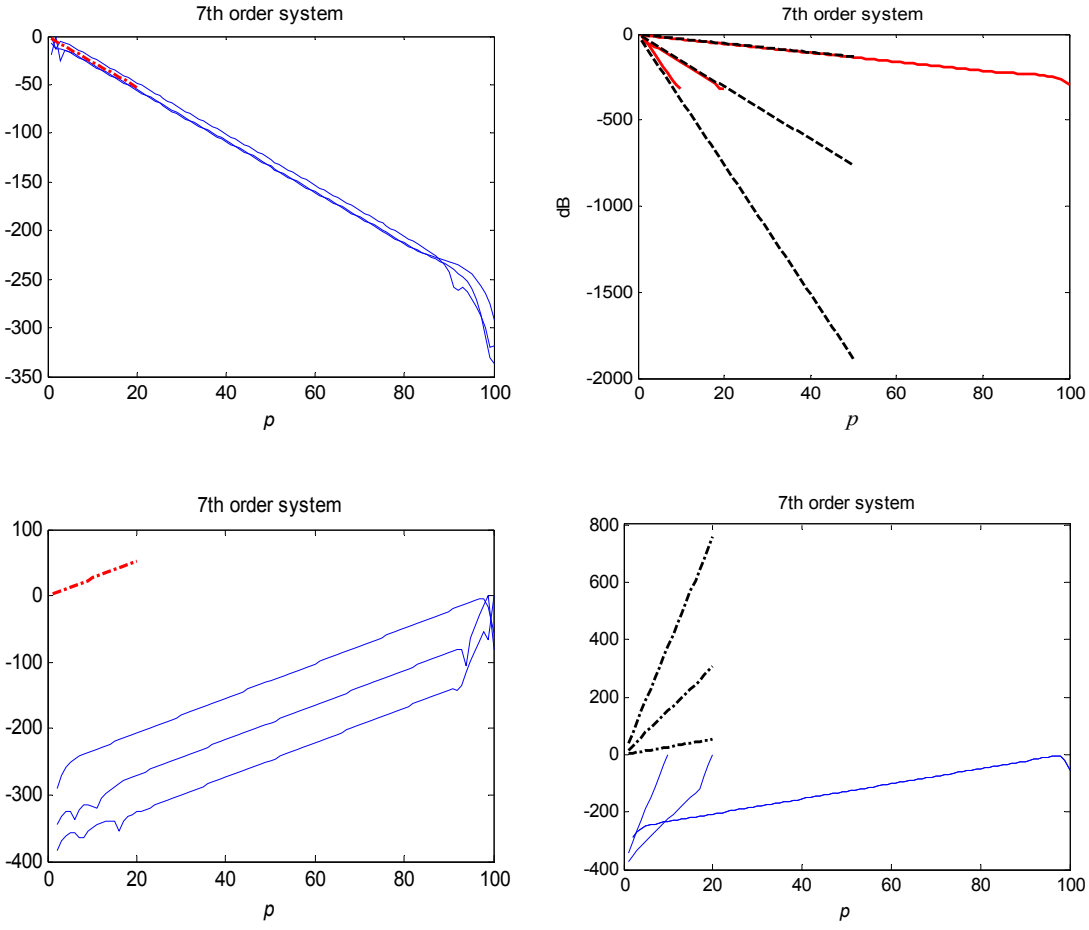

Figure 2-6. Top left, magnitudes of the last three output singular vector components for 100 by 100 *P* matrix for 7th order model. Top right, solid line is magnitude of last output singular vector components for 10 by 10, 20 by 20, and 100 by 100 *P* matrices. Also shown are slopes associated with zeros outside unit circle. Bottom plots are corresponding plots for input singular vectors.

Now consider a larger pole excess so that there are several anomalous singular values. Based on the result in Figure 2-5, one expects that there will be a different slope for the absolute values of the singular vector components for each singular value, and each corresponds to one of the zeros outside the unit circle. Initially we have trouble observing this. Consider the seventh order system, sampling at 0.01 sec, there are three zeros outside the unit circle, which are equal to -1.3528, -5.7910, and -77.8752. Figure 2-3 shows that the last two singular values are computationally almost the same. The top left plot in Figure 2-6 shows the absolute values of the components of the three singular vectors associated with the three smallest singular values. Unexpectedly, all these curves are almost identical in terms of slope and the slope corresponds to the absolute value of (-1/1.3528) taken to the k^{th} power shown as a dotted line. To investigate this phenomenon, the top right of Figure 2-6 makes a plot of the magnitudes of the components vs. time step for the output singular vector associated with the smallest singular value, for three different size P matrices, 10 by 10, 20 by 20, and 100 by 100. The plot shows all three expected slopes associated with three zeros outside the unit circle as dotted lines. When p is 10, the smallest singular value has an output singular vector with slope related to the decay of (1/77.8752) associated with the zero furthest outside the unit circle. When p is 20 the steepest slope is no longer present and the last output singular vector exhibits the slope associated with the zero -5.7910, and when p is 100 the singular vector for the smallest singular value has the slope associated with the zero at -1.3528. One expects that the numerical algorithms are not capable of correctly computing the actual smallest singular values, and associated singular vectors. The bottom of Figure 2-6 gives the corresponding figures for the input singular vectors.

2.4 Internal Instability Due to Intrinsic Zeros

Intrinsic zeros are images of the original continuous time NMP zeros. It is not practical to use a set of models to generalize this type of NMP system. Thus, here we demonstrate the instability problem consider a simple system with one zero at 0.8 in continuous-time

$$\frac{0.8-s}{(s+1.6)(s+0.5)} \tag{2-21}$$

when sampled at 10 Hz using a zero order hold, there is a discrete time zero located at $z = 1.0835$. Following the same analysis for sampling zeros, the inverse solution in this case contains an exponentially growing term which is equal to a constant multiplied by $(1.0835)^k$, and k reaches the total time steps in the desired trajectory.

As shown in Fig. 2-7, the left plot is the control action, it grows to nearly magnitude 6000 in 100 time steps, and the right plot is the error if using this inverse solution. The error for all time from time step 91 to time step 100 is shown in the plot. The errors at the sampled time steps are zero, but the error between sampled times grows exponentially from step to step.

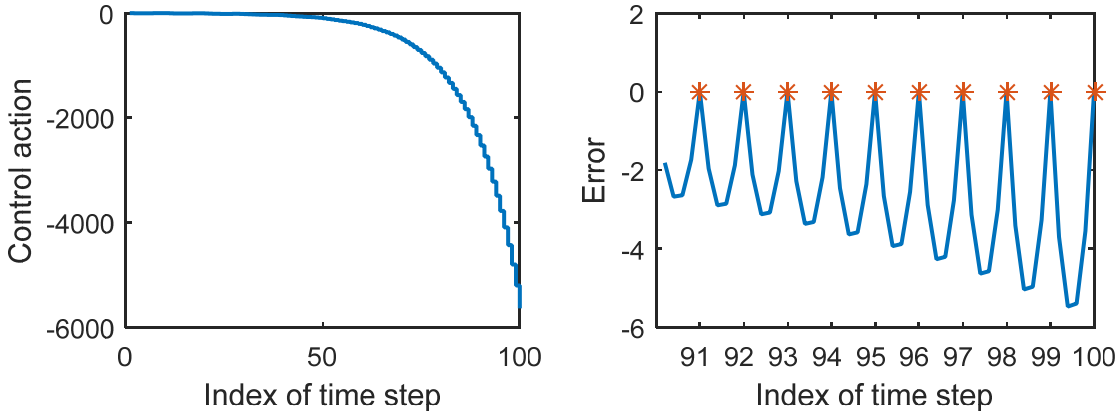

Figure 2-7. Control action (left) and error (right) if using inverse solution for system with intrinsic zeros

The left plot in Fig. 2-8 shows the singular values of a 100 by 100 P matrix, there is one bad singular value at around 10^{-4}. The associated singular vectors are shown in the right plot, one can

observe the same behavior as for the sampling zeros, the output singular vector decays exponentially and the input singular vector grows exponentially. Figure 2-9 shows the same behavior as we observed in sampling zeros. The anomalous singular value get worse as the size of the *P* matrix increases, decreasing by a factor of the reciprocal of the zero location magnitude for each unit increase in dimension (solid line). This suggests that the anomalous singular value is related to the NMP zero.

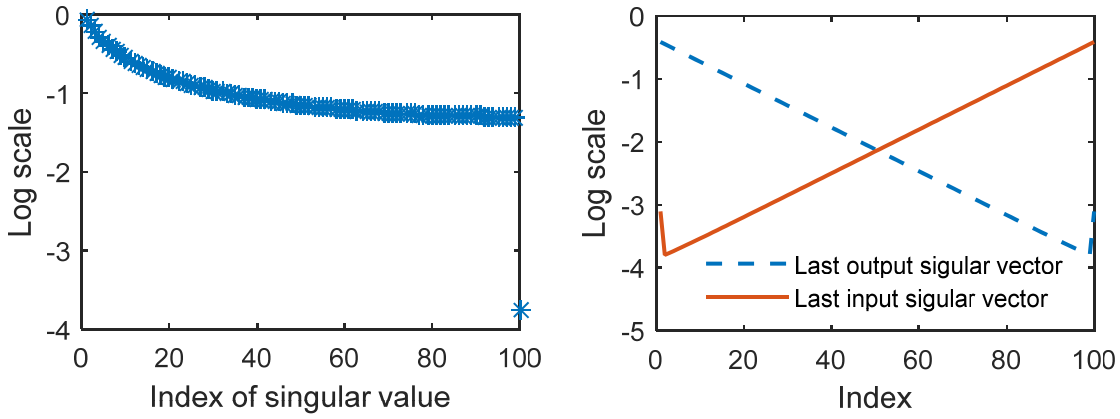

Figure 2-8. Singular values (left), and singular vectors associated with the bad singular value (right)

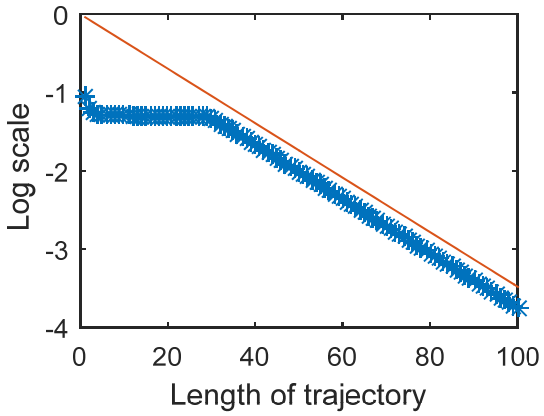

Figure 2-9. Magnitude of the anomalous singular value vs length of trajectory

2.5 What Happens in the Real World?

However, in the real world, one usually won't see this instability of the control action needed for zero error at the sample times. What one usually observes is that the error seems to stop improving, and appears as if it has converged to a non-zero and disappointing level. An example is shown in Fig. 2-10. Consider the same 3rd order system in Eq. (2-15) sampled at 100 Hz. Assume the desired output is $1-\cos(2\pi t)$ and the disturbances and initial conditions are zero. Given the desired output as the command in the first iteration, we use the *P* transpose law and perform 10,000 iterations. The left plot is the root mean square (RMS) error for each iteration, versus iteration number. It decays very fast in the first few iterations and then stops at around 10^{-4}. The right plot gives the control action vs. time step in the last iteration. Except for some wiggles at the beginning and at the end, the control action is well behaved.

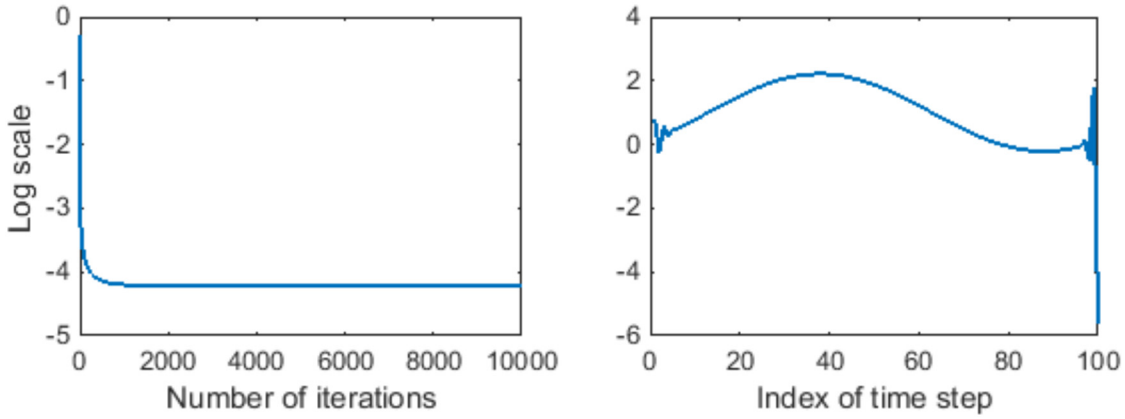

Figure 2-10. The RMS error in different iterations (left) and the control action in the 10,000th iteration (right)

Figure 2-11 gives the same simulation result for system in Eq.(2-21) sampled at 10 Hz. One can see the decay of the RMS error with iterations for 1000 iterations, and the RMS error seems to stop decaying after it reaches RMS error equal one. The input time history at iteration 1000 is

given in the right plot of the figure. No internal instability is visible within the first 1000 iterations, but the error seems to have stopped converging.

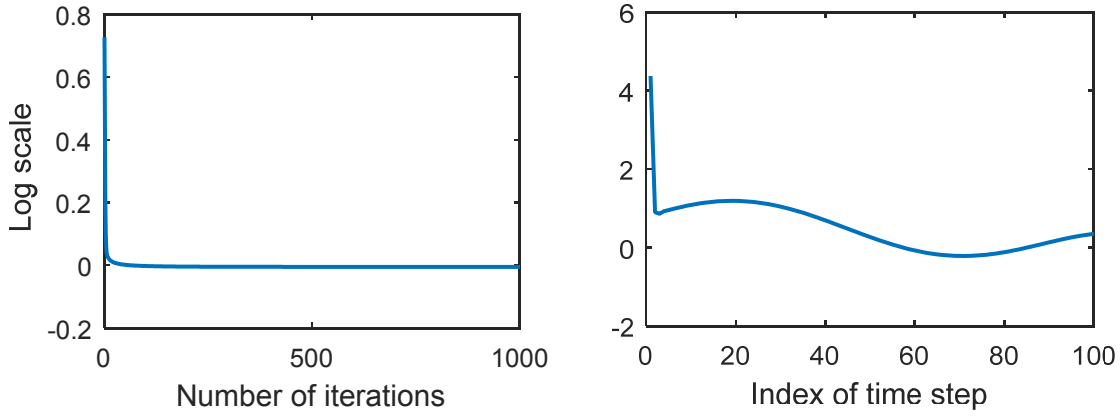

Figure 2-11. RMS error in different iterations (left) and the control action in the 1000th iteration (right)

It is of interest to examine how the particularly small singular value influences the performance of an iterative learning control law. Consider the quadratic cost ILC law in Eq. (2-13). Denote the singular value decomposition of matrix P by

$$P = USV^T \tag{2-22}$$

$$U = \begin{bmatrix} \bar{u}_1 & \bar{u}_2 & \cdots & \bar{u}_p \end{bmatrix} \qquad V = \begin{bmatrix} \bar{v}_1 & \bar{v}_2 & \cdots & \bar{v}_p \end{bmatrix} \tag{2-23}$$

and S is the square diagonal matrix of singular values. We will look at only the behavior associated with the smallest singular value. Totally analogous equations apply to other small singular values when there is more than one zero outside the unit circle introduced by the discretization. The vectors $\bar{u}_p$ and $\bar{v}_p$ are the singular vectors associated with the smallest singular value in the bottom right corner of the matrix S, denoted $\sigma_{\min}$. Convert the error vector to a new set of coordinates using the orthonormal basis functions in U, i.e. let $\underline{\varepsilon}_i = U^T \underline{e}_j$. Then the component of the error that is in the part of the space associated with the small singular value is $\varepsilon_j(p) = \bar{u}_p^T \underline{e}_j$

where the argument p indicates the p^{th} component, and P is a p by p matrix. For simplicity, let the ILC start up conditions be $\underline{u}_0 = 0$, so that $\underline{y}_0 = 0$ and $\underline{e}_0 = \underline{y}_d$. In these coordinates Eq. (2-13) becomes

$$\underline{\varepsilon}_{j+1} = [I - S(S^2 + rI)^{-1} S]\underline{\varepsilon}_j \tag{2-24}$$

and the initial error $\varepsilon_0(p)$ in the part of error space under study, evolves with iterations according to

$$\varepsilon_j(p) = R_{QC}\varepsilon_{j-1}(p) = R_{QC}^j \varepsilon_0(p) \quad ; \quad R_{QC} = r/(\sigma_{\min}^2 + r) = [1 + (\sigma_{\min}^2 / r)]^{-1} \tag{2-25}$$

Now consider the change in the control actions with iterations for this part of the error space. Use the singular value decomposition in Eq. (2-13), and apply the result to Eq. (2-8) to produce

$$\underline{u}_{j+1} = \underline{u}_j + [\sigma_{\min}/(\sigma_{\min}^2 + r)]\varepsilon_j(p)\overline{v}_p \ ; \ \varepsilon_j(p) = R_{QC}\varepsilon_{j-1}(p) = R_{QC}^j \varepsilon_0(p) \tag{2-26}$$

The input at iteration $j = N$ is

$$\begin{aligned} \underline{u}_N &= [\sigma_{\min}/(\sigma_{\min}^2 + r)]\varepsilon_0(p)[R_{QC}^0 + R_{QC}^1 + \cdots + R_{QC}^{N-1}]\overline{v}_p \\ &= [\sigma_{\min}/(\sigma_{\min}^2 + r)]\varepsilon_0(p)[(1 - R_{QC}^N)/(1 - R_{QT})]\overline{v}_p \\ &= \frac{1}{\sigma_{\min}}[1 - (R_{QC})^N]\varepsilon_0(p)\overline{v}_p \end{aligned} \tag{2-27}$$

Note that for $N = 1$ the factors multiplying $\varepsilon_0(p)\overline{v}_p$ are approximately $\sigma_{\min}/r$ so the control action starts with a very small amplitude for this part of the space. But eventually it converges to

$$\underline{u}_\infty = \frac{1}{\sigma_{\min}}\varepsilon_0(p)\overline{v}_p \tag{2-28}$$

If we say that the smallest singular value is 10^{-50} (consider the case in Fig. 2-4 when p is 100), this could require a control action of the size 10^{+50} times the initial error to be corrected in this part of the error space.

Let us examine the learning rate in Eqs. (2-25) and (2-27). From Eq. (2-25), after N iterations, the initial error component in this part of the space has been decreased by the factor $[1+(\sigma_{\min}^2/r)]^{-N} = 1 - N(\sigma_{\min}^2/r) + \cdots$. A rough estimate of how many iterations N are needed to learn could be made by finding when the second term on the right is of the same order of magnitude as the initial unity term, and this happens when $N \sim (1/\sigma_{\min}^2)r$. If the true $\sigma_{\min}$ is something like 10^{-50}, then we need to have N equal something like $10^{100}r$. Using any reasonable value for r in the cost functional, this is truly an astronomical number of iterations needed to accomplish any significant learning in this part of the space.

The analysis of the convergence speed for Eq. (2-27) is similar. Denote $(\sigma_{\min}^2/r)$ by α. Then the coefficient on the right of Eq. (2-27) multiplying the initial error, after using the same expansion as above, is $(1/\sigma_{\min})[1-(1+\alpha)^{-N}] \approx N\sigma_{\min}/r$. In order for $\underline{u}_N$ to make a significant change in the output for error in this part of the space, it has to be of the magnitude of $(1/\sigma_{\min})$, this magnitude can then be associated with having $(1/\sigma_{\min}) \sim N\sigma_{\min}/r$. This produces the same astronomical number for N.

Normally, one cannot use unstable control systems in practical applications. This is equally true when the instability is an internal instability, as in this case in which the control action magnitude grows exponentially with time step according to the reciprocal of the zero location to the power of the time step. The fact that we are only interested in finite time trajectories can only help if the

number of time steps in the trajectory is very small. The irony is that in application of ILC laws designed to get zero tracking error, there is this internal instability, and nevertheless the laws significantly reduce the error, and no difficulty is encountered related to the internal instability. Of course, the reason is that no one ever does 10^{100} iterations in hardware. If the singular value is sufficiently small, then the real world behavior for a reasonable value of N is that nothing significant has been accomplished in eliminating error in this $\bar{u}_p$ part of the error space, one feels like the iterative process has finished its convergence, but one is perhaps disappointed in the final error level. One often wonders, why did the ILC law not make more progress in eliminating the tracking error? After enough iterations N to make the error negligible associated with all singular values but the smallest one, and for an N that has made no significant progress in eliminating the error associated with the smallest singular value, the Euclidean norm of what appears to be the final error level will be $\|\underline{e}\| = |\varepsilon(p)| = \left\|\bar{u}_p^T \underline{e}_0\right\|$, i.e. the initial error in the part of the space that did not learn yet. Even though this is only one component of the error out of p components, where p is the number of time steps in the desired trajectory, this error is not necessarily particularly small. This behavior can be disappointing when one knows there is a rigorous mathematical proof that the error converges to zero.

An additional irony is that perhaps in real hardware, there is no internal instability, that one would not observe the instability even if one could perform 10^{100} iterations. The corrective update in Eq. (2-26) on any given iteration can be extremely small. If it is so small that the update is beyond the last digit in the digital to analog and analog to digital converters in the hardware, then no accumulation of corrective signal is possible as the iterations progress. The control action in this part of the space is then never updated. The finite word length of the converters has stabilized the internal instability.

These comments indicate that in practice, the application of ILC laws are mathematically very often subject to this internal instability, but that one never observes the phenomenon. And what one does observe is that the error in all other parts of the space goes to zero, and one is left with the error one had initially in the part of the space associated with the small singular value. This error is

$$\varepsilon_0(p) = \bar{u}_p^T \underline{e}_0 \tag{2-29}$$

The ILC user considers this the final error level achieved and may wonder why it is not closer to zero. In simulations, when other parts of the space go to a numerical zero like 10^{-15}, this slowly learning component of the error can make the final error level appear to stop improving at numbers such as 10^{-4} as shown in Fig. 2-10.

2.6 Conclusions

This section studies the instability of the ILC inverse problem. It is shown that if the system has non-minimum phase zeros, ILC tries to converge to an inverse solution that is unstable. The control action and intersample error both grow exponentially. We study this phenomenon from two perspectives.

First, we look at the poles and zeros. When inverting the system, the NMP zeros become unstable poles. Given a desired output, one can find the desired input by solving the difference equation. However, because there are unstable poles in the inverse system, the homogeneous solution has a growing term(s) that will not decay with time. We then examine the singular values and singular vectors of the system matrix *P*. It is shown that the inverse of this matrix is ill-conditioned. There are some anomalous singular values and singular vectors that do not possess the frequency response information of the system.

In practice, one may not see the growing control action. It is very likely that when ILC is applied, the error decays very fast in a few iterations and then stops as if the learning has been finished. The reason why the instability of the non-minimum phase zeros cannot be observed is due to the fact that: 1) The error update at each iteration is so small that many more iterations are required before the error component on the output singular vector associated with the bad singular value will become small. 2) For digital systems with D/A and A/D converters, the error updates can be so small that the update may be beyond the last digits of these converters, and so the corrective action for the error in this part of the space cannot accumulate.

Thus, the aim here is to not only eliminate the internal instability, so that the intersample error does not have an exponential growth, but also to succeed in making the error in this part of the space go to zero. In later chapters, we will introduce a new design that can achieve this goal.

2.7 References

[1] M. J. Blachuta, "On zeros of sampled systems," *American Control Conference*, Albuquerque, NM, 1997, Vol. 5, pp. 3205-3209.

[2] K. Åström, P. Hagander, and J. Strenby, "Zeros of Sampled Systems," *Automatica*, Vol. 20, Issue 1, 1984, pp. 31-38.

[3] K. Chen and R. W. Longman, "Creating Short Time Equivalents of Frequency Cutoff for Robustness in Learning Control," *Advances in the Astronautical Sciences*, Vol. 114, 2003, pp. 95-114.

[4] Y. Li and R. W. Longman, "Addressing Problems of Instability in Intersample Error in Iterative Learning Control," *Advances in the Astronautical Sciences*, Vol. 129, 2008, pp. 1571-1591.

[5] Y. Li and R. W. Longman, "Using Underspecification to Eliminate the Usual Instability of Digital System Inverse Models," *Advances in the Astronautical Sciences*, Vol. 135, 2010, pp. 127-146.

[6] Y. Li and R. W. Longman, "Characterizing and Addressing the Instability of the Control Action in Iterative Learning Control," *Advances in the Astronautical Sciences*, Vol. 136, 2010, pp. 1967-1985.

[7] H. S. Jang and R. W. Longman, "A New Learning Control Law with Monotonic Decay of the Tracking Error Norm," *Proceedings of the Thirty-Second Annual Allerton Conference on Communication, Control and Computing,* Monticello, IL, September, 1994, pp. 314-323.

[8] H. S. Jang and R. W. Longman, "Design of Digital Learning Controllers Using a Partial Isometry," *Advances in the Astronautical Sciences*, Vol. 93, 1996, pp. 137-152.

[9] D.H. Owens and N. Amann, "Norm-Optimal Iterative Learning Control," *Internal Report Series of the Centre for Systems and Control Engineering*, Unviersity of Exeter, 1994.

Chapter 3

Eliminate the Internal Instability in ILC Due to Sampling Zeros Using Multiple Zero Order Holds

3.1 Introduction

The original mathematical problem statement asks to produce zero error at every time step by adjusting the zero order hold input each time step. But eventually we discover that this solution does not address the intended physical problem that caused one to ask for zero tracking error at sample times. First, the solution asks for control action whose magnitude grows exponentially with time step and in rather few time steps one reaches actuator limits and cannot be implemented in hardware. If the NMP zeros are on the negative real axis, the control action alternates in sign each time step, so that at each sample time, the error can be zero, but the exponential growth of control action produces peak errors between time steps that grow exponentially. Hence, the second issue is, even if it were possible to implement the control producing zero error at sample times, the error between sample times grows exponentially. That does not address the original physical objective of obtaining little or no tracking error.

Reference 1 suggests that using an interpolating hold might solve the problem, and it does for a small class of problems, but it does not solve the problem in general. Reference 2 compares the phase-lead ILC with D-type and P-type ILC for NMP systems. It suggests that phase-lead ILC outperforms D-type and P-type ILC, and performance can be maximized by using a carefully selected filter to eliminate the influence of unstable frequencies. Reference 3 uses the phase-lead based reference shift algorithm that can shift the reference from iteration to iteration. The

experimental results demonstrate that this approach can achieve faster convergence speed and smaller final error level for a finite number of iterations.

Reference 4 models the influence of NMP zeros in the application of Norm Optimal ILC. The slow learning phenomenon is explained by introducing a model which predicts the error level at which it seems to stop improving. It suggests this error level can be small if the initial error is small in norm or small in some time interval.

Stable inversion presented by Paden in Ref. 5, was implemented to non-linear NMP systems, aiming to produce stable non-causal inverse mapping between command and output. Many researchers try to use the stable inversion in the context of ILC. Reference 6 studies the relationship of adjoint-type ILC and stable inversion. Reference 7 designs a new inversion-based algorithm which works for both minimum and non-minimum phase systems with gain and time-constant uncertainty.

This chapter studies a different approach. Instead of using a single zero order hold throughout each time step, we allow the zero order control action to be updated multiple times within each time step. This makes a kind of generalized hold, and it implies that there is no unique answer to the inverse problem. We wish to examine to what extent this can be used to eliminate instability of the inverse model.

We also consider using a cost function that asks for zero error at each time step, and simultaneously minimizes a quadratic function of the tracking error and the control action for each of the introduced intermediate steps. If no limit is placed on the control action it will just be the same unstable inverse solution at the fast sample rate. But if the penalty is chosen well, the exponential growth of magnitude of the control is eliminated, and one can also aim for improved tracking at the intermediate points.

Here we focus on applying the multiple zero order holds approach in eliminating the internal instability due to sampling zeros (NMP zeros introduced during discretization). In the next Chapter, we will extend its use to NMP system with intrinsic zeros.

3.2 Multiple Zero Order Hold ILC

When a normal zero order hold is applied, there is only one scalar quantity at each time step that can be used to adjust the output at the end of the time step. Given any initial run input-output pair $\underline{u}_0, \underline{y}_0$ with output error $\underline{e}_0$, the needed change in the input to produce the desired output is unique and given by $\delta \underline{u} = -P^{-1}\delta \underline{e} = P^{-1}\underline{e}_0$. And this solution has the ill-conditioning as discussed before. The concept that we want to investigate is to allow more zero order hold values between the time steps for which one asks for zero error. For example, in the numerical results below we ask for zero error every time step running at 50Hz sample rate, but we update the control action at 200Hz sample rate, thus allowing four zero order hold values to be used to adjust the error at the next 50Hz sample time. One can think of this as a form of generalized hold. For purposes of illustration below, consider that the trajectory is one second long, or 200 time steps at 200Hz. The question is whether having this extra freedom can bypass the ill-conditioning problem.

In this section we investigate the ILC problem where we aim to converge to zero error for every time step associated with the slower sample rate, we call these the addressed time steps. And for the fast time steps between these addressed steps we ask that the error be the result of a compromise between zero error and the size of the control action. This allows us to try to get small error at these points while the penalty on the size of the control action can prevent exponential growth of the control action magnitudes. The hope is:

(1) To converge to zero error at the slow steps.

(2) This includes converging to zero error at a reasonable rate for what used to be the part of the space that took astronomically long to learn.

(3) At the same time we want to eliminate the internal instability in the control action.

(4) And we want to get relatively small error between the addressed time steps.

To generate the mathematics needed for this ILC law, we rewrite Eq. (2-7) in Chapter 2 in the following form

$$\begin{bmatrix} \delta_j \underline{y}_a \\ \delta_j \underline{y}_n \end{bmatrix} = \begin{bmatrix} P_a \\ P_n \end{bmatrix} \delta_j \underline{u} = - \begin{bmatrix} \delta_j \underline{e}_a \\ \delta_j \underline{e}_n \end{bmatrix} \tag{3-1}$$

This starts with the *P* matrix including all time steps. Using the example values given above, it is a 200 by 200 matrix, and the 4th time step, and every integer multiple of 4 time steps, is an addressed time step. These are collected in the output vector $\underline{y}_a$ with the corresponding output error $\underline{e}_a$. The associated rows of *P* are collected to form P_a. The time steps that are not addressed are collected to form the corresponding $\underline{y}_n$, $\underline{e}_n$, and P_n. For the example numbers used here, P_a is a 50 by 200 matrix, and P_n is a 150 by 200 matrix. Form the singular value decomposition $P_a = U_a [S_a \quad 0] V_a^T$, partition V_a^T into an upper part $V_{a,1}^T$ and lower part $V_{a,2}^T$ to match with the partitions of $[S_a \quad 0]$. Define $\underline{\mu}_j = V_a^T \underline{u}_j$, so that $\underline{u}_j = V_a \underline{\mu}_j$, and again partition $\underline{\mu}_j$ into upper part $\underline{\mu}_{1,j}$ and lower part $\underline{\mu}_{2,j}$. Then

$$\underline{u}_j = V_{a,1} \underline{\mu}_{1,j} + V_{a,2} \underline{\mu}_{2,j} \qquad \delta_j \underline{u} = V_{a,1} \delta_j \underline{\mu}_1 + V_{a,2} \delta_j \underline{\mu}_2 \tag{3-2}$$

$$\delta_j \underline{\mu}_1 = V_{a,1}{}^T \delta_j \underline{u} \;\; \delta_j \underline{\mu}_2 = V_{a,2}{}^T \delta_j \underline{u} \tag{3-3}$$

$$\delta_j \underline{y}_a = -\delta_j \underline{e}_a = P_a \delta_j \underline{u} = U_a [S_a \quad 0] \begin{bmatrix} V_{a,1}{}^T \\ V_{a,2}{}^T \end{bmatrix} \delta_j \underline{u} = U_a S_a \delta_j \underline{\mu}_1 \tag{3-4}$$

Note that the control space has been portioned into two parts. The part associated with $\underline{\mu}_1$ is the only part that influences the errors at the addressed points. At the same time it influences the errors at the unaddressed time steps. The part associated with $\underline{\mu}_2$ represents the remaining freedom. Anything done in this part of the space will not influence the addressed points, and we can independently pick it to influence the errors at the unaddressed points. Of course, we should not ask for zero error since this just produces the *P* inverse solution at the faster sample rate, so we can consider a compromise between error and size of control action for this part of the space.

Considering the third system in Eq. (2-15) presented in Chapter 2, Figure 3-1 gives the singular values of matrix P_a which is a matrix of dimension 50 by 200. The fact that the singular values are now well behaved means that the process of introducing 3 extra zero order hold values between addressed time steps solved the ill conditioning problem, and it is now reasonable to expect all parts of the error space for addressed points to go to zero with iterations. When zero error was requested for all time steps, there was a unique answer for the change in the input needed from the initial input, to produce the needed change in the output, and this formula is given in the text above Eq. (3-1), and uses P^{-1}. For the new situation the needed change in the input must satisfy the equation $P_a \delta \underline{u} = -\delta \underline{e}_a = \underline{e}_{a.0}$. There is an infinite number of solutions to this underspecified set of equations. One of the solutions is of course, the same solution as when all time steps were addressed, and this solution is not what we want. In addition there is for example, the Moore-Penrose pseudo inverse solution which finds that solution that minimizes the Euclidean norm of $\delta \underline{u}$

$$\delta \underline{u} = V_a \begin{bmatrix} S_a^{-1} \\ 0 \end{bmatrix} U_a^T \underline{e}_{a.0} \tag{3-5}$$

The fact that all the singular values are well-behaved indicates that this Eq. (3-5) solution will not have bad intersample error levels.

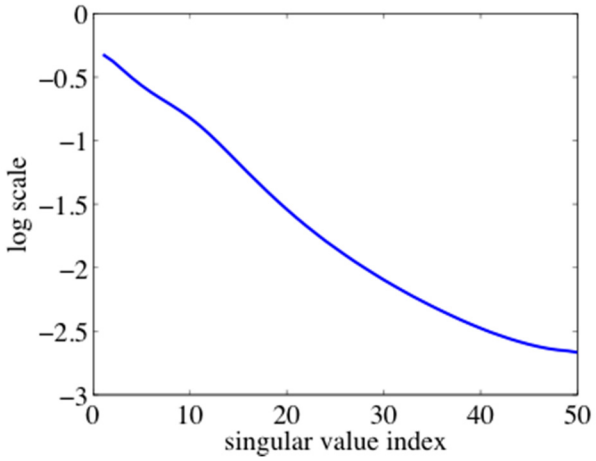

Figure 3-1. Singular values of 50 by 200 matrix P_a.

3.3 New Anomalous Singular Values and Singular Vectors in P_a

As discussed earlier in Chapter 2, using one zero order hold during the time interval between sample times at which one asks for zero error, produces the anomalous singular values that make the inverse problem being addressed not correspond to the physical problem of interest. Multiple zero order holds approach aims to eliminate such kind of anomalous singular values. However, our simulations suggest that some new anomalous singular values may appear. This section investigates the properties of this new type of anomalous singular values and singular vectors.

The numerical studies presented in this section use the systems in Eq. (2-14) in Chapter 2. We ask for zero error every time step running at 50 Hz sample rate, but will update the control action at 100 Hz sample rate, which is equivalent to using two zero order holds for each addressed time

step. We can start with the full 100 by 100 P matrix associated with 100 Hz sample rate, and associated with desired trajectories that are one second long. We now address only the even number rows in P, making a 50 by 100 matrix P_a as defined in previous section. S_a is a 50 by 50 diagonal square matrix that contain the singular values, U_a has the dimension 50 by 50. Each column of the matrix represents singular vectors associated with each of the singular values, while V_a has twice the size of U_a, the first 50 columns correspond to the singular values, and the rest are picked by MATLAB as orthonormal vectors that span the null space of P_a.

3.3.1 The Number of New Anomalous Singular Values in the Skip Step Formulation

Figures 3-2 and 3-3 plot the 50 singular values of the skip step P_a matrix. Also shown as a solid line is the frequency response of the discrete time system. The singular value plot follows the frequency response curve relatively closely, but again there are some anomalous singular values, and these are highlighted by circles in the plots. We describe them as new or extra anomalous singular values. Note that these singular values are not as small as the original anomalous singular values detailed in previous Chapter.

Pole excess of 3 has one singular value circled, but it is not obvious that it is anomalous. In a later section we establish that it is unrelated to frequency response, based on its singular vectors, and therefore it is anomalous. However, because this anomalous singular value is so close to the normal ones, it won't cause the difficulty as we discussed before. Thus, in previous section, we claim that all singular values in Fig. 3-1 are well-behaved.

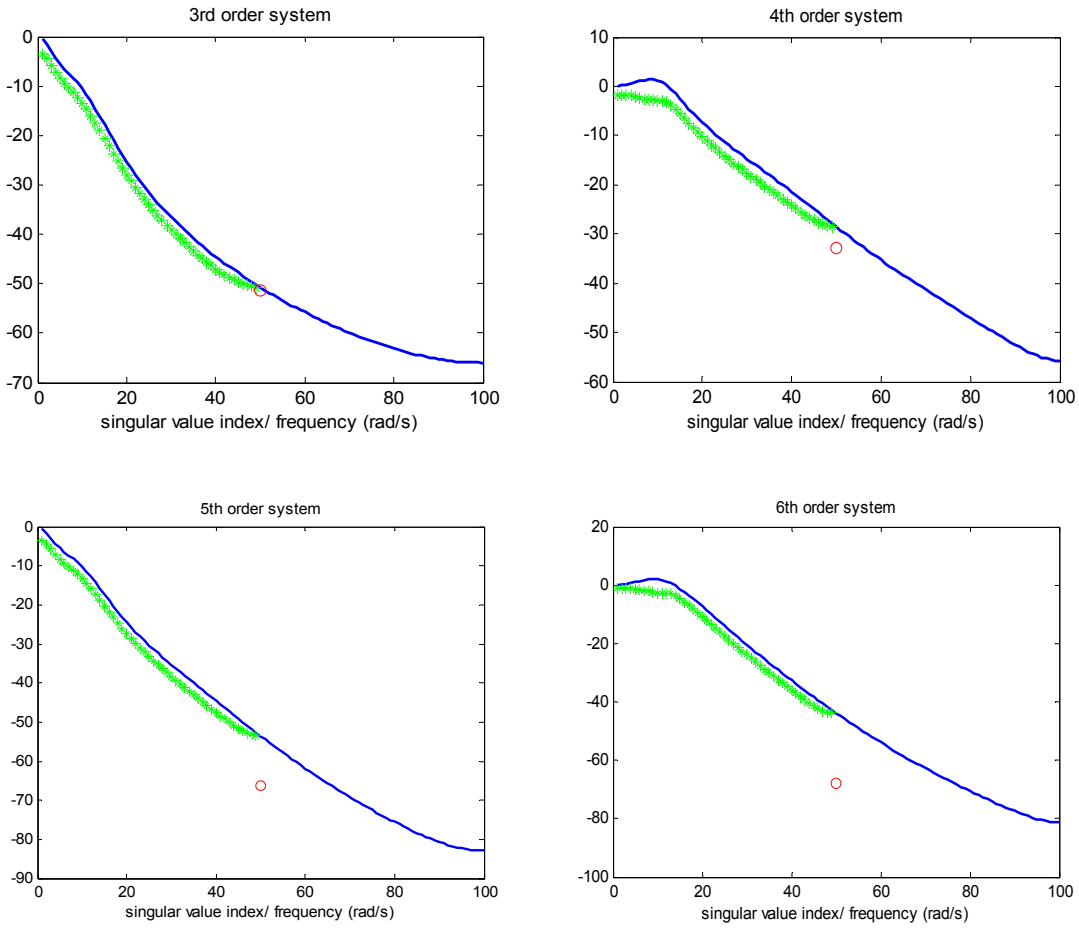

Figure 3-2. Singular Values of P_a and Magnitude Frequency Response, Pole Excesses 3,4,5,6.

Table 3-1 adds one more column to Table 2-2 in Chapter 2. It gives the number of original anomalous singular values of P as determined in the previous Chapter, and the last column addresses the anomalous singular values of matrix P_a. The original anomalous singular values are gone, and the number of new anomalous singular values is listed. This number exhibits a rather different pattern.

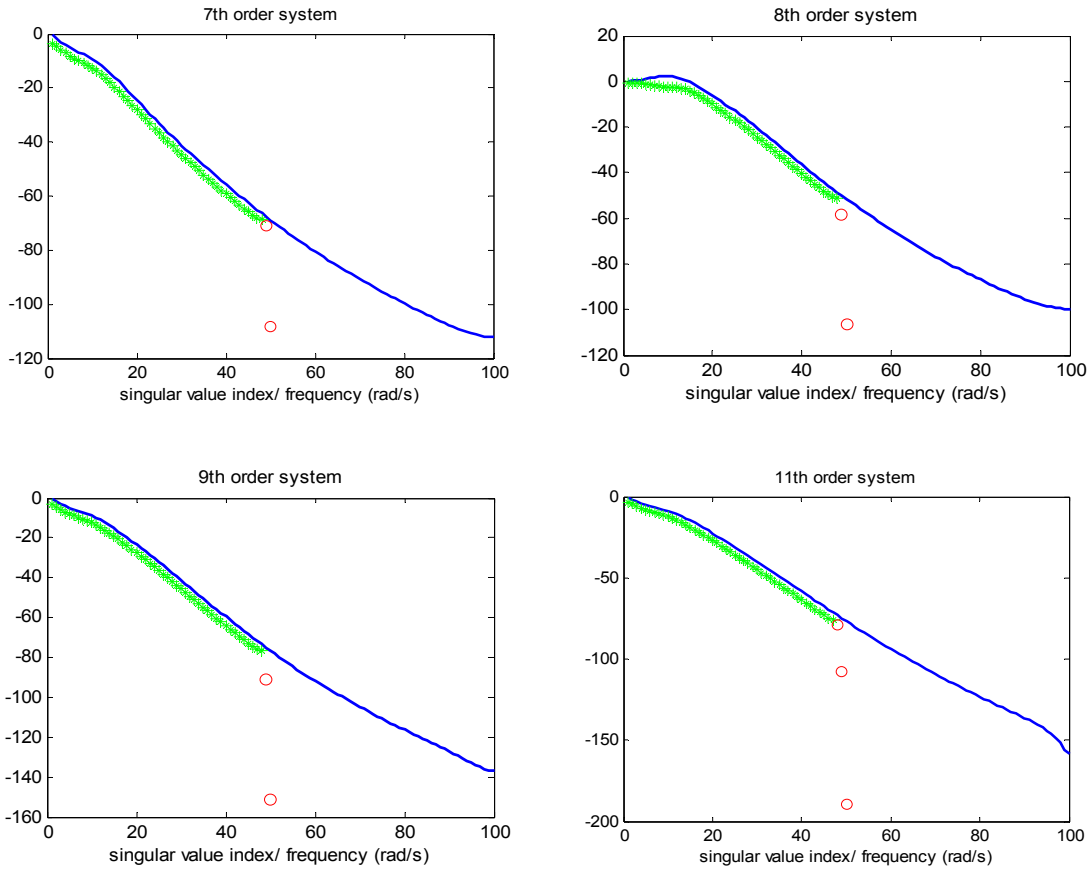

Figure 3-3. Singular Values of P_a and Magnitude Frequency Response, Pole Excesses 7, 8, 9, 11.

Table 3-1. Number of Particularly Small Singular Values

Pole excess	Number of zeros outside the unit circle	Number of anomalous singular values	Number of anomalous singular values(every other time step)
2	0	0	0
3	1	1	1
4	1	1	1
5	2	2	1
6	2	2	1
7	3	3	2
8	3	3	2
9	4	4	2
10	4	4	2
11	5	5	3

3.3.2 New Anomalous Singular Values as a Function of Matrix Size, Sampling Time, and Number of Skipped Steps

As discussed in previous chapter, the anomalous singular values for the full P matrix were seen to relate to the location of the NMP zeros, and got smaller by the factor (the reciprocal of the NMP zero) for every unit increase in matrix size. The top left plot of Figure 3-4 examines whether the new anomalous singular value in P_a has this same dependence, giving this singular value as a function of matrix size, $p/2$ by p, for the 3rd order system sampled at 100 Hz. As the matrix size increases this singular value drops quickly, and then by $p/2 = 10$ it essentially stops decaying and remains constant as the matrix size is increased further, becoming independent of matrix size. Clearly the new anomalous singular value is not related to the zero location, and does not challenge the computational ability of the singular value decomposition algorithm.

The top right plot of Figure 3-4 holds the matrix size as 50 by 100, but changes the sample time interval T. The smallest singular value is seen to decrease as the sample time T is decreased. Comparing to Figure 2-4, the smallest singular value for the full P matrix was related to the zero location which was -2.9028, and which relatively quickly saturates as the zero location approaches its asymptotic value of -3.7321 as T gets small. Again, the new anomalous singular value seems to not be related to the zero location. The bottom curve in Figure 3-4 shows the smallest singular value as a function of the number of time steps that are skipped between addressed time steps. The plot keeps the time interval between addressed time steps constant, i.e. the slow sample rate is fixed at 0.02 s, the number of addressed time steps is held constant at 50, and the fast sample rate it increased to allow for 1, 2, 3, etc. zero order hold values between these addressed steps. So the P_a matrix is 50 by 100, 150, …, 500 for the number of skipped steps going from 2 to 10. Interestingly, we observe that the problematic smallest singular value of the full P matrix becomes

a much more reasonable singular value as soon as there are two zero order holds between the fixed addressed time steps, but when more holds are introduced no further improvement appears in the new anomalous singular value.

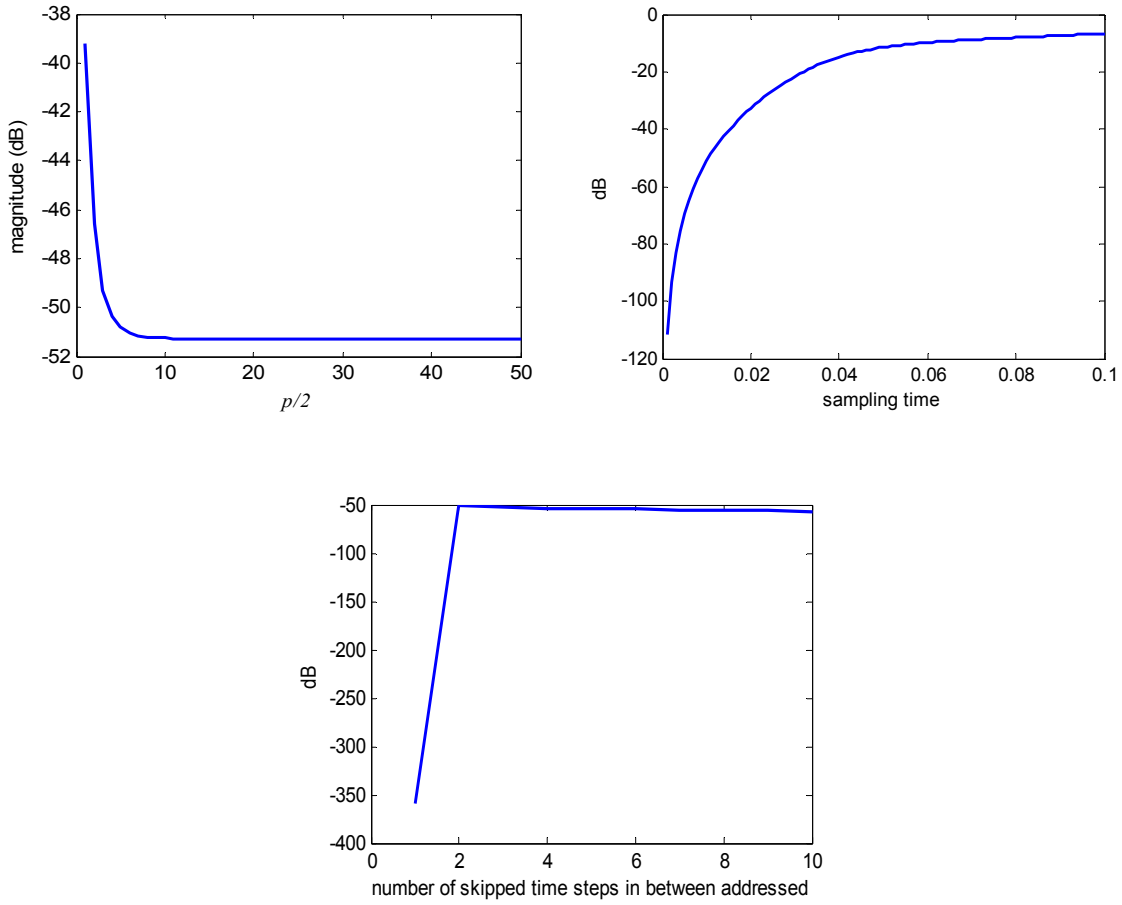

Figure 3-4. Smallest singular value of P_a matrix vs. matrix output size, sampling time interval and number of skipped time steps between addressed time steps (pole excess of 3).

3.3.3 The New Anomalous Input and Output Singular Vectors

We now examine the behavior of the singular vectors associated with the new anomalous singular values of P_a. Figure 3-5 considers the usual 3rd order system with the addressed time steps corresponding to 0.02 s sample time interval, and P_a of dimension 50 by 100. The left plot gives the 50 dimensional output singular vector, component magnitude at each time step, while the right

plot is the corresponding 100 dimensional input singular vector component magnitudes. Figures 2-5 and 2-6 for the full *P* matrix had exponentially decaying magnitudes with time step of the output singular vectors, and exponentially growing magnitudes of the input singular vector components with time step. Some understanding of this comes form the realization that the control action that eliminates an error near the beginning of a trajectory produces an unstable control action that keeps growing to the end of the trajectory. The new anomalous singular vectors have different behavior, both input and output singular vectors have component magnitudes that decay exponentially with time step.

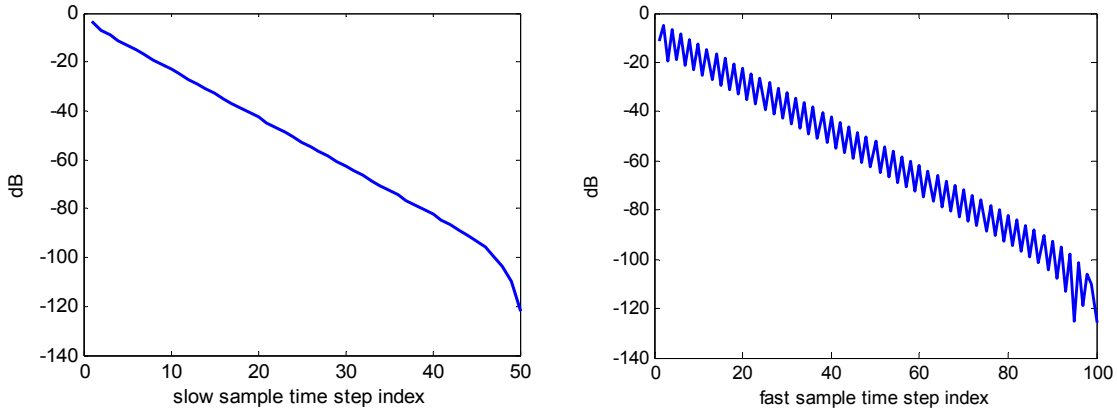

Figure 3-5. Magnitudes of Last Output and Input Singular Vector Components (pole excess of 3)

The zig-zag of the input singular vector must relate to the fact that every other time step is addressed. Figure 3-6 presents both the left and the right singular vector component magnitudes on the same plot, but eliminates every other entry on the input singular vector. The solid line and the circles which are on top of the line show that these two plots are essentially identical, and the decay with time step is much slower than with the singular vector for the full *P* matrix. In the limit as the number of time steps tends to infinity, we know that the singular values and singular vectors tend to steady state frequency response. But for finite time trajectories we already know that there

can be singular values related to unstable zero locations that preclude reaching a steady state in control action. Of course, finite time trajectories also have transients related to the poles of the transfer function, and the associated solutions decay with time for an asymptotically stable system. The finite time input-output relationship represented by *P* must somehow handle these transients at the beginning of the trajectory, and for short enough trajectories the transients must become an important part of the response. To examine possible connection between the singular vectors and the zero location, and the locations of the three poles, the plot also shows the slope for the zero, the slope for the real pole, and the slope for the decay of the complex pair of poles. None of these slopes seems to relate to the slope observed for these new singular vectors.

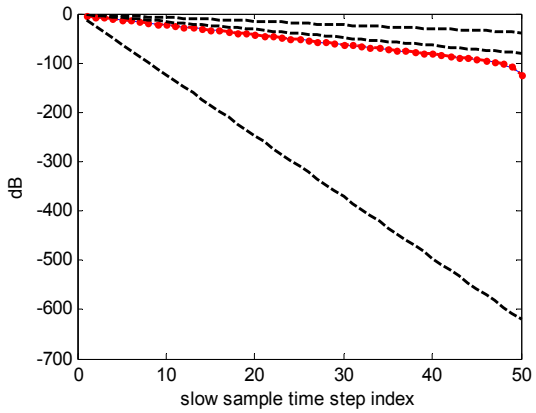

Figure 3-6. Comparison of the Magnitudes of Last Output Singular Vector Components and Every Other Point of Last Input Singular Vector (pole excess of 3). Three Decaying Slopes are Given for $(0.9158)^k$ and $(0.8311)^k$ (System Poles) and $(1/3.31)^k$ (Zero).

Figures 3-7 and Figure 3-8 examine how the slope of the output singular vector magnitude plot changes with different factors. For the 7th order system we consider 100 Hz sample rate, and to facilitate comparison to the full *P* case presented in Figure 2-6, the P_a matrix is made 100 by 200. According to Table 3-1, there are two zeros outside the unit circle. The left plot in Figure 3-7 gives the two singular vector component magnitudes associated with the two anomalous singular values;

compare this plot with the top left plot of Figure 2-6. This time we can see two different decay rates even when the dimension is increased. The right plot of Figure 3-7 shows the singular vector associated with the smallest singular value, four lines with different lengths represent the singular vector with different size of P_a: 10 by 20, 20 by 40, 50 by 100, and 100 by 200. We can see that the slope does not change with time length. Unlike the original singular vector associated with smallest singular values for the full P matrix, the computed slope of the new smallest one only follows one decay rate.

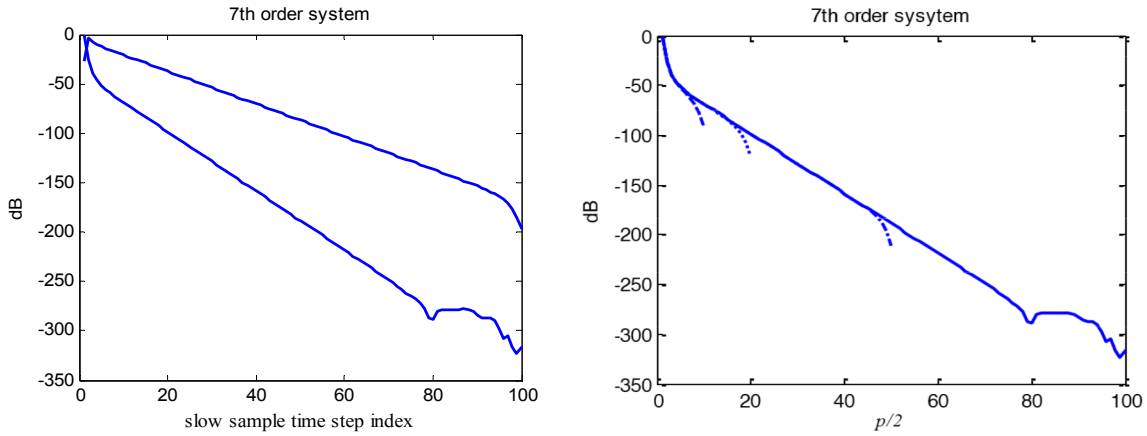

Figure 3-7. Magnitudes of Last Two Output Singular Vector Components and Magnitudes of the Last Output Singular Vector Components of Different Time Length (0.1 sec, 0.2 sec, 0.5 sec, 1 sec)

Figure 3-8 considers the 3rd order system to further examine the properties of such singular vectors. The left plot uses fixed slow sampling rate of 50 Hz and number for addressed time steps equal to 50 so that the trajectory of interest is one second long. The P_a matrix is 50 by 100, 150, 200, and 400 for n = 2, 3, 4, and 8 respectively. One observes that if one skips more time steps, the slope of the output singular vector associated with the particularly small singular value becomes smaller. And the right plot considers a 50 by 100 P_a matrix using different sampling rates, skipping one time step in between addressed ones. It demonstrates that for slower sampling, the

slope is smaller. This property suggests that by skipping more time steps between addressed steps, and sampling slower can both diminish detrimental influence of the new particularly small singular values. Note that in the limit that the sample time updates are made so infrequently that the system reaches steady state response before the next update, then only the DC gain matters in the updates and dynamics is not an issue.

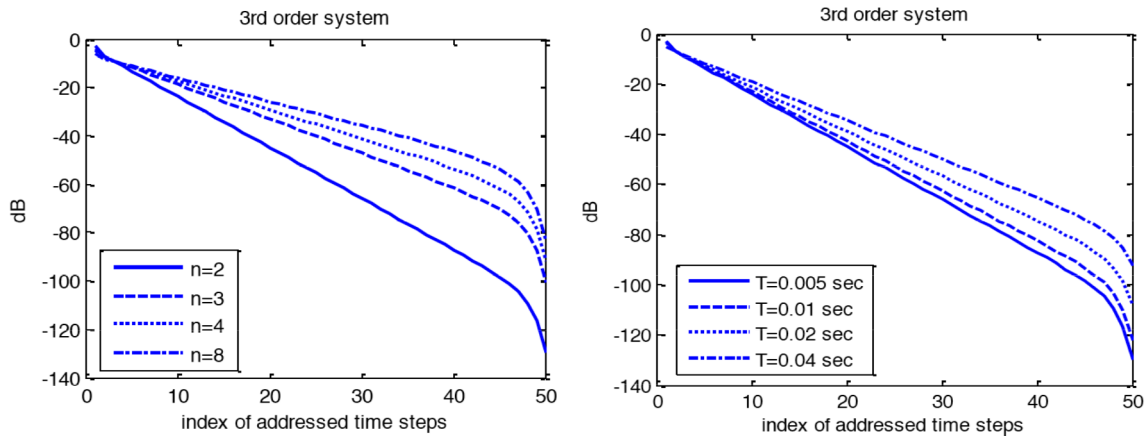

Figure 3-8. Magnitudes of Last Output Singular Vector Components vs Number of Time Steps Being Skipped (n), and the Sample Time Interval (pole excess of 3)

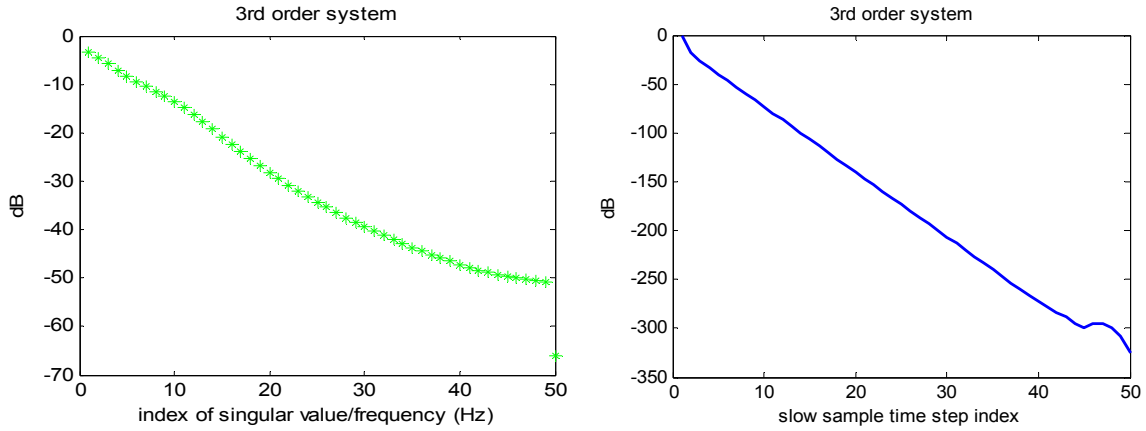

Figure 3-9. Magnitude Response vs Singular Values and Magnitudes of Last Output Vector Components for Addressing Time Steps Starting with the First Step (pole excess of 3)

All of the computations made above that address every other time step, address the error at even numbered time steps, starting with the second time step of the output. Figure 3-9 treats the 3rd order

system, and considers the alternative of addressing odd numbered time steps, starting with the first time step. We observe that his choice is not advisable. The left plot in Figure 3-9 gives the singular values to be compared to the first plot in Figure 3-3, and we see that this time the anomalous singular value is very evident, whereas for Figure 3-3 we needed to look at the associated singular vector to know that it was anomalous. The right plot of Figure 3-9 should be compared to the left plot of Figure 3-5, and we observe the decay rate of the singular vector component magnitudes is much faster in Figure 3-9. The extra time step at the beginning before the first addressed step has very beneficial effects.

3.4 Eliminating All Anomalous Singular Values (Modified Version of Multiple Zero Order Hold Approach)

The new anomalous singular values do not always appear. In a later Chapter, we will see that if the NMP zero gets closer to the unit circle when sampling slower, using two zero order holds between each sample time can eliminate the original bad singular value and no new anomalous singular value is observed. Nevertheless, we would like to have these new anomalous singular values eliminated as well. According to Table 3-1 when doing skip step, addressing every other sample time starting with the second step, pole excesses of 3, 4, 5, 6 have one anomalous singular value, although in the case of excess 3, one must look at the associated singular vector to see that it is anomalous, i.e. corresponding to a decaying signal instead of relating to a frequency. Pole excesses of 7, 8, 9, 10 have two anomalous singular values, and pole excess of 11 has 3 anomalous singular values. We show here that if one skips a number of time steps (at the slow sample rate) at the beginning of the trajectory equal to the number of new anomalous singular values, then the resulting P_a matrix no longer has any anomalous singular values.

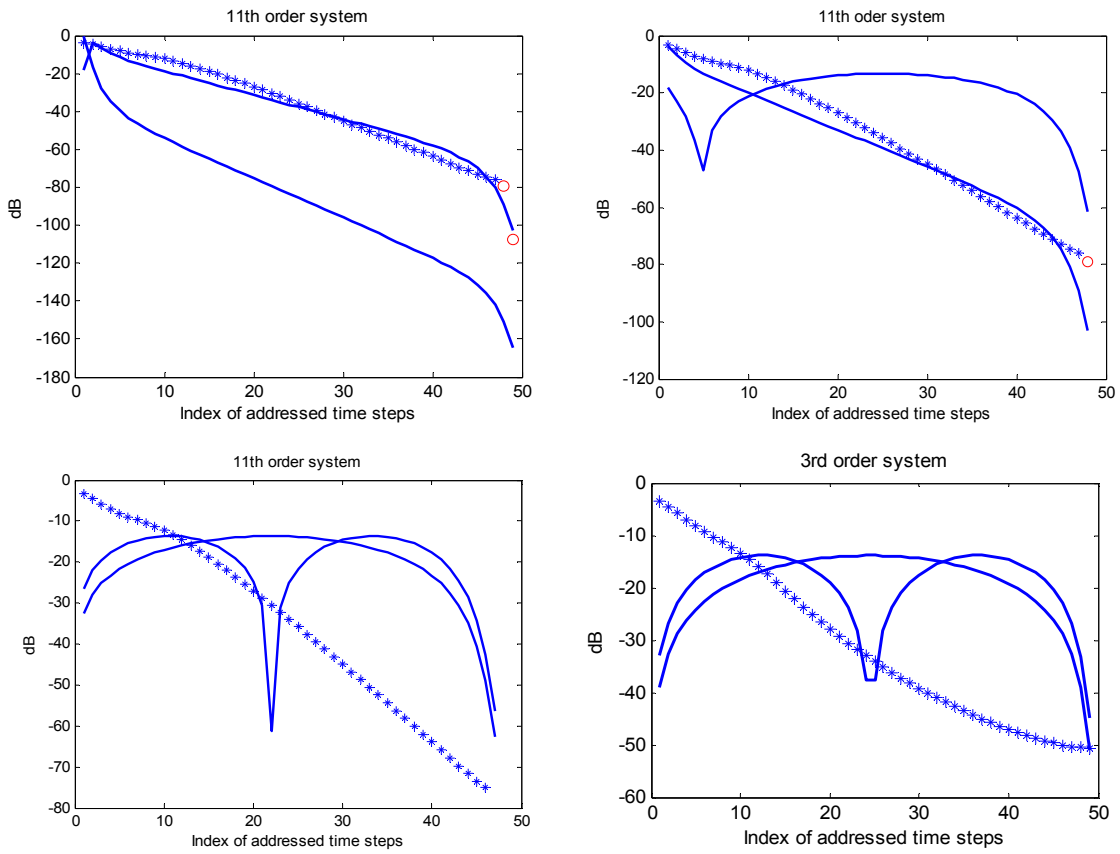

Figure 3-10. Singular Values and Last Two Output Singular Vectors of 11th Order and 3rd Order Systems When Leaving Out One, Two and Three Initial Time Step(s) After Skipping

Figure 3-10 illustrates this for the 11th order system. The top left plot shows the decay of the singular values, and uses circles to highlight the two anomalous singular values that result if we delete just the first row of the original skip step P_a, i.e. starting with the row associated with fast time step 4, slow time step 2. On the same plot is a plot of the magnitudes of the components of the last two output singular vectors. The fact that they are decaying also tells us that they are both anomalous and not related to frequency response. The top right plot, eliminates one more row of the original P_a, so that its first row applies to slow time step 3, or fast time step 6. Note that having deleted one more initial row has eliminated one of the two anomalous singular values. The bottom left plot deletes the first 3 rows of the original P_a, this number of rows equals the number of

anomalous singular values indicated on Table 3-1 for an 11th order system. And the singular vector plots show that this has eliminated all anomalous singular values and singular vectors. The bottom right of Figure 3-10 examines the 3rd order model which has one anomalous singular value. We observe that eliminating one row of the original P_a produces a matrix that has no anomalous singular value. When looking at the singular vector component plots for non decaying vectors, keep in mind that the plot is of the absolute value of the component and on a dB scale, the actual components exhibit high frequency behavior.

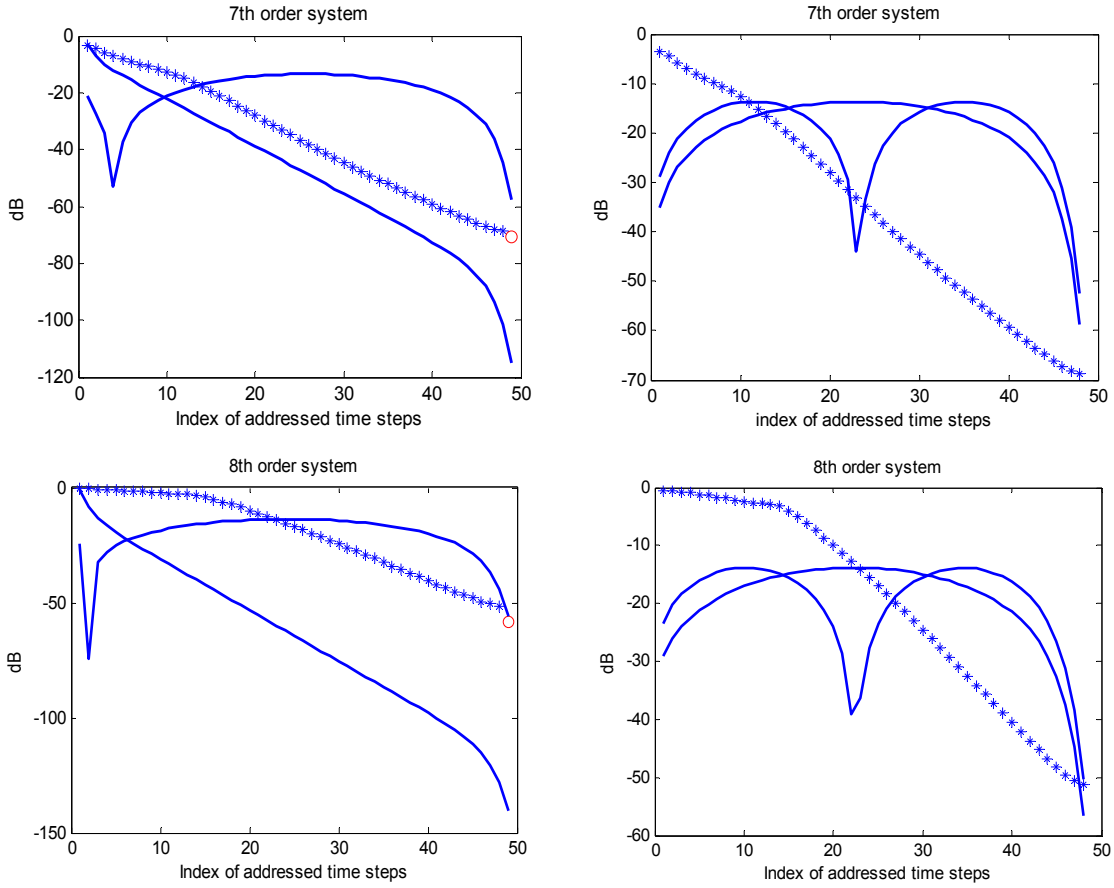

Figure 3-11. Singular Values and Last Two Output Singular Vectors of 7th order (top) and 8th Order System (bottom) When Leaving Out One and Two Initial Time Step(s) After Skipping

Figure 3-11 examines both the 7th order and 8th order systems. According to Table 3-1 they each have two new anomalous singular values. The plots show that eliminating the first row of the original P_a results in having only one anomalous singular value and vector, while eliminating the first two rows eliminates all anomalous singular values and vectors.

3.5 Quadratic Cost Learning Law Design Using Multiple Zero Order Holds

3.5.1 Two New QCL Design

Follow the mathematics in Section 3.2, we now consider two control laws. One simply addresses the $\underline{\mu}_1$ part of the control space (Control Law 1), and the second (Control Law 2), in addition performs a quadratic cost compromise in the not addressed part of the space.

Control Law 1. We design an iterative learning control law to converge to zero error at the addressed points using $\delta_j \underline{\mu}_1$ only, picking it to minimize the quadratic cost

$$J_{a,j} = \underline{e}_{a,j}{}^T \underline{e}_{a,j} + r_a \delta_j \underline{\mu}_1^T \delta_j \underline{\mu}_1 \tag{3-6}$$

Note that $\delta_j \underline{\mu}_2$ does not influence $\underline{e}_{a,j}$, so that $\underline{e}_{a,j} = \underline{e}_{a,j-1} + \delta_j \underline{e}_a = \underline{e}_{a,j-1} + P_a V_{a,1} \delta_j \underline{\mu}_1$. Using this in Eq. (3-6) and computing $dJ_{a,j} / \delta_j \underline{\mu}_1 = 0$ produces the control update law

$$\delta_j \underline{\mu}_1 = (V_{a,1}{}^T P_a^T P_a V_{a,1} + r_a I)^{-1} V_{a,1}{}^T P_a^T \underline{e}_{a,j-1} \tag{3-7}$$

If our only interest is to obtain zero tracking error at the addressed time steps, we could use this result in Eq. (3-2), and leave $\underline{\mu}_2$ unchanged. This produces the following ILC law

$$\delta_j \underline{u} = V_{a,1} (V_{a,1}{}^T P_a^T P_a V_{a,1} + r_a I)^{-1} V_{a,1}{}^T P_a^T \underline{e}_{a,j-1} \tag{3-8}$$

We will see that this law using extra zero order hold steps can address the problem associated with the ill-conditioning of the ILC problem.

Proof of Convergence. Note that $\delta_j \underline{e}_a = -P_a \delta_j \underline{u} = U_a S_a V_{a_1}{}^T \delta_j \underline{u} = U_a S_a \delta_j \underline{\mu}_1$. Write Eq. (3-7) in the form $\delta_j \underline{\mu}_1 = L_{\mu 1} \underline{e}_{a,j-1}$. Then, $\underline{e}_{a,j} = [I - U_a S_a L_{\mu 1}] \underline{e}_{a,j-1}$. The $L_{\mu 1}$ contains the product $V_{a,1}{}^T P_a^T$ which equals $S_a U_a^T$. And the product $P_a V_{a,1}$ in $L_{\mu 1}$ is equal to $U_a S_a$. Note that the Euclidean norm of the error $\underline{e}_{a,j}$ is the same as that of $U_a^T \underline{e}_{a,j}$. Then the error propagation from iteration to iteration can be written as

$$(U_a^T \underline{e}_{a,j+1}) = [I - S_a (S_a^2 + r_a I)^{-1} S_a](U_a^T \underline{e}_{a,j}) = diag[r_a / (r_a + \sigma_1^2), \ldots, r_a / (r_a + \sigma_{p_a}^2)](U_a^T \underline{e}_{a,j}) \quad (3\text{-}9)$$

where p_a is the number of addressed time steps, equal to the number of singular values of P_a. Because each element of the diagonal matrix is positive and less than one in magnitude, all error components decay to zero as the time steps progress. We will see that the smallest singular value can have a reasonable value for this matrix. This proves that the error at the addressed time steps converges to zero. A later section will create a formula for the final error associated with the not addressed time steps.

Control Law 2. Numerical results indicate that the problem of internal instability of the control action in time steps is addressed by introducing the extra zero order hold values. We may also want to make use of the extra freedom available, in order to decrease tracking error between the addressed steps. We know that we cannot ask for zero tracking error because that would again produce the bad singular value difficulty. But we can decide to try to make the tracking error small, but do so as a compromise between control effort and tracking error. By penalizing control effort

we can prevent the control action from exponentially growing which is the source of the bad error levels between time steps.

Previously, the updated control action $\delta_j \underline{u}$ on the right of Eq. (3-2) only contained the $\delta_j \underline{\mu}_1$ update term, and the $\delta_j \underline{\mu}_2$ term was zero. We will create an update for this latter term. Note that now $\delta_j \underline{e}_a = -P_a \delta_j \underline{u} = U_a S_a V_{a_1}{}^T (V_{a,1} \delta_j \underline{\mu}_1 + V_{a,2} \delta_j \underline{\mu}_2) = U_a S_a \delta_j \underline{\mu}_1$. Because of the orthogonality of the columns of $V_{a,1}$ and $V_{a,2}$, anything we choose to do to $\underline{\mu}_2$ will not influence the convergence of the error $\underline{e}_{a,j}$ to zero as *j* goes to infinity.

Then for the $\underline{e}_n$ part of the error we create a learning rule using $\underline{\mu}_2$

$$J_{n,j} = \underline{e}_{n,2}^T \underline{e}_{n,2} + r_{n,1} \underline{\mu}_{2,j}^T \underline{\mu}_{2,j} + r_{n,2} \delta_j \underline{\mu}_2^T \delta_j \underline{\mu}_2 \tag{3-10}$$

This time $\underline{e}_{n,j} = \underline{e}_{n,j-1} + \delta_j \underline{e}_n = \underline{e}_{n,j-1} - P_n \delta_j \underline{u}$. Also, $\underline{\mu}_{2,j} = \underline{\mu}_{2,j-1} + \delta_j \underline{\mu}_2$. The needed control update computed from $dJ_{n,j} / \delta_j \underline{\mu}_2 = 0$ is

$$\delta_j \underline{\mu}_2 = (V_{a,2}^T P_n^T P_n V_{a,2} + r_{n,1} I + r_{n,2} I)^{-1} V_{a,2}^T [P_n^T \underline{e}_{n,j-1} - P_n^T P_n V_{a,1} \delta_j \underline{\mu}_1 - r_{n,1} \underline{u}_{j-1}] \tag{3-11}$$

Then substituting Eq. (3-7) produces the ILC law for the unaddressed time steps in terms of the overall control action $\underline{u}_{j-1}$ in the previous run, and both parts, $\underline{e}_{n,j-1}$ and $\underline{e}_{a,j-1}$ of the previous run error

$$\begin{aligned} \delta_j \underline{\mu}_2 = (V_{a,2}^T P_n^T P_n V_{a,2} + r_{n,1} I + r_{n,2} I)^{-1} [V_{a,2}^T P_n^T \underline{e}_{n,j-1} \\ - V_{a,2}^T P_n^T P_n V_{a,1} (V_{a,1}^T P_a^T P_a V_{a,1} + r_a I)^{-1} V_{a,1}^T P_a^T \underline{e}_{a,j-1} - r_{n,1} V_{a,2}^T \underline{u}_{j-1}] \end{aligned} \tag{3-12}$$

Substituing Eq. (3-7) and Eq. (3-12) or Eq. (3-11) into Eq. (3-2) produces the full ILC Control Law 2

$$\underline{u}_j = \underline{u}_{j-1} + V_{a,1}\delta_j \underline{\mu}_1 + V_{a,2}\delta_j \underline{\mu}_2 \tag{3-13}$$

3.5.2 Final Error Level at Unaddressed Time Steps

It is of interest to develop formulas that describe the final error levels for the unaddressed time steps, using Control Law 1 and using Control Law 2. For purposes of this section, we consider a difference operator δz that is the final value of variable z minus the value in run number zero. Whatever the input is in run number zero, $\underline{u}_0$, it produces an output $\underline{y}_0$ and error $\underline{e}_0$, from which one can compute $\underline{e}_{n,0}$ and $\underline{\mu}_{1,0}$ and $\underline{\mu}_{2,0}$.

For Control Law 1, $\delta \underline{e}_a = -P_a \delta \underline{u}$. The desired change in the error is $-\underline{e}_{a,0}$ to result in zero tracking error, so $\underline{e}_{a,0} = P_a \delta \underline{u} = P_a U_a S_a \delta \underline{\mu}_1$. From this we conclude that

$$\delta \underline{\mu}_1 = S_a^{-1} U_a^T \underline{e}_{a,0} \tag{3-14}$$

Similarly, $\delta \underline{e}_n = -P_n \delta \underline{u} = -P_n V_{a,1} \delta \underline{\mu}_1$ so that the final value of the error at the unaddressed points, call it $\underline{e}_{n,\infty}$, is given by

$$\underline{e}_{n,\infty} = \underline{e}_{n,0} - P_n V_{a,1} S_a^{-1} U_a^T \underline{e}_{a,0} \tag{3-15}$$

Control Law 2 will converge to the same $\delta \underline{\mu}_1$ as in Eq. (3-14), but we need to compute the $\delta \underline{\mu}_2$. When convergence has been reached, the cost function in Eq. (3-10) becomes independent of j, so the value of $\delta \underline{\mu}_2$ minimizes

$$J_n = \underline{e}_{n,\infty}^T \underline{e}_{n,\infty} + r_{n,1} \underline{\mu}_{2,\infty}^T \underline{\mu}_{2,\infty} \tag{3-16}$$

The dependence on $\delta \underline{\mu}_2$ is given by

$$\underline{\mu}_{2,\infty} = \underline{\mu}_{2,0} + \delta \underline{\mu}_2 \tag{3-17}$$

$$\begin{aligned} \underline{e}_{n,\infty} &= -P_n(\underline{u}_0 + \delta \underline{u}) + f_n \\ &= -P_n(\underline{u}_0 + V_{a,1}\delta \underline{\mu}_1 + V_{a,2}\delta \underline{\mu}_2) + f_n \\ &= [-P_n(\underline{u}_0 + V_{a,1}S_a^{-1}U_a^T \underline{e}_{a,0}) + f_n] + [-P_n V_{a,2}]\delta \underline{\mu}_2 \\ &= [\underline{e}_{n,0} - P_n V_{a,1}S_a^{-1}U_a^T \underline{e}_{a,0}] + [-P_n V_{a,2}]\delta \underline{\mu}_2 \\ &= \beta + \gamma \delta \underline{\mu}_2 \end{aligned} \tag{3-18}$$

where we have used $\underline{e}_{n,0} = -P_n \underline{u}_0 + f_n$ to eliminate f_n in terms of initial input and output history data. Substituting into Eq. (3-16) and minimizing produces the final $\delta \underline{\mu}_2$, and the final error history for the unaddressed points

$$\delta \underline{\mu}_2 = (\gamma^T \gamma + r_{n,1} I)^{-1}(\gamma^T \beta + r_{n,1} \underline{\mu}_{2,0}) \tag{3-19}$$

$$\underline{e}_{n,\infty} = \underline{e}_{n,0} - P_n V_{a.1}\delta \underline{\mu}_1 - P_n V_{a.2}\delta \underline{\mu}_2 \tag{3-20}$$

Comparing this to Eq. (3-15), what is new is just the last term in the final error expression. Note that γ is independent of the data from the initial run, but the initial error and initial control history is contained in $\underline{e}_{n,0}$, β, and $\underline{\mu}_{2,0}$.

3.5.3 Simulation Result

Figure 3-12 gives the error and the control action using the original quadratic cost ILC law as shown in Eq. (2-19) in Chapter 2 with the value of *r* set to 0.1. The results are those obtained after

10^4 iterations. Figure 3-13 shows the final error and the control action using Control Law 1. We see that there is no exponential growth of the control action, and we see that it does produce a numerically zero error at each addressed point as desired. This is its main advantage, and it can achieve this zero error even in the part of the space that took astronomically long to learn using ILC law Eq. (2-19). On the other hand, the intersample error levels are similar to the error levels in Figure 3-12. Figure 3-14 shows the results for Control Law 2 using a cost function weight of $r_{n,1} = 10^{-12}$. We see that amazingly, it is possible to get extremely close to zero error at both addressed and unaddressed time steps without any serious deterioration of the nature of the control input, i.e. without starting to see any exponential growth in the control action with time step, and the accompanying deterioration of intersample error between the fast sample times. We conclude that one can get very close to zero error at all time steps with well-behaved control action, thus solving for practical purposes the internal instability problem of the control action in ILC.

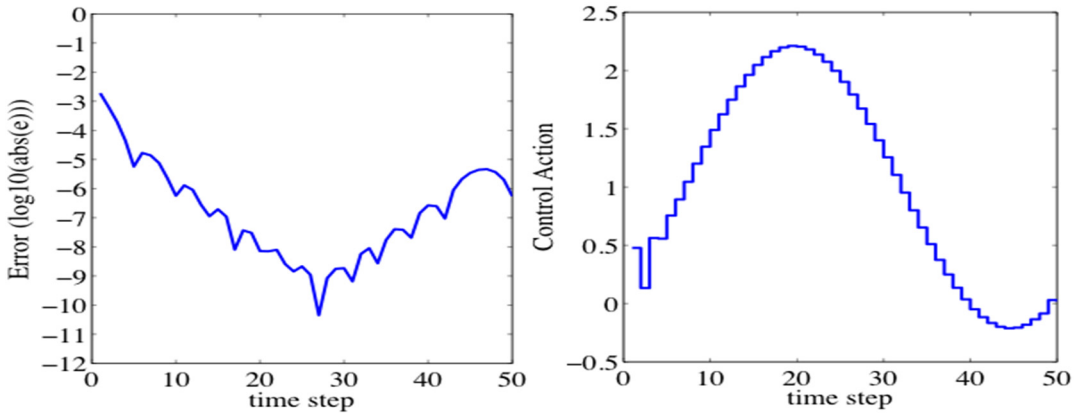

Figure 3-12. The error (sampled at 200Hz) vs. time step and the control action at iteration number 10^4 using one zero order hold between addressed time steps and ILC law Eq. (2-19) with $r = 0.1$.

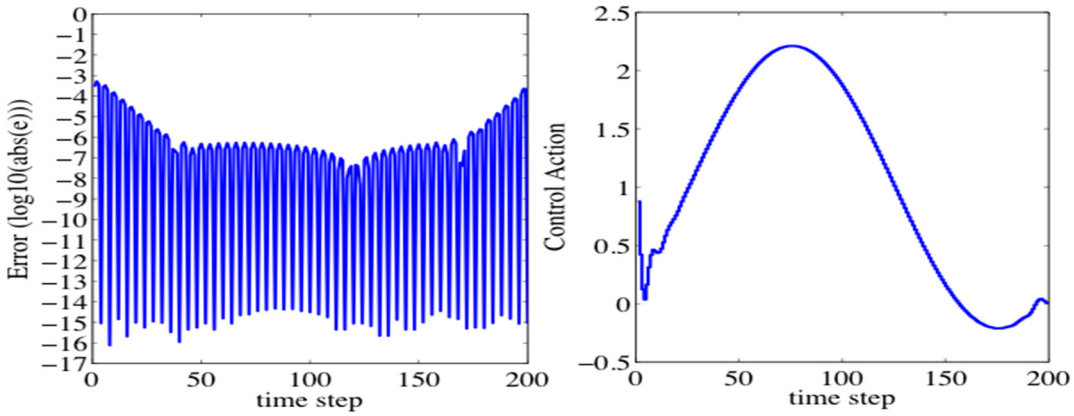

Figure 3-13. The final error (sampled at 200Hz) vs. time step and the control action on a logarithmic scale using Control Law 1 (50Hz sample rate for addressed time steps).

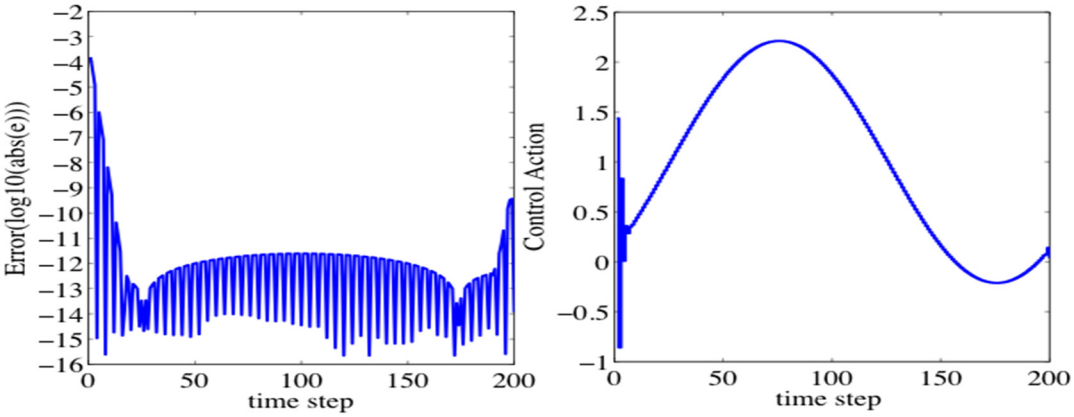

Figure 3-14. The final error (sampled at 200Hz) vs. time step and the control action on a logarithmic scale using Control Law 2 (50Hz addressed rate) with the weight $r_{n,1} = 10^{-12}$.

3.6 Initial Deletion Approach

Reference 8-10 suggest another effective approach which does not ask for zero tracking error at a number of initial time steps. The number of rows to be left free is equal to the number of non-minimum phase zeros, provided that the relative degree in discrete time is one. This approach was developed based on the thinking of initial conditions. As we discussed before, take the third order system in Eq. (2-15) for example, given desired output, desired input is equal to

$C_1(r_1)^k + C_2(r_2)^k + u_p$. Let's assume that r_1 is the non-minimum phase zero, and u_p is the particular solution associated with the desired output. C_1 and C_2 are constants which determined by the initial conditions. We believe that the extra freedom at the beginning gets the right start up in the sense that C_1 can be made zero so that the exponentially growing term disappear.

If not asking for zero tracking error at a few initial time steps, then zero error at all other time steps can be achieved. This approach can also be seen as using multiple zero order holds for the first addressed times steps. Mathematically, one modifies Eq. (2-5) by eliminating the chosen number of entries at the beginning of the error vector on the left, and the corresponding entries of $\underline{f}$ on the right. And one eliminates the same number of initial rows in the matrix P, producing matrix P_a which contains only the rows associated with the addressed time steps. We refer these remaining time steps where we are aiming to get zero tracking error as the addressed time steps, and all the other time steps as unaddressed time steps.

In this case, the control action in each iteration is updated only based on error at addressed time steps, but the control action is still a full length vector updated for all steps. The errors at the initial unaddressed time steps are not zero and are determined by the control action at the initial time steps chosen by the ILC law. In implementation, one can always append some extra time steps before the start of the actual desired trajectory.

The singular values of the deleted matrix P_a are all well behaved and match the magnitude response of the system. The bad singular values have been eliminated. Consider the system in Eq. (2-15) sampled at 100 Hz with a desired trajectory that is 100 time steps long. Let the desired output be $1-\cos(2\pi t)$ and apply the partial isometry law. In Fig. 3-15, we do not ask for zero tracking error at just the first time step since there is one non-minimum phase zero. The RMS error

at addressed time steps reduces quickly within a few initial iterations, and continues to decrease to around 10^{-7} in 10,000 iterations. The error at the first time step decays to around 10^{-3}. This result is obtained using unity gain $\phi = 1$. Control action at the 10,000th iteration is given in the right plot.

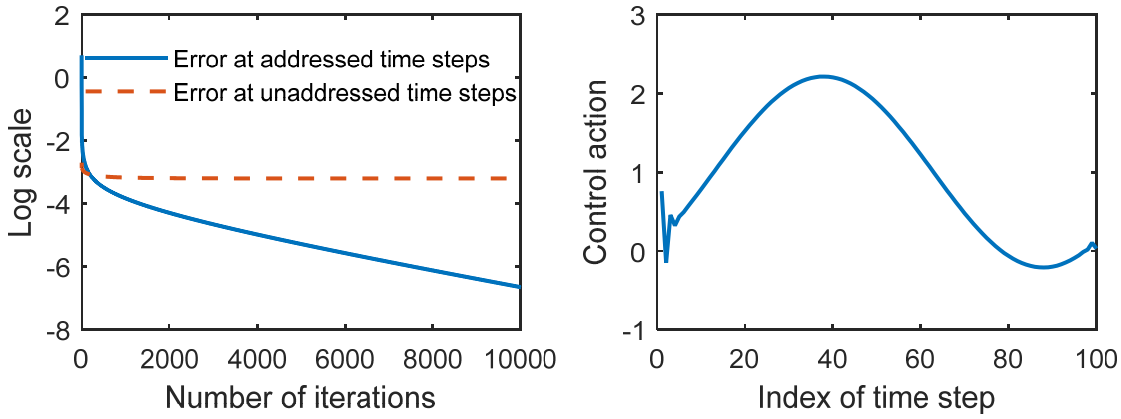

Figure 3-15. RMS error in different iterations and control action in the 10,000th iteration if using Approach (A)

3.7 Conclusions

Iterative learning control is very often an ill-posed problem when the system has non-minimum phase zeros. There are two ways in which it is ill posed. (1) In practice it most likely does not reach a final error level that is really close to zero for the addressed points. (2) If it does really reach a good final error level at the addressed points, then the control action and error between time steps is likely to be growing exponentially with time.

We propose a new approach which allows more than one zero order hold for each addressed sample. Addressed samples means the time steps where we ask for zero error. The previous chapter shows that there are anomalous singular values and singular vectors when the system has non-minimum phase zeros. It is shown in this chapter that original anomalous singular values and vectors of the system matrix can be eliminated when applying a multiple zero order holds approach. But there are new anomalous singular values and vectors. They are not as serious from a practical

point of view: (i) They are not as small, and hence might be tolerable. (ii) The number of these new anomalous singular values is smaller. (iii) When there are a reasonable number of time steps in the trajectory, these singular values become unaffected by increases in the total number of steps in the trajectory, rather than getting progressively worse. (iv) They are not associated with instability and have both input and output singular vectors that decay with time step. This decay with time step suggests that there may be a relationship between these singular values and the decaying of transients in the system, but no connection is established here.

One can modify the problem further so that there are no anomalous singular values. By not asking for zero error at a number of initial time steps equal to the number of new anomalous singular values in the skip step matrix, all singular values will relate to the system frequency response. This then forms the basis for the design of iterative learning control systems associated with well-behaved inverse solutions.

Finally, we develop a quadratic cost based ILC law that allows more than one zero order hold input between successive time steps at which one seeks convergence to zero error. It is demonstrated that this form of a generalized hold eliminates the ill-conditioning problem, making the ILC problem well posed with respect to both of the items above. It eliminates the difficulty of the extremely slow learning in one part of the error space, and it can reach zero error at the addressed points. In order to fully benefit from the approach to produce significantly improved overall error levels both at the sample times and between the sample times, one should use the extra zero order hold inputs between each addressed sample time and before if there are new anomalous singular values, in order to reduce the error between the addresses sample times. This error cannot be made zero, but it is seen that we can make it very close to zero.

Simulation results presented in this chapter are focused on systems with sampling zeros. it is also shown that initial deletion is another effective approach to address the instability issue. We extend these approaches to the solution of the internal instability due to intrinsic zeros in the next chapter.

3.8 References

[1] 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.

[2] Freeman, C.T., Lewin, P.L. and Rogers, E., "Experimental Evaluation of Iterative Learning Control Algorithms for Non-minimum Phase Plants," *International Journal of Control*, Vol. 78, No. 11, pp. 826-846.

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

[4] D. H. Owens, B. Chu, Z. Cai, E. Rogers, C. T. Freeman and P. L. Lewin, "Modeling the Influence of Non-minimum Phase Zeros on Gradient Based Linear Iterative Learning Control," *IEEE International Conference on Control Applications*, Yokohama, 2010, pp. 392-397.

[5] S. Devasia, D. Chen and B. Paden "Nonlinear Inversion-Based Output Tracking," *IEEE Transactions on Automatic Control*, July 1996, Vol. 41, No. 7, pp. 930-942.

[6] Kinosita, K., Sogo, T. and Adachi, N., "Iterative Learning Control Using Adjoint Systems and Stable Inversion," *Asian Journal of Control*, March 2002, Vol. 4, No. 1, pp. 60–67.

[7] X. Wang and D. Chen, "An Inversion-Based Iterative Learning Control Algorithm for a Class of Nonminimum Phase Systems," *IEEE Proceedings - Control Theory and Applications*, Jan. 2005, Vol. 152, No. 1, pp. 72-78.

[8] P. A. LeVoci and R. W. Longman, "Intersample Error in Discrete Time Learning and Repetitive Control," *Proceedings of the 2004 AIAA/AAS Astrodynamics Conference*, Providence, RI, August, 2004.

[9] R. W. Longman, P. A. LeVoci, and T. Kwon, "Making the Impossible Possible in Iterative Learning Control," *Proceedings of the Thirteenth Yale Workshop on Adaptive and Learning Systems*, Center for System Science, Yale University, New Haven, CT, May-June 2005, pp 99 - 106.

[10] R. W. Longman, T. Kwon, and P. A. LeVoci, "Making the Learning Control Problem Well Posed - Stabilizing Intersample Error," *Advances in the Astronautical Sciences*, Vol. 123, 2006, pp 1143 - 1162.

Chapter 4

Applying the Multiple Zero Order Hold Approaches to Systems with Intrinsic Non-minimum Phase Zeros

4.1 Introduction

As mentioned before, there are two types of non-minimum phase zeros in digital systems. Sampling zero is introduced during discretization for systems with pole excess of 3 or more using a zero order hold input. Intrinsic zero is mapping of the non-minimum phase zeros of the continuous time system to their discrete time image. In Chapter 3, we proposed multiple zero order holds approach to solve the instability problem for systems with sampling zeros. Instead of asking for zero tracking error at every time step, the basic concept here is to only address certain time steps.

We have introduced two approaches, the first approach simply avoids asking for zero error at a few initial time steps of the desired trajectory. The number of steps that must be skipped is equal to the number of non-minimum phase zeros. The second approach uses multiple zero order hold inputs between addressed time steps. It provides extra freedom that allows one to obtain zero tracking error at addressed time steps while keeping the error at unaddressed steps to a minimum level.

References 1-2 propose an approach named advanced output data ILC (ADILC). This approach shifts the mapping between output and input by deleting a few rows at the beginning and columns at the end. And the number of rows (columns) to delete is equal to the number of relative degree plus the number of non-minimum phase zeros. The system matrix in this approach is still square,

meaning the input and output has the same dimension. The number of rows (columns) to delete at the beginning (end) has to be exactly the number as indicated before. If deleting less or more, the bad (or anomalous) singular values are still there, and in some cases, become even worse.

Our approach of deleting initial rows gives a non-square system matrix with more inputs than outputs, the number of rows to delete is the same as the number of NMP zeros (assuming the relative degree is one). This approach also requires the exact number to get rid of all bad singular values. However, if deleting more than enough, there is no bad behavior except some wiggles at the beginning in both control action and error. If deleting less rows than needed, the number of bad singular values eliminated is equal to the number of rows deleted. And the worst bad singular value will be gone first, so the remaining bad singular values are not as bad.

When applying the two approaches to the general non-minimum phase digital systems, our results suggest that the number of initial time steps one needs to skip is greater than the number of non-minimum phase zeros when the system contains intrinsic zeros. For skipping time steps in between addressed steps, the original bad singular values are gone, but new bad singular values may be introduced. To eliminate these new ones, it is not enough to only delete the number of initial addressed time steps identical to the number of new bad singular values.

This Chapter investigates the behavior of these two approaches for different non-minimum phase systems. Numerical experiments indicate that zeros and poles inside the unit circle influence the performance of the approaches. It is also shown that the performance of the approaches are affected by the locations of zeros. The closer the location of an intrinsic zero is to the unit circle, the more time steps need to be skip at the beginning.

Here we introduce a filter that eliminates these difficulties, and results in good performance. The filter inverts the stable part of the system, cancelling all poles and zeros inside the unit circle

of the *z*-transform plane. A method is developed to implement this filter in the raised system modeling used in discrete time ILC. The results indicate that again the instability of the inverse model difficulty is circumvented when one leaves a number of initial time steps in the trajectory to be free, where this number equals the number of non-minimum phase zeros. Instability of the inverse model issues are also eliminated when using the second approach with multiple zero order holds between each time step for which one asks for zero tracking error.

4.2 Two Approaches to Apply to NMP Systems with Intrinsic Zeros

This section gives a summary of the two approaches we try to apply in non-minimum phase system with intrinsic zeros.

Approach A: Initial Time Steps Deletion

Instead of asking for zero tracking error at all time steps, the first approach suggests that one does not ask for zero tracking error at a few initial time steps. The right number of rows to skip at the beginning is identical to the number of non-minimum phase zeros, assuming the relative degree in discrete time is one. One can append extra time steps before the start of the actual trajectory in practice.

The control action in each iteration is a full length vector contains all sample times, but it is updated based on only the error of addressed time steps. Chapter 3 shows that this approach can successfully eliminate the internal instability due to sampling zeros, except for these appended time steps, zero error can be achieved at the rest. The control action is also well-behaved.

Approach B: Multiple Zero Order Holds

If applying a zero order hold, only one scalar quantity is used at each time step. In the second approach, we allow more than one zero order hold value between the time steps where we asked for zero tracking error, and such time steps are named as addressed time steps. If one wants zero

error at the time steps at the original sample rate, then one increases the sample rate to allow adjusting the hold action at several intermediate time steps. When applying this approach, it is equivalent to take every *n*th row of the full *P* matrix written for all time steps, where *n*-1 represents the number of time steps being skipped. For example, if the aim is to ask for zero error every time step running at 100Hz sample rate as in the example given in the previous section, we can update the control action at 200Hz sample rate, thus allowing two zero order hold values to be used to adjust the error at the next 100Hz sample time. This can be considered as a form of generalized hold.

However, there may be some extra anomalous singular values introduced during this skipping process. These new bad singular values have different properties than the original ones in the sense that: (i) They are not as small, and hence error in the associated part of the error space might converge fast enough in applications. (ii) The number of these new anomalous singular values is smaller. (iii) Beyond a small threshold value, these singular values are not influenced by the number of time steps in the desired trajectory, whereas the original bad singular values get progressively worse with trajectory length. (iv) They are not associated with instability and have both input and output singular vectors that decay with time step.

However, the magnitude of these new singular values are still relatively small compared to the ones related to frequency response. They can be eliminated by skipping the number of rows equal to the number of new bad singular values. This can also be seen as a combination of deleting initial rows and using multiple zero order holds between addressed time steps.

Both approaches use a matrix that only contains addressed time steps, as defined before, we denote this rectangular matrix as P_a, and singular value decomposition is

$$P_a = U_a[S_a \quad 0]\begin{bmatrix} V_{a,1}^T \\ V_{a,2}^T \end{bmatrix} = U_a S_a V_{a,1}^T \tag{4-1}$$

4.3 Difficulties of Applying the Approaches in Systems with Intrinsic Zeros

The internal instability coming from the sampling NMP zeros can be eliminated using the two approaches described before. In this section, we consider the intrinsic NMP zeros, they usually appear on the right z plane, and will approach +1 as the sampling rate goes to zero. They can be real valued or complex conjugate pairs.

We use the same example as shown in Chapter 2, the model is

$$\frac{0.8 - s}{(s+1.6)(s+0.5)} \tag{4-2}$$

Sampling at 10 Hz there is a zero mapped to z = 1.0835. The SVD of the full P matrix gives one bad singular value which is shown in Fig. 2-8 in Chapter 2. We construct a 100 by 100 P matrix and apply Approach A which deletes the first row of this matrix to form P_a. Its singular values are shown in the left plot of Fig. 4-1, and we observe that there is still one bad singular value around $10^{-1.5}$, although it is not as bad as the original one for the full P matrix which is close to 10^{-4}.

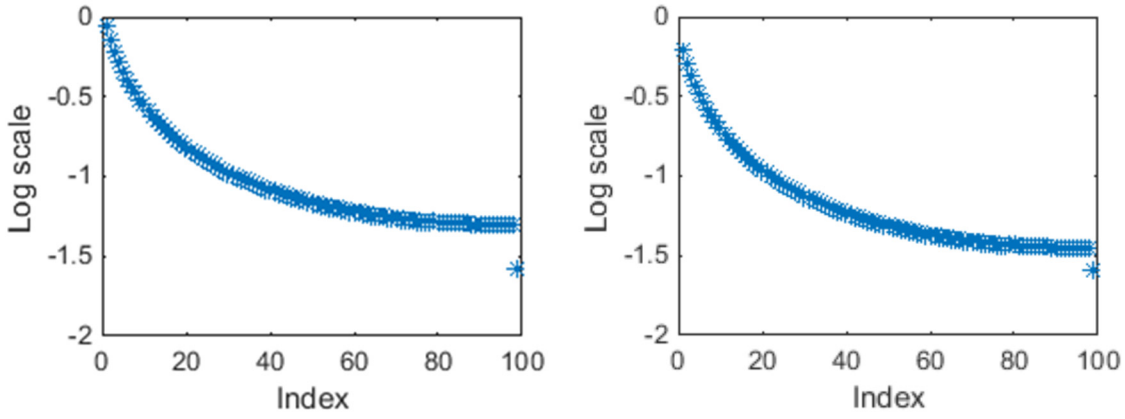

Figure 4-1. Singular values of P_a if using Approach A (left) and B (right) for system with intrinsic zeros

Now apply Approach B that uses two zero order holds between each sample time step at 10Hz. To model this, we construct a 200 by 200 P matrix for sample rate 20 Hz, then delete all odd numbered rows. After doing this, delete the first row to form the associated P_a. The singular values of this matrix are shown on the right of Fig. 4-1, and again we observe that there is still one bad singular value. Therefore, although Approaches A and B solve the bad singular value problem for sampling zeros, they fail to solve the problem for intrinsic zeros. Both approaches have new bad singular values (numerical experience indicates that when the intrinsic zero is far enough from the unit circle, it is possible that no new bad singular values appear). Below we will show that by deleting enough additional initial rows in both approaches, one can eliminate all bad singular values.

Examine the singular vectors associated with these new bad singular values for Approach A. They have different properties than the singular vectors of the original bad singular values. The properties of new bad singular values for Approach B applied to sampling zeros, listed in previous section, are now exhibited for new bad singular values introduced by both approaches when applied to intrinsic zeros. For the original bad singular values, the associated input singular vector grows each time step by a factor equal to the zero location, and the associated output singular vector decays with time step by the reciprocal of this zero location. The growth of the input singular vector with time step is the source of the internal instablity of the control action. In constrast to this, the new bad singular values have singular vectors that both decay. The left plot in Fig. 4-2 shows this decay clearly for output singular vector (solid line), with the same slope. The dashed curve for the input singular vector is somewhat irregular, but clearly displays this same slope of decay (rather than growth) when the size of the P matrix is increased to 500 by 500, as shown in the right plot of Fig. 4-2.

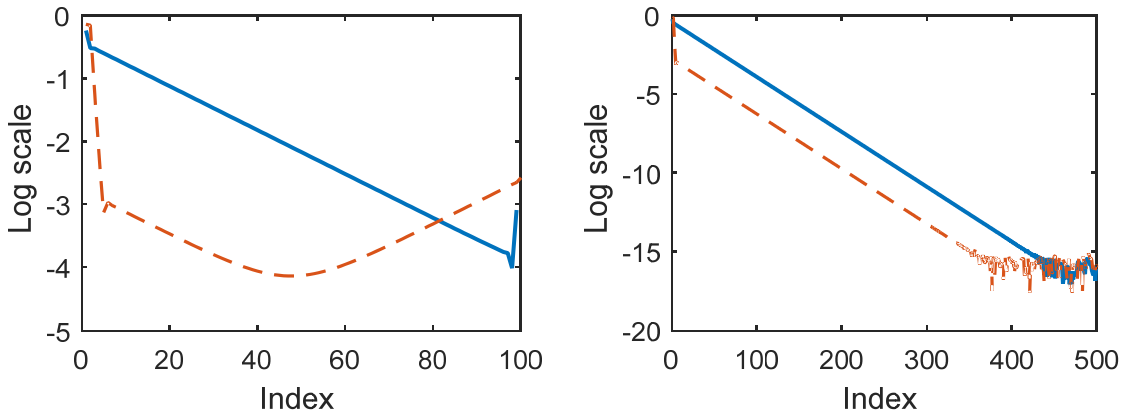

Figure 4-2. Singular vectors associated with the new bad singular value after applying Approach A

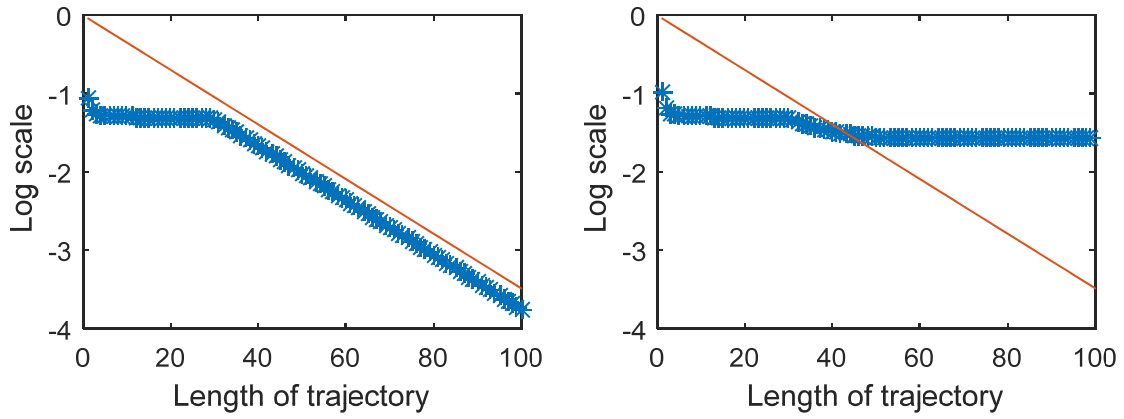

Figure 4-3. Magnitude of the original bad singular value (left) and new bad singular value (right) vs length of trajectory, Approach A

Figure 4-3 shows a very important difference between the original and the new bad singular values. The original bad singular values get worse as the size of the P matrix increases (left plot), decreasing by a factor of the reciprocal of the zero location magnitude for each unit increase in dimension (solid line). The new bad singular value is rather insensitive to the dimension of P_a.

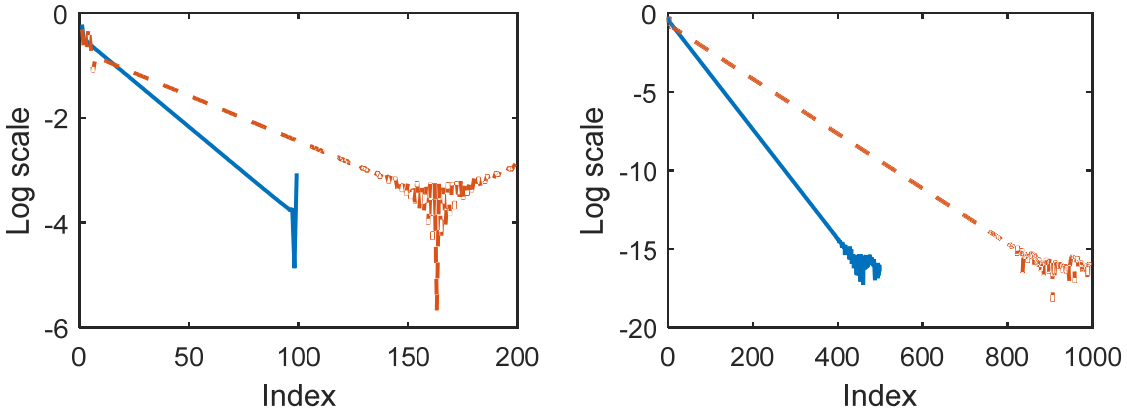

Figure 4-4. Singular vectors associated with the new bad singular value after applying Approach B

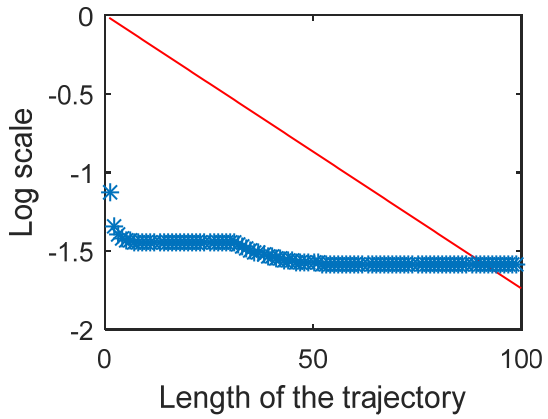

Figure 4-5. Magnitude of the bad singular value vs length of trajectory, Approach B

Figures 4-4 and 4-5 show the figures for Approach B corresponding to Figs. 4-2 and 4-3. The same properties are again exhibited. This time the *P* matrix starts with dimension 200 by 200. The left plot of Fig. 4-4 gives the column vector entries for each time step for the singular vectors associated with the bad singular value in the right plot of Fig. 4-1. The solid line corresponds to the output singular vector which has dimension 99, and the dashed curve is for the input singular vector of dimension 200. The two slopes are actually the same when viewed in real time, and appear different because of the faster sample rate on the input than on the output. As in the right plot of Fig. 4-2, increasing the size of the original *P* matrix makes the slope more consistent all the

way to the numerical noise floor. Again, Fig. 4-5 shows that the new bad singular value is nearly independent of the size of the P matrix.

Approach A for sampling zeros asks one to delete the number of initial rows of the original P matrix equal to the number of such zeros outside the unit circle. Approach B asks one to first skip every odd numbered time step of a P for the increased sample rate, and then delete this same number of initial time steps from the new matrix. It is logical to ask, can we eliminate the new bad singular values that have appeared when these methods are applied to intrinsic zeros, by deleting more initial rows in each case. We will show that this can work, but there seems to be no rule indicating how many extra rows need to be deleted. In the next section we will present a different method to eliminate the new bad singular values, that asks one to cancel the poles and zeros inside the unit circle as part of the ILC law. Once this is done, the rule for the number of rows to delete matches that given for sampling zeros, i.e. the number of zeros outside the unit circle.

Table 4-1 investigates the influence of the location of non-minimum phase zeros on the number of initial rows to be deleted for both approaches, such that there are no bad singular values remaining. In order to compare the difference between sampling zeros and intrinsic zeros, we use the third order system in Chapter 3 that has one sampling zero when convert to digital system. The transfer function is given as

$$G(s) = \left(\frac{a}{s+a} \right) \left(\frac{\omega_0^2}{s^2 + 2\zeta\omega_0 s + \omega_0^2} \right) \tag{4-3}$$

where $a = 8.8$, $\zeta = 0.5$, $\omega_0 = 37$.

The first part of the table considers the system in Eq. (4-3) which only has sampling zeros, and three different zero location for the non-minimum phase zero are presented, based on picking three different sample rates. The number of initial rows that must be deleted using Approach A is

unaffected by the zero locations, and is equal to one because there is only one zero outside the unit circle. The first column under Approach B indicates how many new bad singular values are present in the P_a matrix for double the original sample rate, when one has skipped every odd numbered row without deleting an initial row after doing so. This number is one new bad singular value except when the zero location gets close to -1, in which case no new bad singular value appears. The second column under Approach B says how many additional rows need to be deleted so that no new bad singular value is present, and this number is equal to the number of new bad singular values indicated in the first column of Approach B. The rule asking to delete rows based on the number of zeros outside the unit circle still applies.

Table 4-1. Number of initial rows to skip vs locations of non-minimum phase zero

Zero type	Zero location	Approach A	Approach B	
		Number of initial rows to delete	Number of new anomalous singular value	Extra addressed rows to delete
Sampling zeros	-3.3104	1	1	1
	-2.1448	1	1	1
	-1.0475	1	0	0
Intrinsic zeros	1.0835	3	1	3
	1.1755	2	1	2
	2.3179	1	1	1

The second part of Table 4-1 makes the same study for intrinsic zeros, using the system in Eq. (4-1) with different choices of sample rate. This time the number of rows that must be deleted using Approach A is a function of how far the single zero is from the unit circle. Deleting one row is enough when the zero is at 2.3179, but one must delete 2 rows when the zero is at 1.1755, and one needs 3 rows deleted when the zero is at 1.0835. For all three zero locations, Approach B has one new bad singular value if no initial rows are deleted. But the number of rows that must be

deleted in Approach B to eliminate this new bad singular value is the same as in Approach A to eliminate the original bad singular value.

Table 4-2. Number of initial rows to skip vs locations of poles and zeros inside the unit circle

Zeros	Poles	Approach A	Approach B	
		Number of initial rows to delete	Number of new anomalous singular value	Extra addressed rows to delete
1.0835	0.4, 0.5	1	1	1
	0.5, 0.6	2	1	1
	0.8, 0.9	3	1	2
1.5, -0.6	0.7, 0.8, 0.9	1	1	1
1.5, 0.6	0.7, 0.8, 0.9	2	1	1

We now fix the non-minimum zero location, and see how the performance is affected by the zeros and poles inside the unit circle. Table 4-2 gives some examples. The first part of the table shows that for a 2nd order system with one intrinsic zero, when changing the two pole locations inside the unit circle, the number of rows that must be skipped at the beginning varies for both Approaches A and B. The latter part of the table considers a 3rd order system with fixed pole locations inside the unit circle, and fixed non-minimum phase zero location outside the unit circle, and shows the effect of a different zero location inside the unit circle. If the zero location is on the negative real axis which can happen for sampling zeros, only one row needs to be deleted, but when it is on the positive real axis, 2 rows must be deleted using Approach A. Only one is needed using Approach B to eliminate the new bad singular value introduced when skipping steps.

4.4 Proposed Method for Designing ILC Laws for Systems with Intrinsic Zeros

4.4.1 Introduce a Pole-Zero Cancelation Filter in Matrix Form into ILC Laws

Approaches A and B were developed for sampling zeros, and prescribed the number of initial rows that need to be deleted to be equal to the number of non-minimum phase sampling zeros. The

previous section showed that this rule does not work for intrinsic zeros, and that the number of initial rows that need to be deleted to eliminate all bad singular values can be strongly influenced by the location of poles as well as zeros inside the unit circle. Repetitive control designs often design a transfer function compensator to apply to the error in the previous run that cancels all cancelable zeros and poles, i.e. those inside the unit circle.[3] Reference 4 develops effective repetitive control design methods based on the inverse of the frequency response of the system, and Ref. 5 shows how to combine this with cancelation of zeros and poles inside the unit circle. In this section we develop the method to perform the equivalent of pole-zero cancellation for ILC, which must be in the form of a matrix operator instead of a filter in difference equation form.

Consider a general discrete-time non-minimum phase system which can contain intrinsic zeros mapped from the original continuous-time NMP zeros, and sampling zeros which are introduced during the discretization process

$$G(z) = G_O(z)G_I(z) \tag{4-4}$$

which we divide into two parts. G_I represents the invertible part, i.e. all zeros and poles inside the unit circle. And G_O which contains all non-minimum phase zeros. P is the system matrix for $G(z)$ that produces the output history for any history of inputs for zero initial conditions. We wish to write this P as a product

$$P = P_O P_I \tag{4-5}$$

where P_I is the system matrix associated with transfer function $G_I(z)$. The $G_O(z)$ is a transfer function of only zeros, and contains no poles. Hence it cannot be generated by the following equation

$$y(k) = CA^k x(0) + \sum_{i=0}^{k-1} CA^{k-i-1}Bu(i) + v(k) \tag{4-6}$$

but nevertheless there must exist a matrix P_O that relates the input history of $G_O(z)$ to its output history.

To understand the nature of these three matrices, consider a 2nd order discrete time system with one non-minimum phase zero. First we develop matrix *P*. The corresponding difference equation can be written as

$$y(k+2)+a_1y(k+1)+a_2y(k)=b_1u(k+1)+b_2u(k) \tag{4-7}$$

To convert this difference equation to state-space, we define an intermediate variable $\overline{y}(k)$ that is the solution of

$$\overline{y}(k+1)+a_1\overline{y}(k)+a_2\overline{y}(k-1)=u(k) \tag{4-8}$$

which can be converted to state variable form using the state definition

$$x(k)=\begin{bmatrix}\overline{y}(k-1)\\ \overline{y}(k)\end{bmatrix} \tag{4-9}$$

The solution to difference Eq. (4-8) can be written in terms of these state variables by superposition

$$y(k)=b_1\overline{y}(k)+b_2\overline{y}(k-1) \tag{4-10}$$

When the initial conditions are all zero (and there is no disturbance), multiplying matrix *P* times the input history vector in $\underline{y}_j=P\underline{u}_j+\overline{A}x(0)+\underline{v}$ produces the output history vector. Setting $x(0)=0$, requires $\overline{y}(-1)=0$ and $\overline{y}(0)=0$. Now we relate these initial conditions for $\overline{y}$ to the corresponding initial conditions for *y*. When *k* equals 0, Eq. (4-10) gives the first initial condition of the difference equation

$$\text{Initial condition I: } y(0)=0 \tag{4-11}$$

When *k* equals to -1, Eq. (4-8) gives

$$u(-1)=a_2\overline{y}(-2) \tag{4-12}$$

and Eq. (4-10) gives

$$y(-1) = b_2 \overline{y}(-2) \tag{4-13}$$

Equations (4-12) and (4-13) gives the second initial condition for Eq. (4-7)

$$\text{Initial condition II: } a_2 y(-1) = b_2 u(-1) \tag{4-14}$$

One can pick any $u(-1)$ and $y(-1)$ satisfying this equation and the initial state for Eq. (4-9) is zero.

Now consider $G_1(z)$. This transfer function has the zero removed, and hence the associated difference equation has the form

$$\hat{y}(k+2) + a_1 \hat{y}(k+1) + a_2 \hat{y}(k) = cu(k) \tag{4-15}$$

The input to this equation is the same input history as in Eq. (4-7), but of course the output is different and we denote it by $\hat{y}$. Pick the state variables for this equation as

$$\hat{x}(k) = \begin{bmatrix} \hat{y}(k-1) \\ \hat{y}(k) \end{bmatrix} \tag{4-16}$$

Zero initial conditions require that $\hat{x}(0) = 0$, or $\hat{y}(-1) = 0$ and $\hat{y}(0) = 0$. Superposition gives the output history of $G(z)$ in terms of $\hat{y}$ as

$$y(k) = d_1 \hat{y}(k) + d_2 \hat{y}(k+1) \tag{4-17}$$

This equation represents the action of $G_O(z)$ which cannot be put in the form of a state variable equation in order to produce P_O, but we still seek a matrix to represent this transfer function. Because $\hat{y}(-1) = 0$ and $\hat{y}(0) = 0$, Eq. (4-17) implies $y(-1) = 0$. In order to satisfy the second initial condition of the original system as in Eq. (4-14), $u(-1)$ is also equal to 0. When *k* equals -1, Eq. (4-15) becomes

$$\hat{y}(1) + a_1 \hat{y}(0) + a_2 \hat{y}(-1) = cu(-1) \tag{4-18}$$

Because $\hat{y}(-1)$, $\hat{y}(0)$ and $u(-1)$ all equal 0, $\hat{y}(1)$ is also a hidden initial condition that must be equal to zero. This suggests that in Eq. (4-17), the output of the first time step needs an input $\hat{y}(2)$

$$y(1) = d_2 \hat{y}(2) \tag{4-19}$$

The original system takes $u(0)$ as the first input, and $y(1)$ as the first output, there is one time step delay through the system, and the associated CB is nonzero. Thus given an input at $u(0)$, in order to have $y(1)$ as the output, P_I needs to produce $\hat{y}(2)$ to feed into P_O. This means that we should construct a P_I whose first input is $u(0)$ and first output is $\hat{y}(2)$, i.e. the associated CB is zero for system Eq. (4-15). Assume the disturbance is 0. Because $\hat{x}(0) = 0$, $y_j(k) = CA^k x(0) + \sum_{i=0}^{k-1} CA^{k-i-1} B u_j(i) + v(k)$ for G_I becomes

$$\hat{y}_j(k) = \sum_{i=0}^{k-1} CA^{k-i-1} B u_j(i) \tag{4-20}$$

This time, k starts from 2 and i starts from 0, the P_I matrix of this system is

$$P_I = \begin{bmatrix} CAB & 0 & 0 & \cdots & 0 \\ CA^2B & CAB & 0 & \cdots & 0 \\ CA^3B & CA^2B & CAB & \ddots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ CA^{p-1}B & CA^{p-2}B & CA^{p-3}B & \cdots & CAB \end{bmatrix} \tag{4-21}$$

and $CAB \neq 0$, thus the inverse of the P_I matrix is guaranteed to exist, and the ill-conditioning associated with non-minimum phase zeros is not present. We can use P_I^{-1} as a filter to cancel everything inside the unit circle. The remaining system can use the following matrix to represent

$$P_O = P P_I^{-1} \tag{4-22}$$

We use P_I^{-1} as a filter in the ILC design. The ILC law is generated for system P_O, and P_I^{-1} is applied to the data to cancel the P_I dynamics of the real world, leaving P_O to represent the remaining system. Consider the contraction mapping law applied to P_O

$$\underline{u}_{j+1} = \underline{u}_j + \phi P_O^T \underline{e}_j \tag{4-23}$$

The error update from iteration to iteration is given by

$$\underline{e}_{j+1} = [I - \phi P_O P_O^T]\underline{e}_j \tag{4-24}$$

The two candidate approaches ask for zero error at the chosen addressed time steps. Denote the error vector at these time steps as e_a, and Eq. (4-24) becomes

$$\underline{e}_{a,j+1} = [I - \phi P_{O,a} P_{O,a}^T]\underline{e}_{a,j} \tag{4-25}$$

where $P_{O,a}$ is the P_O matrix with only addressed rows which can be given by

$$P_{O,a} = P_a P_I^{-1} \tag{4-26}$$

The P_a is described for both Approaches in Section III. Equation (4-25) then becomes

$$\underline{e}_{a,j+1} = [I - \phi P_a P_I^{-1} P_{O,a}^T]\underline{e}_{a,j} = [I - P_a \phi P_I^{-1} P_{O,a}^T]\underline{e}_{a,j} \tag{4-27}$$

Thus the new contraction mapping law to apply both Approach A and B with filter introduced is given as

$$L_1 = \phi P_I^{-1} P_{O,a}^T \tag{4-28}$$

Following the same logic, we can derive the new partial isometry law as

$$L_2 = \phi P_I^{-1} V_{O,a,1} U_{O,a}{}^T \tag{4-29}$$

where $V_{O,a,1}$ and $U_{O,a}$ are from the singular value decomposition of $P_{O,a}$

$$P_{O,a} = U_{O,a}[S_{O,a} \quad 0]\begin{bmatrix} V_{O,a,1}{}^T \\ V_{O,a,2}{}^T \end{bmatrix} = U_{O,a} S_{O,a} V_{O,a,1}{}^T \tag{4-30}$$

The new Quadratic Cost ILC Law is

$$L_3 = \phi P_I^{-1}(rI + P_{O,a}^T P_{O,a})^{-1} P_{O,a}{}^T \tag{4-31}$$

Examining the development in Eqs. (4-17) to (4-19) one can write the P_O matrix as

$$P_O = \begin{bmatrix} d_2 & 0 & \cdots & 0 & 0 \\ d_1 & d_2 & \cdots & 0 & 0 \\ 0 & d_1 & \ddots & \vdots & \vdots \\ 0 & 0 & \ddots & d_2 & 0 \\ 0 & 0 & \cdots & d_1 & d_2 \end{bmatrix} \tag{4-32}$$

In order to perform the pole-zero cancelation applying the learning law to the physical world, we must use Eq. (4-26) instead of Eq. (4-32).

4.4.2 Simulation Results

Consider the same example as in Eq. (4-2). When sampled at T = 0.1 s, the discrete system transfer function is

$$G = \frac{-0.08635(z-1.0835)}{(z-0.9512)(z-0.8521)} \tag{4-33}$$

so that $G_I = K/[(z-0.9512)(z-0.8521)]$, where K is chosen so that the largest singular value of P_O is unity. Its singular values are shown in Fig. 4-6, and there is one particularly small singular value, corresponding to the NMP zero. In Fig. 4-7, the left plot shows that when deleting one initial row of P_O the bad singular value is gone. The right plot shows that if one keeps every other row, the bad singular value is also gone, and all remaining singular values have almost the same magnitude equal to one. We see that both approaches eliminate the bad singular value when using the filter.

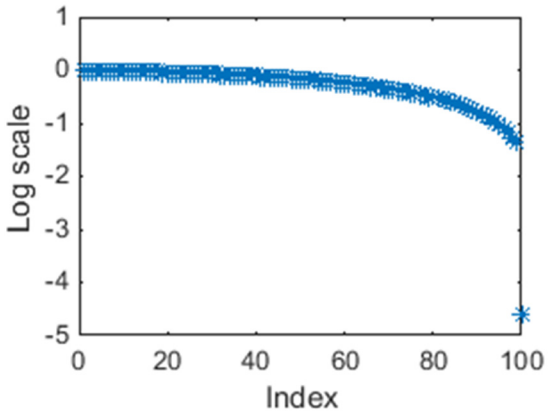

Figure 4-6. Singular values of P_O

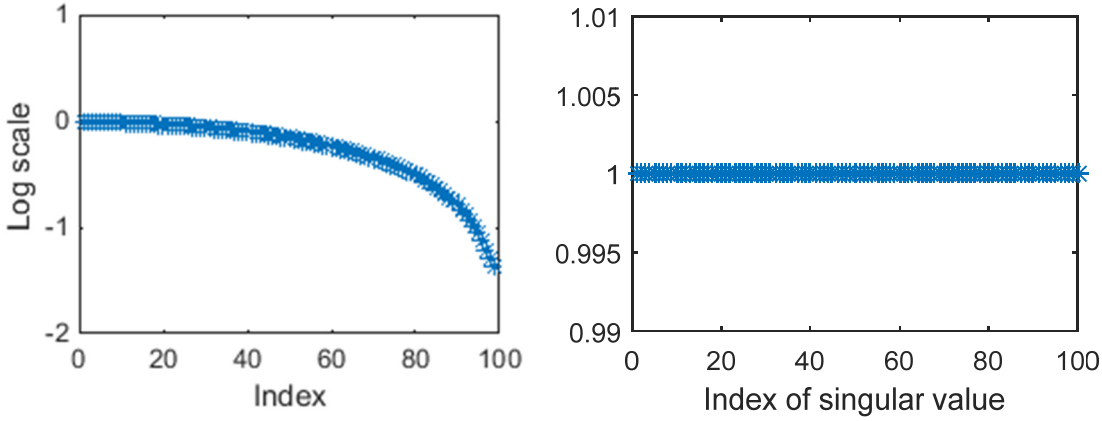

Figure 4-7. Singular values of $P_{O,a}$ using Approach A (left) and B (right)

Now we study the behavior of the ILC approaches when using the *P* transpose law with gain $\phi = 1$. Consider a 100 time step trajectory, and a desired trajectory $0.5[1-\cos(0.2\pi t)]$. For comparison purposes, note that Fig. 4-8 gives the inverse solution that produces zero error at every time step, when the sample rate is 10Hz and the trajectory is 100 time steps, the control action exponentially grows to magnitude 6000 after 100 time steps and the error between time steps grows exponentially. This plot is the same as Fig. 2-7 in Chapter 2.

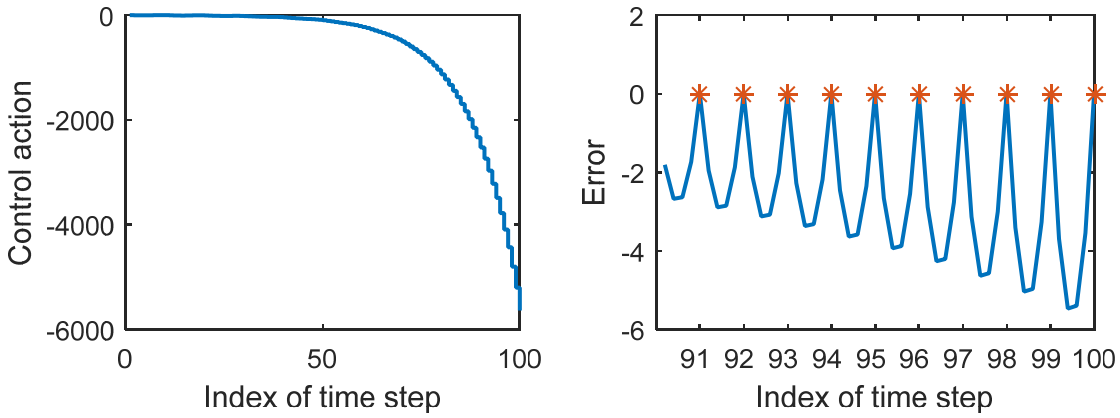

Figure 4-8. Control action (left) and error (right) if using inverse solution for system with intrinsic zeros

Figure 4-9 applies Approach A without using the filter. It is necessary to delete 3 initial rows in order to eliminate all bad singular values. The left plot gives the RMS error decay of the addressed points, and the dotted curve gives the nearly constant RMS error at the 3 unaddressed time steps. The error decays to RMS value of 10^{-12} at iteration 10,000 and continues to decrease. The control action at this iteration is given in the right plot and has a maximum value of about 10, which is to be compared to the value "6000" for the inverse solution.

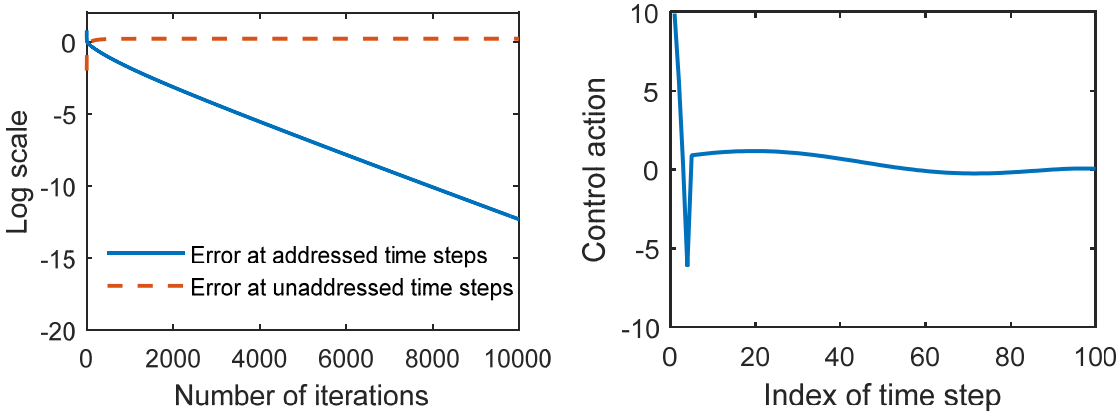

Figure 4-9. RMS error in different iterations and control action in the last iteration (Approach A, no filter)

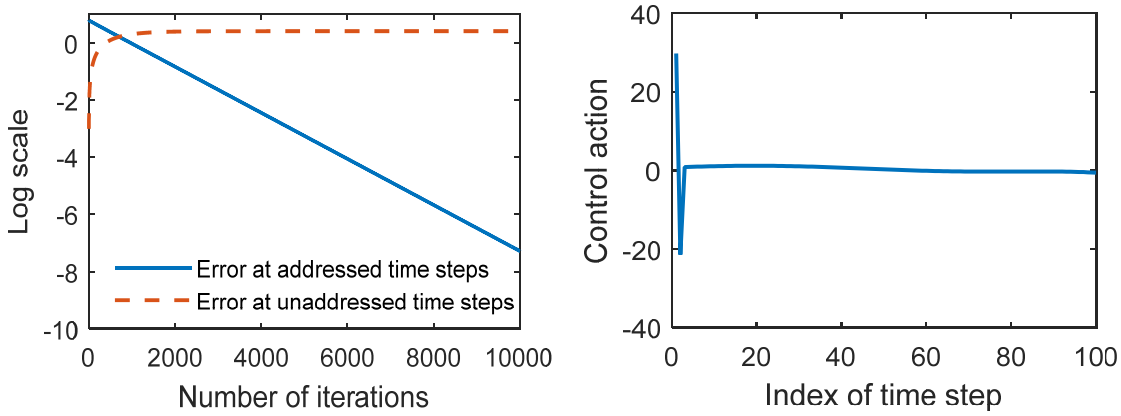

Figure 4-10. RMS error in different iterations and control action in the last iteration (Approach A, with filter)

Figure 4-10 introduces the filter that cancels poles and zeros inside the unit circle, using the normalized P_O. The error in the left plot reaches 10^{-8} at iteration 10,000 and the control action at this iteration has an initial peak of 30, to be compared to the "6000" value.

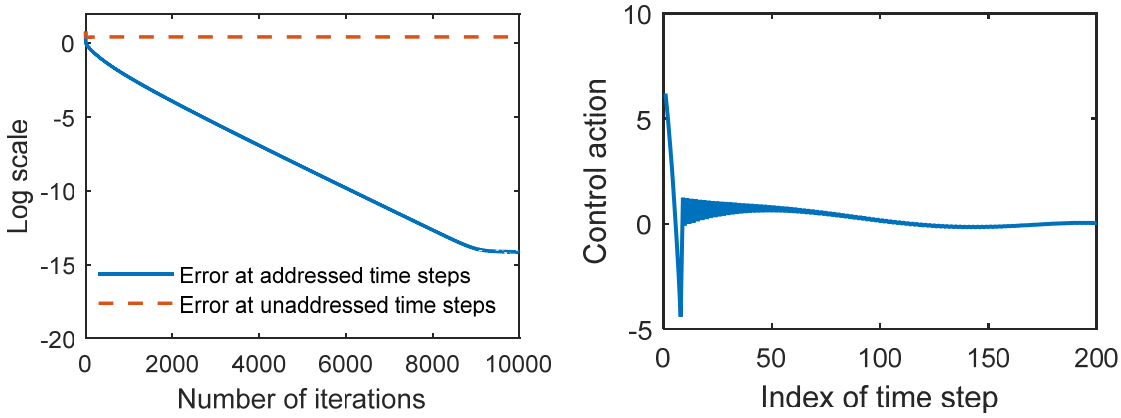

Figure 4-11. RMS error in different iterations and control action in the last iteration (Approach B, no filter)

Figure 4-11 applies Approach B without a filter. Again one must delete 3 initial rows, done after skipping every other row of the enlarged matrix. This time the RMS error has decayed to a numerical zero before reaching iteration 10,000. The associated control action is particularly small

at about 6 compared to "6000" and note that this time the control action has reduced the error at the addressed time steps to a numerical zero. However, the control action has a jitter in it wiggling every time step.

Figure 4-12 introduces the filter. This time the learning gain is reduced to $\phi = 0.1$. For this problem, after skipping every other time step, there is no bad singular value so it is not necessary to skip an initial row. From Fig. 4-7 all the singular values of $P_{O,a}$ are essentially one. This means that $(I - \phi P_{O,a} P_{O,a}^T)$ has all eigenvalues equal to $1-\phi$. With $\phi = 1$ the control action is inverting the system and producing zero error in one iteration. The approach has generated the inverse solution at all time steps with a control action that only reaches about 40, compared to "6000" for the true inverse, but again there is a jitter in the control action wiggling every time step. This new inverse solution is different because it uses two times the number of inputs as outputs.

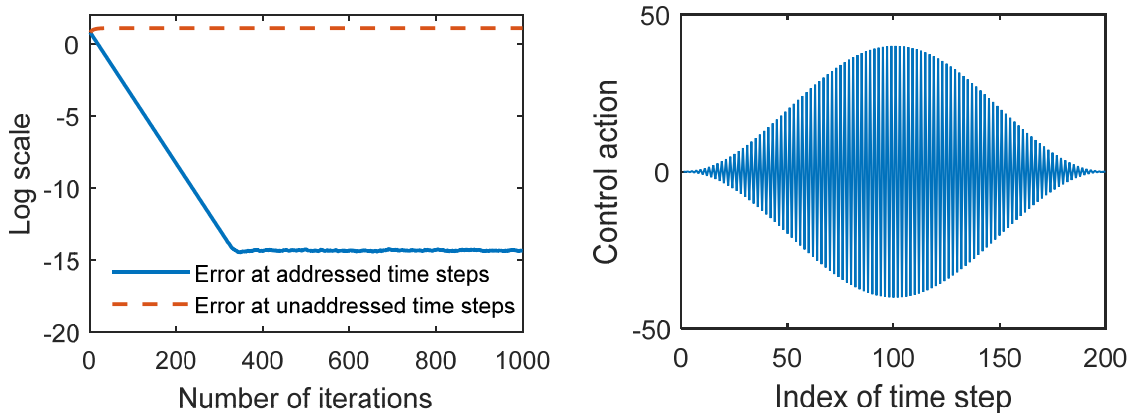

Figure 4-12. RMS error in different iterations and control action in the last iteration (Approach B, with filter)

Table 4-3 examines the influence of NMP zero location on the number of time steps that must be deleted in order to have no bad singular values. The zero locations are adjusted by changing the sample rate used on system Eq. (4-2). Compared to Table 4-1, we are able to see that after applying

the filter, the zero locations no longer have any influence on the number of initial rows that must be skipped at the start of the trajectory for the two approaches.

Table 4-3. Number of initial rows to skip vs locations of NMP zeros (with filter)

Zero type	Zero location	Approach A	Approach B	
		Number of initial rows to delete	Number of new anomalous singular value	Extra rows to delete
Intrinsic zeros	1.0835	1	0	0
	1.1755	1	0	0
	2.3179	1	0	0

Now consider a system that has both a sampling zero and an intrinsic zero

$$G(s) = \frac{16(0.8 - s)}{(s + 1.6)(s + 0.5)(s^2 + 4s + 16)} \tag{4-34}$$

Sampled at 10 Hz, there is a zero at 1.0833, and another zero at -3.1173.

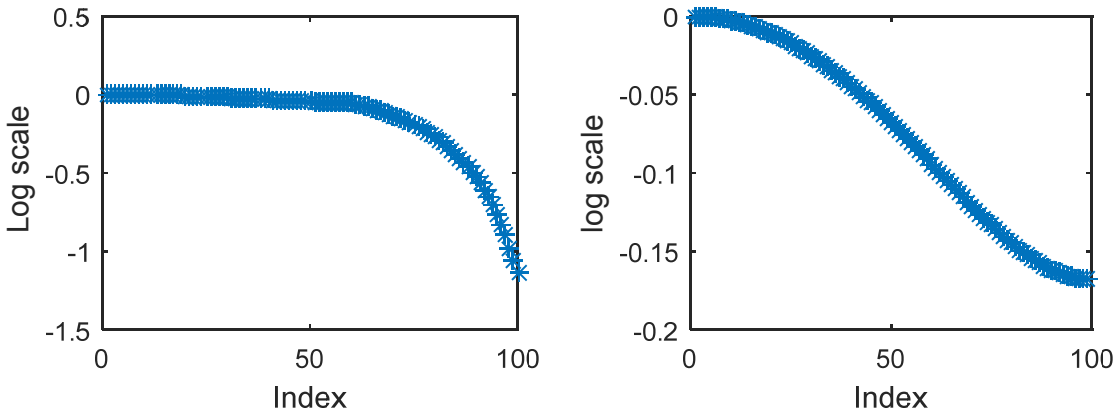

Figure 4-13. Singular values of $P_{O,a}$ after using Approach A (left) and B (right)

Applying Approaches A and B with the filter, there is no bad singular value as seen in Fig. 4-13. Note that for Approach B, we address every other time step and then delete one extra addressed

time step. The fact that all singular values are well-behaved means that there is no difficulty for the error at addressed time steps to converge to zero in a reasonable number of iterations.

4.5 Conclusion

ILC aims to get zero tracking error for systems performing a repeated tracking command. One wishes to see that the tracking error decreases from iteration to iteration, and finally converges to zero. The speed of convergence depends on the learning law. However, it is usually problematic when applying ILC in non-minimum phase systems. What one observes is that the error decreases, perhaps substantially, for some initial iterations and then appear to have finished converging, but not to zero error, rather to some disappointing non-zero error. The Toeplitz matrix relating the input history to the output history will have some particularly small singular value related to each non-minimum phase zero. When the error appears to have converged, the error components have converged to zero except for the component of the error on the output singular vector associated with the small singular value. Mathematically, the error will eventually go to zero after very many iterations. In practice it may not, because the control update correction in each iteration in the part of the space that is learning so slowly may be beyond the last digit in the digital converters.

A series of previous papers designed two approaches that chose not to ask for zero error at some time steps, but achieve zero tracking error at the remaining time steps. The two approaches have been applied to non-minimum phase systems that have sampling zeros, i.e. zeros introduced in the process of converting from continuous time system fed by a zero order hold to the equivalent discrete time model. The internal instability causing exponential growth of the control action is eliminated, and the error at the addressed time steps converges to zero.

This chapter examines the application of these approaches to general NMP systems with sampling zeros and intrinsic zeros, i.e. zeros that are images of non-minimum phase zeros in

continuous time. The rules of the two approaches do not work, and one often must no longer ask for zero error at more initial time steps, and we see no rule to say how many will be enough in a given problem. We address this issue by introducing a filter or compensator that cancels all system poles and zeros inside the unit circle. It is shown how this can be done in the matrix formulation necessary for iterative learning control. Effectively this leaves a system composed only of non-minimum phase zeros. The simulation result shows that the rules developed for sampling zeros again apply to intrinsic zeros and to combinations of both. The ILC law created by the second approach is observed to behave like an inverse if a full unity gain is applied, and can give very fast convergence determined by the chosen learning gain. This is an unusual inverse because it produces the inverse but employs two times as many control inputs as outputs to be controlled.

4.6 References

[1] Gu-Min Jeong and Chong-Ho Choi, "Iterative Learning Control with Advanced Output Data for Nonminimum Phase Systems," *American Control Conference,* Arlington, VA, 2001, Vol. 2, pp. 890-895.

[2] Gu-Min Jeong, Chong-Ho Choi, "Iterative learning control for linear discrete time nonminimum phase systems", *Automatica*, Vol. 38, Issue 2, 2000, pp. 287-291.

[3] 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.

[4] 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.

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

Chapter 5

Robustification of Iterative Learning Control Produced by Multiple Zero Order Hold Approaches

5.1 Introduction

Chapter 3 and 4 show that the two approaches we developed can successfully eliminate the internal instability problem of ILC for non-minimum phase systems. The system can be non-minimum phase due to either sampling zeros or intrinsic zeros or both.

This Chapter tries to investigate the stability or convergence robustness of these approaches to addressing the problem of instability of the inverse problem, and evaluate the robustness for three important classes of ILC laws. The model uncertainty is characterized by probability distributions for the uncertainty in the model coefficients. A Monte Carlo style study is made to determine what approaches and what ILC laws have the best stability robustness properties. For comparison purpose, we also presented the results for several other candidate approaches, these approaches can be found in References 1-4.

Chapter 3 shows that the rules for sampling NMP zeros are no longer effective when the system has intrinsic NMP zeros. A filter is then introduced to incorporate with the ILC laws. This requires the exact cancelation of zeros and poles inside the unit circle, which maybe in contrast with model parameter uncertainty. Thus, the numerical study presented in this chapter is mainly focused on only sampling NMP zeros.

5.2 Candidate Approaches to Address Instability of the Inverse Model

In ILC, a model is used to design the learning law, aiming to achieve zero tracking error in the model of the real world. To study the robustness of ILC with respect to model error, we use subscript W on P_W to indicate the matrix describing the true world. A P without a W subscript is used to denote the matrix based on our model of the world that we use to design the ILC law. Then the formulations connecting input to output presented in Chapter 2 is modified as

$$y_j(k) = CA^k x(0) + \sum_{i=0}^{k-1} CA^{k-i-1} B u_j(i) + v(k);\ \underline{y}_j = P_W \underline{u}_j + \overline{A} x(0) + \underline{v} \tag{5-1}$$

$$P_W = \begin{bmatrix} CB & 0 & 0 & \cdots & 0 \\ CAB & CB & 0 & \cdots & 0 \\ CA^2B & CAB & CB & \ddots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ CA^{p-1}B & CA^{p-2}B & CA^{p-3}B & \cdots & CB \end{bmatrix};\quad \overline{A} = \begin{bmatrix} CA \\ CA^2 \\ CA^3 \\ \vdots \\ CA^p \end{bmatrix} \tag{5-2}$$

The error history is given by

$$\underline{e}_j = -P_W \underline{u}_j + (\underline{y}^* - \overline{A} x(0) - \underline{v}) = -P_W \underline{u}_j + \underline{f} \tag{5-3}$$

where $\underline{y}^*$ means the desired output. Define a backward difference operator in iterations $\delta_j z(k) = z_j(k) - z_{j-1}(k)$ for any variable $z(k)$, and note that $\delta_j x(0) = 0$, $\delta_j v(k) = 0$, $\delta_j y^*(k) = 0$, $\delta_j \underline{f} = 0$. A general ILC learning law has the following form

$$\underline{u}_{j+1} = \underline{u}_j + L\underline{e}_j \quad \text{or} \quad \delta_{j+1}\underline{u} = L\underline{e}_j \tag{5-4}$$

with learning gain matrix L. The difference operator applied to Eq. (5-3) produces the equation for the error propagation from iteration to iteration

$$\delta_j \underline{e} = -P_W \delta_j \underline{u} \quad ; \; \underline{e}_{j+1} = (I - P_W L)\underline{e}_j \tag{5-5}$$

We examine 5 approaches to the problem of instability of inverse model, which is characterized by ill-conditioning of P_W. Note that P_W denotes the real world behavior as in Eq. (5-1), *P* denotes our model of P_W used to design the learning gain matrix *L*, and now we use P_i for *P* modified according to the approaches below. P_{Wi} is used when we consider that *L* uses the world model and approach *i*. Subscript *i* can be *a*, *b*, *c*, *d*, or *e*. Associated with each choice are values of $\underline{y}$, $\underline{y}^*$, $\underline{e}$, $\underline{f}$, which now have subscripts $\underline{y}_{i,\ell}, \underline{y}_i^*, \underline{e}_{i,\ell}, \underline{f}_i$ with the new subscript ℓ denoting the chosen learning law with gain matrix $L_{i,\ell}$. We will study the convergence robustness for each of these choices.

Approach (a): This first approach simply ignores the problem and uses matrix $P_a = P = USV^T$ in whatever learning law is chosen. Initially one would assume that this produces the unstable control action with exponentially growing error between samples. But in practice this approach can be practical when the bad singular values are much smaller than the smallest singular value related to frequency response, i.e. for reasonably fast sample rates. The error in the frequency response part of the space finishes convergence long before there is any significant progress learning the part of the space that creates the instability. Learning rates are treated below. Thus, after it appears to have converged, there is error at all time steps, and the user may be disappointed in the "final" error level. Continuing the iterations, it may take too many iterations to see the instability of the control action appear in practice, so the user never realizes there is an internal instability. It can also happen that the instability will never be observed, because the corrective action each iteration in that part of the error space is so small that it is beyond the last digit of the

digital converters. Then the learning has been stabilized by the quantization error in the digital control.

Approach (b): In order to ensure that the control instability is not observed in practice, one can guarantee stability by setting the bad singular values to zero to create P_b

$$P_b = \begin{bmatrix} U_{b1} & U_{b2} \end{bmatrix} \begin{bmatrix} S_b & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} V_{b1}^T \\ V_{b2}^T \end{bmatrix} \; ; \; P_b^{\dagger} = \begin{bmatrix} V_{b1} & V_{b2} \end{bmatrix} \begin{bmatrix} S_b^{-1} & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} U_{b1}^T \\ U_{b2}^t \end{bmatrix} \tag{5-6}$$

For the cases studied here, the bad singular values are sufficiently bad that there is no difference in the results when P_b is used in place of P in Approach (a). But note that if the sample time is made slow enough, all introduced zeros that are outside the unit circle eventually come inside, so slow sample rates can always allow one to experience the instability as the zeros get smaller in magnitude before entering the unit circle.

Here we define Approach (b) to consider use of an "inverse model" in ILC. A logical initial reaction to the ILC problem asks why don't you just find $\underline{u}^*$ by using model P in place of P_W. Of course, one cannot use the inverse, but one can use the pseudo-inverse also given in Eq. (10) to invert everything but the part that produces instability of the control action. The first application of this $P_b^{\dagger}$ will get zero error for this part of the space – provided $P = P_W$. Then we iterate with this same update aiming to eliminate the error remaining because our model is not perfect. ILC aims for zero error in the world, not zero error in our model of the world. This distinction is an important motivation for ensuring substantial convergence robustness to model error.

The next three approaches are what investigated before. We introduce extra sample time steps, either before the start of the desired trajectory or between the original sample times, and we do not

ask for zero error at these newly introduced steps. The column vectors $\underline{y}_{i,\ell}, \underline{y}_i^*, \underline{e}_{i,\ell}, \underline{f}_i$ are defined to only include the original sample times, but there are control actions at all time steps in $\underline{u}_{i,\ell}$. This makes P_i and $L_{i,\ell}$ rectangular, and the singular value decomposition of P_i for *i* equal *c*, *d*, or *e*, takes the form

$$P_i = U_i \begin{bmatrix} S_i & 0 \end{bmatrix} \begin{bmatrix} V_{i1}^T \\ V_{i2}^T \end{bmatrix} = U_i S_i V_{i1}^T \tag{5-7}$$

Approach (c): Initial deletion. One can always eliminate the bad singular values of *P* by eliminating the first few rows of *P*, the number of rows needed equals to non-minimum phase zeros. In the case of 3rd order pole excess as in Eq. (2-15), this means not asking for zero error for the first time step. Here we prefer to use the following approach, instead of eliminating any initial time step of the original desired trajectory, we add one time step to the beginning of the desired trajectory but do not ask for zero error for this step. Reference 4 motivates this by observing the process of solving for the control input history from the difference equation The extra time step is enough extra freedom to adjust the initial conditions to make the coefficient C_1 equal to zero in the solution of the homogeneous equation, which eliminates the instability. Note that the pseudo-inverse can make it acceptably small, rather than zero. One may be concerned that this might be an ill-conditioned process, which motivates the study of robustness for this approach as reported here.

Approach (d): Multiple zero order holds. We use a generalized hold, in place of the single control action for each time step of the original zero order hold. This eliminates the "anomalous singular values" discussed above, and hence eliminates the unstable growth of the error between addressed time steps. This approach has some extra freedom in the control action at the beginning

of the trajectory, but also has extra freedom in choice of control throughout the trajectory. One might consider that this extra freedom might improve robustness to model error in P, which is investigated here.

Based on our previous investigation, a list of conclusions is given as follows:

(1) The original bad singular values are gone.

(2) "New bad singular values" may be introduced.

(3) The "new bad singular values" have input and output singular vectors that both decay with time step. Hence, they do not produce unstable behavior between time steps – instability of the inverse problem is eliminated.

(4) The number of "new bad singular values" is smaller than the original bad singular values.

(5) The "new bad singular values" are not as bad as the original ones. Sometimes they are so close to the frequency response curve that they cannot be determined to be "bad" by looking at this curve. This caused Reference 4 to report one too few "new bad singular values" for pole excesses of 3, 7, and 11. To determine that a "new bad singular value" is in fact "bad", sometimes one must look at the associated singular vector to see that is represents a decay instead of a frequency.

(6) The above considers introducing one step between each addressed step. If one introduces enough extra steps, then the "new bad singular values" disappear, but this requires many extra time steps.

Approach (e): This approach combines Approaches (c) and (d), and is a modified version of multiple zero order holds approach. The numerical results reported here do the following. Start

with (d) and introduce multiple zero order holds between original time steps, then extend the desired trajectory as in (c) by the number of original time steps requested, equal to the number of zeros outside the unit circle. For the 3rd order system with one extra step at the beginning and between each addressed time step, at the faster sample rate, the first addressed time step is number 4. The original bad singular values are eliminated, and the new bad singular values are also gone.

An alternative is to start with (c) written for the faster sample rate, introduce the required extra initial step (s), and then apply (d) learning every second step starting at the original start time. This makes the first addressed fast time step number 3. For a 5th order system the two approaches happen to start at the same time step.

5.3 ILC Laws

Four ILC laws are used to evaluate stability robustness to model error for the above approaches. Reference 5 gives a unified formulation of various ILC laws, including the laws considered here. The control history $\underline{u}_{i,\ell}(j)$ for each iteration j contains a control action for all time steps, the original addressed time steps, and the non-addressed time steps, and it depends on the control law ℓ and the approach i. The corresponding error history vector only contains the error at the original addressed time steps, denoted $\underline{e}_{i,\ell}(j)$. The ILC update is then

$$\underline{u}_{i,\ell}(j+1) = \underline{u}_{i,\ell}(j) + L_{i,\ell}\underline{e}_{i,\ell}(j) \tag{5-8}$$

ILC Law (1): Pseudo-Inverse Law. As discussed above, this law considers using the inverse of the part of the system that is stably invertible (Approach (b)), and we include a scalar gain ϕ that can reduce the learning rate for possibly improved robustness. We can also consider using the pseudo-inverse from Approaches for i equal (c), (d), and (e)

$$L_{i,1} = \phi P_i^{\dagger} \tag{5-9}$$

ILC Law (2): Contraction Mapping or *P* Transpose Law (Reference 6)

$$L_{i,2} = \phi P_i^T \tag{5-10}$$

Note that this law has small gain and corresponding slow learning for singular values of P_i corresponding to the magnitude frequency response at high frequency.

ILC Law (3): Parial Isometry Law (Reference 7)

$$L_{i,3} = \phi V_{i1} U_i^T \tag{5-11}$$

This law can be obtained by replacing S_i with the identity matrix in P_i^T, so the gains related to high frequency in the learning gain matrix do not decay with frequency. But the system response is still small at high frequency. Reference 8 gives an alternative $\phi V_{i1} R U_i^T$ with *R* a diagonal matrix whose diagonal elements can tune the learning rate independently for each singular value or each "frequency". This is used in repetitive control in Reference 9 to improve robustness at frequencies where it is needed.

ILC Law (4): Quadratic Cost Law (References 10 and 11). The learning updates each iteration are tuned by minimizing a quadratic cost function with a parameter *r* that adjusts the speed of learning

$$J_{j+1} = \underline{e}_{i,\ell}^T(j+1)\underline{e}_{i,\ell}(j+1) + r\delta_{j+1}\underline{u}_{i,\ell}^T\delta_{j+1}\underline{u}_{i,\ell} \tag{5-12}$$

$$L_{i,4} = (P_i^T P_i + rI)^{-1} P_i^T \tag{5-13}$$

5.4 Convergence Analysis

5.4.1 Stability

To develop the stability criterion for the approaches and the ILC laws, we modify Eqs. (5-3) and (5-5) analogously to the modified Eq. (5-4) in Eq. (5-8) resulting in

$$\underline{e}_{i,\ell}(j) = -P_{Wi}\underline{u}_{i,\ell}(j) + \underline{f}_i \tag{5-14}$$

$$\underline{e}_{i,\ell}(j+1) = [I - P_{Wi}L_{i,\ell}]\underline{e}_{i,\ell}(j) \tag{5-15}$$

Because $\underline{e}_{i,\ell}(j)$ contains only addressed time steps, P_{Wi} is introduced which is P_W with all rows associated with unaddressed time steps deleted (later we will denote by $\hat{P}_{Wi}$ the rows associated with unaddressed time steps). From Eq. (5-15) the necessary and sufficient condition for $\underline{e}_{i,\ell}(j)$ to converge to zero tracking error as $j \to \infty$ for all possible initial errors on the initial iteration is that the spectral radius be less than unity, i.e. the eigenvalue with maximum magnitude must have magnitude less than unity

$$\max_m \left|\lambda_m(I - P_{Wi}L_{i,\ell})\right| < 1 \tag{5-16}$$

In practice, satisfying this condition ensures convergence, but the transients during the learning process can be poor. One can ensure that the root mean square (RMS) of the tracking error (or equivalently, the Euclidean norm of the error history) decays monotonically if the maximum singular value is less than unity

$$\max_m \sigma_m(I - P_{Wi}L_{i,\ell}) < 1 \tag{5-17}$$

5.4.2 Convergence Rate and Final Value of the Error at Addressed Time Steps

This section considers that our model *P* and the real world P_W are the same, and studies the rate of convergence and final error level.

Approach (a): First examine Approach (a) which simply ignores the issue of instability of the inverse problem, and analyze the behavior (Reference 1). Pick ILC Law (2) and substitute the singular value decompositions of $P_W = P$ into Eq. (5-15) to obtain

$$U^T \underline{e}_{a,2}(j+1) = [I - \phi S_a] U^T \underline{e}_{a,2}(j) \tag{5-18}$$

Provided ϕ is chosen small enough to satisfy $\left|1 - \phi\sigma_m^2\right| < 1$ for all singular values, the error will converge monotonically to zero for all time steps as $j \to \infty$, and the magnitude of the component of the error vector on the m^{th} row of U^T will decrease by $\left|1 - \phi\sigma_m^2\right|$ every iteration. Now suppose that there is only one bad singular value σ_p, and the last singular value related to frequency response is then σ_{p-1}. Let the associated error components on the last two rows of U^T be denoted by $\varepsilon_p(j)$ and $\varepsilon_{p-1}(j)$. Then

$$\begin{aligned} \varepsilon_{p-1}(j) &= [1 - \phi\sigma_{p-1}^2]^j \varepsilon_{p-1}(0) = [I - j\phi\sigma_{p-1}^2 + \cdots]\varepsilon_{p-1}(0) \\ \varepsilon_p(j) &= [1 - \phi\sigma_p^2]^j \varepsilon_p(0) = [I - j\phi\sigma_p^2 + \cdots]\varepsilon_p(0) \end{aligned} \tag{5-19}$$

The next terms in the sequences contain σ_{p-1}^4 and σ_p^4. Suppose that $\phi = 1$ and that the magnitude frequency response at the highest frequency visible below Nyquist frequency is 1/100 so that $\sigma_{p-1} \approx 1/100$, and that the bad singular value is one order of magnitude smaller. One can make an estimate of how many iterations would be needed to get the error to essentially zero by

looking at the 2 terms shown in the square bracket. Error $\varepsilon_{p-1}(j)$ could be essentially zero after 1000 iterations, and $\varepsilon_p(j)$ could be essentially zero after 10,000 iterations. So it is clear that one would observe what looks like the iterations have finished converging by 1000 iterations, and it would take several more 1000 iterations before the instability in the control action would become evident. Now consider the bad singular value as reported above for the 3rd order system, where Matlab inaccurately computes this value to be 10^{-19} while the actual value is 10^{-50} as investigated in Chapter 2. The same estimate of the number of iterations needed asks for $j = (10^{19})^2$ or $(10^{+50})^2$ iterations, and one would not see this part converge during one's lifetime. Hence, often in practice, one never experiences the instability of the inverse problem, and instead suffers from a disappointing "final" error level. Actually, the hardware will likely not be able to learn this part of the error space because the corrective updates each iteration are so small that they cannot be seen with the number of digits in the digital converters – they do not have 38 or 100 digit accuracy. ILC Law (4) will have essentially the same estimates for convergence time, but the partial isometry learns faster because the σ_{p-1}^2 and σ_p^2 are replaced by σ_{p-1} and σ_p as discussed below.

Approach (b): Use $P = P_W$, create P_b by setting the bad singular values to zero. Then substitute into Eq. (5-15) to obtain

$$U^T \underline{e}_{b,1}(j+1) = \begin{bmatrix} I - \phi I & 0 \\ 0 & I \end{bmatrix} U^T \underline{e}_{b,1}(j) \tag{5-20}$$

For ϕ less than unity in magnitude the final error components related to the good singular values converge to zero error. Again consider one bad singular value, and the associated error component remains unchanged throughout the learning process, $\varepsilon_p(j+1) = \varepsilon_p(j) = \varepsilon_p(0)$. We can compute the final error level

$$\underline{e}_{b,1}(\infty) = U\begin{bmatrix} 0 & \cdots & 0 & \varepsilon_p(\infty) \end{bmatrix} = U_p U_p^T \underline{e}_{b,1}(0) \tag{5-21}$$

Here we locally use the symbol U_p^T to represent the last row of U^T, and U_p to be the last column of U. Note that $U_p U_p^T$ is an outer product so generally there will be tracking error at all time steps. Because of the decaying nature of this last row with time step, the majority of the error is at the beginning of the trajectory.

Observe that the disappointing "final error" level described above for Approach (a), is the same as the final error level computed here, based on the concept that "final error" corresponds to all of the error eliminated except for the error in the "bad singular value" part of the error space, which has made no progress learning at the iterations that seem to have finished learning.

Approaches (c), (d), (e): These approaches converge to zero error at the addressed time steps for appropriately chosen ϕ. The convergence rate for each component of the error on the orthogonal unit vectors of U_i are seen from

$$U_i^T \underline{e}_{i,\ell}(j+1) = diag(\alpha_1, \alpha_2, ..., \alpha_p) U_i^T \underline{e}_{i,\ell}(j) \tag{5-22}$$

where *diag* represents a diagonal matrix whose entries are given by

ILC Law (2): $\alpha_m = 1 - \phi\sigma_m^2$

ILC Law (3): $\alpha_m = 1 - \phi\sigma_m$

ILC Law (4): $\alpha_m = r / (r + \sigma_m^2)$

Associated with Law (d) was the alternative $\phi V_{i1} R U_i^T$ with $R = diag(r_1, r_2, ..., r_p)$ whose learning rate can be adjusted individually for each error component, according to $\alpha_m = 1 - \phi \sigma_m r_m$ by choice of the values of r_m.

5.4.3 Final Value of the Error at Unaddressed Time Steps

Approaches (c), (d), and (e): Although these approaches converge to zero tracking error for the addressed time steps, they have introduced extra sample times without asking for zero error at these new steps. It is of interest to know what the error is at the newly introduced times, also called the unaddressed time steps, generalizing Reference 1. Starting with the P_W containing all time steps, separate the rows putting the rows associated with the addressed time steps for the chosen approach first P_{Wi}, and then below that collect all rows for the unaddressed steps in order, denoting this matrix as $\hat{P}_{Wi}$. Similarly, maintain the existing notation $\underline{e}_{i,\ell}(j)$ and $\underline{f}_i$ for the addressed time steps, and introduce $\hat{e}_{i,\ell}(j)$ and $\hat{f}_i$ for the unaddressed time steps. Equations (5-3), (5-4), and (5-5) become

$$\begin{bmatrix} \underline{e}_{i,\ell}(j) \\ \hat{e}_{i,\ell}(j) \end{bmatrix} = -\begin{bmatrix} P_{Wi} \\ \hat{P}_{Wi} \end{bmatrix} \underline{u}_{i.\ell}(j) + \begin{bmatrix} \underline{f}_i \\ \hat{f}_i \end{bmatrix} \quad ; \begin{bmatrix} \underline{e}_{i,\ell}(j+1) \\ \hat{e}_{i,\ell}(j+1) \end{bmatrix} = \begin{bmatrix} \underline{e}_{i,\ell}(j) - P_{Wi} L_{i,\ell} \underline{e}_{i,\ell}(j) \\ \hat{e}_{i,\ell}(j) - \hat{P}_{Wi} L_{i,\ell} \underline{e}_{i,\ell}(j) \end{bmatrix} \tag{5-23}$$

$$\delta_{j+1} \underline{u}_{i.\ell} = L_{i,\ell} \underline{e}_{i,\ell}(j)$$

Define $Q = I - P_{Wi} L_{i,\ell}$ and then from the upper partition $\underline{e}_{i,\ell}(j+1) = Q \underline{e}_{i,\ell}(j)$ and $\underline{e}_{i,\ell}(j) = Q^j \underline{e}_{i,\ell}(0)$. The error at the addressed time steps converges to zero error for all three approaches, for appropriately chosen gain ϕ. The second partition produces the following error at the unaddressed time steps at iteration *j*

$$\hat{e}_{i,\ell}(j) = \hat{e}_{i,\ell}(0) - \hat{P}_{Wi} L_{i,\ell} [I + Q + Q^2 + \cdots + Q^{j-1}] \underline{e}_{i,\ell}(0) \tag{5-24}$$

The sum in square brackets equals $(I-Q)^{-1}(I-Q^j)$, and $Q^j \to 0$ as $j \to \infty$, and $I - Q = P_{Wi} L_{i,\ell}$. Therefore the final error level at the introduced unaddressed time steps is given by

$$\hat{e}_{i,\ell}(\infty) = \hat{e}_{i,\ell}(0) - \hat{P}_{Wi} L_{i,\ell} (P_{Wi} L_{i,\ell})^{-1} \underline{e}_{i,\ell}(0) \tag{5-25}$$

and the final control action satisfies

$$\begin{bmatrix} P_{Wi} \\ \hat{P}_{Wi} \end{bmatrix} \underline{u}_{i.\ell}(\infty) = \begin{bmatrix} \underline{f}_i \\ \hat{f} \end{bmatrix} - \begin{bmatrix} 0 \\ \hat{e}_{i,\ell}(\infty) \end{bmatrix} \tag{5-26}$$

Denote the singular value decompositions of P_{Wi} and $\hat{P}_{Wi}$ as

$$P_{Wi} = U_{Wi} \begin{bmatrix} S_{Wi} & 0 \end{bmatrix} \begin{bmatrix} V_{Wi,1}^T \\ V_{Wi,2}^T \end{bmatrix} \quad ; \; \hat{P}_{Wi} = \hat{U}_{Wi} \begin{bmatrix} \hat{S}_{Wi} & 0 \end{bmatrix} \begin{bmatrix} \hat{V}_{Wi,1}^T \\ \hat{V}_{Wi,2}^T \end{bmatrix} \tag{5-27}$$

From the upper partition of Eq. (5-26) one obtains $V_{Wi,1}^T \underline{u}_{i.\ell}(\infty) = S_{Wi}^{-1} U_{Wi}^T \underline{f}_i$ which leaves the components of $\underline{u}_{i.\ell}(\infty)$ on $V_{Wi,2}^T$ yet to be determined. Denote these components as $V_{Wi,2}^T \underline{u}_{i.\ell}(\infty) = \gamma$. Then the lower partition can be written as $\hat{P}_{Wi} V_{Wi} V_{Wi}^T \underline{u}_{i.\ell}(\infty) = \hat{f}_i - \hat{e}_{i,\ell}(\infty)$, which after substituting the components and using the decomposition of $\hat{P}_{Wi}$ produces the equations for γ and $\underline{u}_{i.\ell}(\infty)$

$$[\hat{V}_{Wi,1}^T V_{Wi,2}] \gamma = \hat{S}_{Wi}^{-1} \hat{U}_{Wi}^T [\hat{f}_i - \hat{e}_{i,\ell}(\infty)] - \hat{V}_{Wi,1}^T V_{Wi,1} S_{Wi}^{-1} U_{Wi}^T \underline{f}_i \quad ; \; \underline{u}_{i.\ell}(\infty) = V_{Wi} \begin{bmatrix} S_{Wi}^{-1} U_{Wi}^T \underline{f}_i \\ \gamma \end{bmatrix} \tag{5-28}$$

Equations (5-25) and (5-28) give the final value of the error at unaddressed points and the final value of all control actions for any learning law $L_{i,\ell}$ using approach *i*. In particular, it applies when

the model used in the ILC law has model error compared to the true world model P_W. If there is no such model error, we can further develop these equations. Examine $L_{i,\ell}(P_{Wi}L_{i,\ell})^{-1}$ for ILC Laws (2), (3), and (4), including a subscript W on all appropriate matrices in Eqs. (5-10) through (5-13) that now are based on the true world model. After some algebra, the result is the same in all cases

$$L_{i,\ell}(P_{Wi}L_{i,\ell})^{-1} = P_{Wi}^{\dagger} \quad \text{for} \quad i = 2,3,4 \tag{5-29}$$

Use this result in Eq. (5-25), and $\hat{e}_{i,\ell}(0) = -\hat{P}_{Wi}\underline{u}_{i.\ell}(0) + \hat{f}_i$, and $\underline{e}_{i,\ell}(0) = -P_{Wi}\underline{u}_{i.\ell}(0) + \underline{f}_i$, from Eq. (5-23) to obtain

$$\begin{aligned}
\hat{e}_{i,\ell}(\infty) &= [\hat{f}_i - \hat{P}_{Wi}P_{Wi}^{\dagger}\underline{f}_i] + \hat{P}_{Wi}[P_{Wi}^{\dagger}P_{Wi} - I]\underline{u}_{i,\ell}(0) \\
\hat{e}_{i,\ell}(\infty) &= [\hat{f}_i - \hat{P}_{Wi}P_{Wi}^{\dagger}\underline{f}_i] + \hat{P}_{Wi}[V_{Wi,1}V_{Wi,1}^T - I]\underline{u}_{i,\ell}(0) \\
\hat{e}_{i,\ell}(\infty) &= [\hat{f}_i - \hat{P}_{Wi}P_{Wi}^{\dagger}\underline{f}_i] + \hat{P}_{Wi}V_{Wi,2}V_{Wi,2}^T\underline{u}_{i,\ell}(0)
\end{aligned} \tag{5-30}$$

The final value of the error at unaddressed points is of course dependent on the Approach used, since the Approach defines what steps are unaddressed. But Eq. (5-30) says $\hat{e}_{i,\ell}(\infty)$:

(i) Is independent of the ILC Law because $\underline{u}_{i,\ell}(0)$ is the command chosen for the first run, and not dependent on the law used. It serves as the initial condition for the ILC iterations.

(ii) If one decides to use zero input on the first iteration, then the final value of the error at unaddressed time steps is not a function of the error in the first iteration. It is a function of the initial condition, the desired trajectory, and the repeating disturbance – for both unaddressed and addressed time steps.

Now consider the value of γ in Eq. (5-28), substituting Eq. (5-30) for $\hat{e}_{i,\ell}(\infty)$. After some matrix algebra, substituting the first term on the right in Eq. (5-30) into the right hand side of Eq. (5-28) gives zero, and the second term produces

$$\begin{aligned} \gamma &= V_{Wi,2}^T \underline{u}_{i,\ell}(0) \\ \underline{u}_{i,\ell}(\infty) &= P_{Wi}^{\dagger} \underline{f}_i + V_{Wi,2} V_{Wi,2}^T \underline{u}_{i,\ell}(0) \end{aligned} \tag{5-31}$$

If one again decides to use zero control input $\underline{u}_{i,\ell}(0) = \underline{0}$ for the initial run, then $\gamma = 0$. The upper partition in Eq. (5-26), $P_{Wi} \underline{u}_{i.\ell}(\infty) = \underline{f}_i$, has many solutions, all of which produce zero error at the addressed time steps. If $\gamma = 0$, then $\underline{u}_{i.\ell}(\infty)$ is that solution of this equation having minimum Euclidean norm, i.e. $\underline{u}_{i.\ell}(\infty) = P_{Wi}^{\dagger} \underline{f}_i$. Note that this final control action is independent of which ILC Law one uses. Of course, one possible solutions is that solution obtained by using the full matrix inverse P_W^{-1} which usually gives an exponentially growing control action. Introducing new unaddressed time steps produces instead a well behaved final control history.

5.5 Approach to Evaluating Robustness

We wish to evaluate stability robustness for each of the approaches, and each of the learning laws. In ILC it is particularly important to have stability robustness to model errors because ILC asks to converge to zero error in the real world, not in our model of the world. To illustrate this distinction, we inverted a model of the feedback control systems on the Robotics Research Corporation robot reported in Reference 12 and obtained an RMS tracking error reduction of a factor of 50, but turning on an ILC law reduced the error by a total factor of 1000. Hence, we want convergence for the largest possible range of models near the nominal model. One can define two classes of robustness, robustness to parameter variation, and robustness to unmodeled high

frequency dynamics, also called residual modes or parasitic poles. Robustness of the latter kind is addressed by using a zero-phase low-pass filter to cut off the learning, and tuning the cutoff frequency in hardware based on observed performance (References 12). Here, we examine the approaches and the learning laws, asking to maximize robustness to parameter variation about the nominal. The following systems are considered

$$G_3(s)=\left(\frac{a}{s+a}\right)\left(\frac{\omega_1^{\,2}}{s^2+2\zeta_1\omega_1 s+\omega_1^{\,2}}\right)\quad ;\; G_4(s)=\left(\frac{\omega_1^{\,2}}{s^2+2\zeta_1\omega_1 s+\omega_1^{\,2}}\right)\left(\frac{\omega_2^{\,2}}{s^2+2\zeta_2\omega_2 s+\omega_2^{\,2}}\right) \qquad (5\text{-}32)$$

$$G_5(s)=\left(\frac{a}{s+a}\right)\left(\frac{\omega_1^{\,2}}{s^2+2\zeta_1\omega_1 s+\omega_1^{\,2}}\right)\left(\frac{\omega_2^{\,2}}{s^2+2\zeta_2\omega_2 s+\omega_2^{\,2}}\right)$$

Note that the subscript denotes the pole excess, indicating that when fed by a zero order hold using the sample rates considered, that there is one zero introduced outside the unit circle for $G_3(s)$ and $G_4(s)$, and two for $G_5(s)$. The nominal values of the constants are $a=8.8$, ω_1 and ω_2 are 37, 125.7 rad/sec, approximately 6, 20 Hz. For odd order systems, we use $\zeta=0.1$. The even order system has a resonant peak without the benefit of the attenuation from the first order term, and ζ is reduced to 0.4 to reduce sensitivity. Initially, consider that these Laplace transfer functions are fed by a zero order hold sampling at 100 Hz, and that the number of time steps in the desired trajectory is $p=100$. Then, for Approach (c) an appropriate number of additional time steps is appended to the start of the trajectory making a 101 by 101 matrix P_W for 3rd and 4th order systems and 102 by 102 for the 5th order system. For Approach (d) either one or two additional time steps are inserted between every original pair of time steps, and between time step zero and the first time step in the desired trajectory, producing a 200 by 200 matrix, or a 300 by 300 matrix. Approach (e) as used here, first modifies according to (d), and then augments the number of time steps at the

original sample interval, 202 by 202 for 3rd and 4th order models. Then delete rows of P_W to produce P_{Wi}.

The procedure for studying robustness follows the method used in Reference 13. Consider that the uncertainty in each of the parameters in the model obey a uniform distribution ranging ±25% around the nominal values, i.e.

$$\begin{gathered} 27.75 < \omega_1 < 46.25 \; ; \; 219.975 < \omega_2 < 157.125 \; ; \; 6.6 < a < 11 \\ 0.075 < \zeta_{odd} < 0.125 \; ; \; 0.3 < \zeta_{even} < 0.5 \end{gathered} \tag{5-33}$$

We randomly pick 3 sets of 400 samples from each distribution to form 3 sets of 400 models. Comparing the results for each set gives understanding of how much influence the random sampling has on the result. Then using Eqs. (5-16) and (5-17) we evaluate how many of the sample models are stable for each run, and how many have monotonic decay of the Euclidean norm of the error each iteration, respectively. From the learning rate information presented above, it is clear that for some ILC laws the learning rate becomes very slow for particularly high frequencies. This means that the eigenvalues in Eq. (5-16) can be very near unity, and numerical round off introduces uncertainty as to whether an eigenvalue is less than or greater than unity in magnitude. Thus, the number of models producing a maximum magnitude of eigenvalues (spectral radius) greater than unity is given, but also the numbers for $|\lambda| > 1.001$ and $|\lambda| > 1.01$. Corresponding results are given for the maximum singular value.

5.6 Robustness Comparisons Using Unity Gain ϕ

This section compares the robustness of the different approaches using ILC Law (2) with unity gain, $\phi = 1$. Table 5-1 presents the results which can be summarized as follows.

(i) The first rows of the Table use ILC Law (2) ignoring any issue with instability of the inverse system (Approach (a)). Observe that 242, 256, and 239 of the 400 random model choices in the 3 runs are unstable according to the computation with $|\lambda| > 1$. Also, essentially all models violated the monotonic decay condition Eq. (5-17). For this system with pole excess of 3, there is one bad singular value introduced by the discretization. The final column of the table tries to say how much smaller this singular value is, 2.3284×10^{-58}, than the smallest singular value related to frequency response, 5.0838×10^{-4}. One must use special techniques (Reference 1) to estimate this bad singular value since Matlab cannot produce the correct answer for a 100 by 100 matrix.

Table 5-1. Contraction Mapping ILC Robustness ($L_{i,2} = P_i^T$, $G_3(s)$)

Contraction Mapping Law		Max. Eigenvalues			Max. Singular Values			Bad singular values	
		>1	>1.001	>1.01	>1	>1.001	>1.01	Number	Deviation
(A) Full *P* matrix (a), (b)	1st run	242	83	82	399	121	114	1	-100%
	2nd run	256	78	73	398	115	111		
	3rd run	239	85	79	400	133	125		
(B) Append initial row(s) (c)	1st run	84	84	79	112	111	104	0	N/A
	2nd run	89	88	87	126	122	115		
	3rd run	77	76	71	119	114	109		
(C) Every other row (d)	1st run	31	31	23	66	65	54	1	-7.58%
	2nd run	23	22	15	68	64	53		
	3rd run	37	35	28	66	64	49		
(D) Every third row (d)	1st run	15	13	6	49	49	34	1	-3.28%
	2nd run	17	16	5	52	49	38		
	3rd run	15	13	6	49	49	34		
(E) Every other row then append (e)	1st run	29	28	23	61	55	49	0	N/A
	2nd run	26	26	21	59	56	45		
	3rd run	34	34	19	67	65	58		

(ii) If Approach (b) were used, replacing this singular value by zero, the results would be identical 10^{-58} is certainly a numerical zero for this computation.

(iii) The second row partition of the Table shows what happens when the desired trajectory has one new time step appended at the beginning, so there are 101 control actions, but only 100 time steps that aim for zero error during the learning process (Approach (c)). This eliminates the one bad singular value. And it eliminates almost all of the systems that were previously unstable according to the numerical computation, with $1<|\lambda|<1.001$. The results for the singular value columns are very similar. We conclude that Approach (c) significantly improves robustness, making many systems that were questionably unstable become clearly stable. The number of more clearly unstable cases with $|\lambda|>1.001$ remains about the same in Approach (a) and (c).

(iv) Approach (d) in the 3rd row partition, inserts extra time steps between the original 100 steps, using 200 control actions whose aim is to produce zero error at the 100 original sample times. There is a substantial improvement over Approach (c), with many fewer models that are clearly unstable, and again nearly no models in the range $1<|\lambda|<1.001$, making this approach very effective at robustication. The 4^{th} row partition, instead of inserting one extra sample time between each pair of original time steps, inserts 2 extra sample times. Again there is significant improvement in robustness. One can get further improvement introducing more sample times between, but with diminishing benefits. The disadvantage of Approach (d) is that there is a new "bad singular value" introduced. The last column of the table says that the introduced singular value is 7.58% and 3.28% smaller than the smallest singular value related to frequency response. These numbers are not extreme like the original bad singular value. Furthermore, there is no instability in the inverse model because both the associated input and output singular vectors decay with time.

(v) The final row partition gives Approach (e) with one time step introduced between original sample times, and in addition appending one slow time step, or 2 fast time steps, to the beginning of the desired trajectory. This Approach (e) is substantially better than Approach (c), and roughly equivalent to Approach (d) that introduced one extra step between original addressed steps, but has the advantage that the new "bad singular value" is no longer present.

(vi) Table 5-1 presents results for ILC Law (2). Results for ILC Laws (3) and (4) exhibit all of the same trends.

5.7 Singular Value Comparisons for Different Approaches

Figure 5-1 compares the singular values of P_i for the various approaches *i*, for the 3rd order pole excess and 5th order pole excess models, using originally 100Hz sample rate, and 200Hz when samples are added between addressed steps in Approaches (d) and (e). A singular value decomposition routine will deliver singular values in descending order. A careful way to match them to frequency response is to do a discrete Fourier transform of the associated singular vectors, but an easy approach used here is to sort the frequency response in descending order to match singular values to frequencies. To keep the scale reasonable, the bad singular values in P_a, and the new bad singular values in P_d are not plotted. Except for this, the singular values for both P_a, P_c are the same, and the singular values for P_d, P_e are the same. Including the extra time steps before the start of the original trajectory eliminates the bad singular values, and leaves the remaining singular values related to frequency the same. To better see how the singular values differ, Figure 5-2 plots the solid line minus the dashed line. Observe that when one sample point is introduced between each of the original sample times, the first singular value for 3rd order, 5th order pole excess, as well as 4th order in Figure 5-3, is decreased by 3dB. We call this original singular value

the "DC gain", since as the trajectory gets long enough, it converges to the DC gain (the associated singular vector does not look much like a constant for the 100 time steps used here). The decrease by 3dB is similar for all frequencies, but it is not uniform with frequency. For the 5th order system, the deviations are bigger near the natural frequencies. Figure 5-3 gives corresponding plots for the 4th order pole excess model, and one observes that approaching Nyquist frequency what was 3dB attenuation becomes 4dB amplification. It is interesting to note that if instead of doubling the sample rate to address every other fast rate time step, one uses 400Hz and address every fourth, the DC gain is reduced by 6dB, and similarly when addressing every eighth time steps, the DC gain is reduced by 9dB.

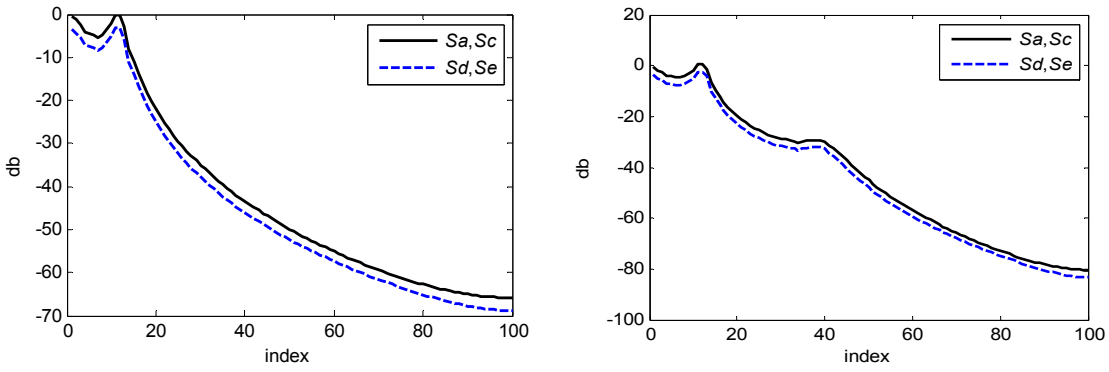

Figure 5-1. Singular values of P_a, P_c (solid line) and P_d, P_e (dashed line) for $G_3(s)$ (left) and $G_5(s)$ (right).

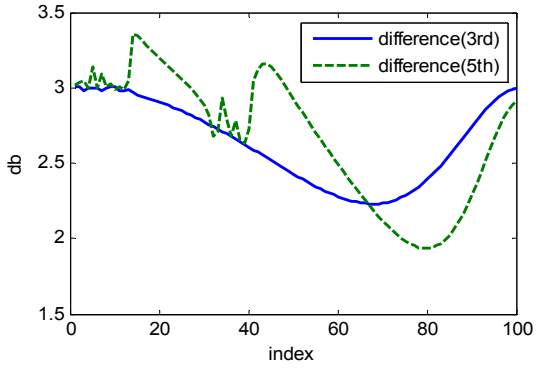

Figure 5-2. Solid line minus dashed line from Figure 1.

Robustness is usually improved when the ILC Law learns more slowly. The 3dB attenuation is not uniform at all frequencies, and in the case of the 4th order system there is even amplification that is larger than the attenuation, but we can make a robustness comparison for the different approaches and different ILC Laws setting gain ϕ to make the learning rate match at DC, where the rate is given by α_1 in Eq. (5-22). This is done as follows: (1) For P_a, P_c the ϕ is set to unity as before, and for the quadratic cost ILC Law (4) r is set to unity. (2) For P_d, P_e the 3dB decrease implies that the new DC singular value equals $(1/1.41)$ time the old, $\sigma_{1new} = (1/1.41)\sigma_{1old}$. In order to match the learning rate for ILC Laws (2), (3), and (4):

ILC Law (2): Match $\phi_{new}\sigma_{1new}^2 = \phi_{old}\sigma_{1old}^2$. Since $\phi_{old} = 1$, pick $\phi_{new} = (1.41)^2 \approx 2$.

ILC Law (3): Match $\phi_{new}\sigma_{1new} = \phi_{old}\sigma_{1old}$, which requires $\phi_{new} = 1.41$.

ILC Law (4): Match $r_{new}/(r_{new} + \sigma_{1new}^2) = r_{old}/(r_{old} + \sigma_{1old}^2)$. Since $r_{old} = 1$, $r_{new} = (1.41)^2 \approx 2$.

The next section performs the robustness comparisons of the approaches and the ILC Laws using these values.

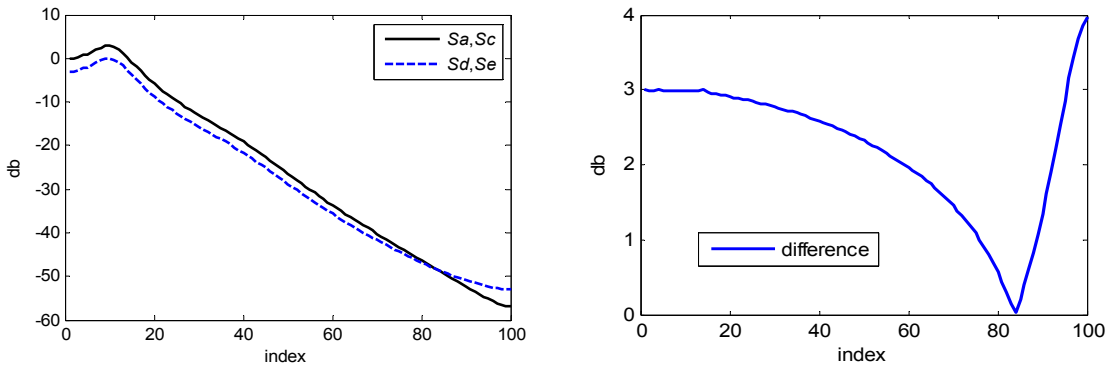

Figure 5-3. Plot as in Figure 1 (left), and as in Figure 2 (right, plotting the absolute value of the difference) for $G_4(s)$

5.8 Robustness Comparison with Matched "DC Gain"

Table 5-2 presents the robustness results for the 3rd order pole excess system $G_3(s)$. In this comparison Run 1 uses the same random samples for all ILC Laws, and similarly for Runs 2 and 3, so that comparison between control laws is more meaningful. Each entry on the table gives the results for ILC Laws 2, 3, and 4 in order. The row partitions (A), (B), (C), and (E) correspond to the row partitions in Table 5-1. Examining the first entries, for contraction mapping $L_{i,2}$, (A) has more than half of all models in the 400 model sets for Runs 1, 2, and 3 with largest eigenvalue magnitude between $1 < |\lambda| < 1.001$ and all models with maximum singular value $1 < \max \sigma < 1.001$. All three Approaches (c), (d), and (e) eliminate almost all of these, and leave the numbers above 1.001 approximately unchanged. These same statements also apply to the Partial Isometry Law and the Quadratic Cost Law, $L_{i,3}, L_{i,4}$. The Partial Isometry Law is expected to be the least robust because it learns much faster at high frequencies, and the results are consistent with this expectation. For ILC Law (2) there are roughly 70 unstable models for each run, compared to roughly 90 for ILC Law (3). Note that (A) shows all 400 models for all 3 runs have maximum singular values greater than unity. The Quadratic Cost Law is substantially more robust than the others, with less than 20 unstable models for each run using Approaches (c), (d), and (e). One would prefer (e) to (d) because of the "new bad singular values" that will have slow learning in some part of the error space. In Table 5-2 Approaches (c) and (e) now have very similar performance, with the advantage of (e) seen in Table 5-1 no longer evident. In addition to the extra robustness in the Quadratic Cost Law, it is shown in Chapter 3 that one can modify the law with a quadratic cost on the unaddressed time steps to have direct control over the error between original time steps. Table 5-3 gives the corresponding results for the 5th order pole excess system, which has two zeros outside the unit circle. All of the trends are similar.

Table 5-2. Robustness of ILC Laws $L_{i,2}, L_{i,3}, L_{i,4}$ for $G_3(s)$

		Max. Eigenvalues			Max. Singular Values		
		>1	>1.001	>1.01	>1	>1.001	>1.01
(A)	Run 1	248/261/207	72/91/17	68/83/12	398/400/397	113/151/65	108/148/56
	2	224/279/201	69/94/17	66/88/13	399/400/400	120/144/61	109/136/52
	3	228/264/186	67/86/19	60/81/15	398/400/400	109/142/62	101/138/54
(B)	Run 1	74/91/18	72/91/17	68/83/12	114/151/65	113/151/65	108/148/56
	2	71/94/18	69/94/17	66/88/13	123/144/61	120/144/61	109/136/52
	3	68/89/19	67/86/19	60/81/15	112/142/68	109/142/62	101/138/54
(C)	Run 1	76/91/18	75/91/17	69/83/12	114/151/66	113/151/65	111/148/57
	2	72/94/18	71/94/17	67/89/14	123/144/62	121/144/61	110/137/52
	3	69/91/19	69/89/19	62/81/16	113/143/68	111/142/63	105/138/54
(E)	Run 1	76/91/18	75/91/17	69/83/12	114/152/66	113/151/65	111/149/57
	2	72/94/18	71/94/17	67/89/14	123/144/62	121/144/61	110/137/52
	3	69/91/19	69/89/19	62/81/16	113/143/68	111/142/63	105/138/54

Table 5-3. Robustness of ILC Laws $L_{i,2}, L_{i,3}, L_{i,4}$ for $G_5(s)$

		Max. Eigenvalues			Max. Singular Values		
		>1	>1.001	>1.01	>1	>1.001	>1.01
(A)	Run 1	322/345/294	108/120/14	104/107/12	400/400/400	147/198/74	138/177/64
	2	318/339/300	98/122/22	92/107/16	400/400/400	130/182/79	124/150/66
	3	338/331/298	121/110/18	118/95/13	400/400/400	163/196/71	157/162/64
(B)	Run 1	110/130/14	108/120/14	104/107/12	199/213/128	147/198/74	138/177/64
	2	98/130/23	98/122/22	92/107/16	184/191/123	130/182/79	124/150/66
	3	122/116/18	121/110/18	118/95/13	210/200/131	163/196/71	157/162/64
(C)	Run 1	111/132/14	111/120/14	105/107/12	199/213/129	148/199/74	140/178/64
	2	99/130/25	98/123/22	93/109/16	184/193/124	134/184/79	126/151/67
	3	125/117/18	124/111/18	119/95/13	210/200/131	164/196/72	158/165/64
(E)	Run 1	111/132/14	111/120/14	105/107/12	199/213/129	148/199/74	140/178/64
	2	99/130/25	98/123/22	93/109/16	184/193/124	134/184/79	126/151/67
	3	125/11718	124/111/18	119/95/13	210/200/131	164/196/72	158/165/64

The fourth order pole excess model $G_4(s)$ results are shown in Table 5-4. Recall that difficulty in stabilizing even order pole excess resulted in using a damping ration of $\zeta = 0.4$ instead of 0.1. The first order term starts the decay of the magnitude plot before the first resonance in the odd order pole excess models, but there is a substantial resonant peak for 0.1 which likely increases

sensitivity. The zero in the discrete time transfer function that approaches -1 as the sample time interval approaches zero, also influences behavior near Nyquist frequency. Table 5-3 shows that the Partial Isometry Law, $L_{i,3}$, does surprisingly well compared to $L_{i,2}$, for Approaches (c), (d), and (e), with less than 10 models being unstable. And the Quadratic Cost Law $L_{i,4}$ actually stabilizes all models.

Table 5-4. Robustness of ILC Laws $L_{i,2}, L_{i,3}, L_{i,4}$ for $G_4(s)$

		Max. Eigenvalues			Max. Singular Values		
		>1	>1.001	>1.01	>1	>1.001	>1.01
(A)	Run 1	288/217/205	179/3/0	175/2/0	400/400/400	197/6/0	194/4/0
	2	283/236/203	179/7/0	173/6/0	400/400/400	197/8/0	189/7/0
	3	288/208/208	192/4/0	182/3/0	400/400/400	218/5/0	214/5/0
(B)	Run 1	180/3/0	179/3/0	175/2/0	198/6/0	197/6/0	194/4/0
	2	179/7/0	179/7/0	173/6/0	198/8/0	197/8/0	189/7/0
	3	192/4/0	192/4/0	182/3/0	219/5/0	218/5/0	214/5/0
(C)	Run 1	180/3/0	179/3/0	175/2/0	198/6/0	197/6/0	194/4/0
	2	179/7/0	179/7/0	173/6/0	198/8/0	197/8/0	190/7/0
	3	192/4/0	192/4/0	182/3/0	219/5/0	218/5/0	214/5/0
(E)	Run 1	180/3/0	179/3/0	175/2/0	198/6/0	197/6/0	194/4/0
	2	179/7/0	179/7/0	173/6/0	198/8/0	197/8/0	190/7/0
	3	192/4/0	192/4/0	182/3/0	219/5/0	218/5/0	214/5/0

5.9 Comment on Use of the Pseudo-Inverse as a Learning Control Law

One cannot use the inverse of one's model because of the bad singular value, but one might consider Approach (b) that simply removes that singular value, and finds the pseudo-inverse of the resulting matrix. Then one might try to continue using this update action as a learning control law (one could introduce a gain in front to slow the learning rate). Of course, this learning law will not be able to learn the part of the error space that was eliminated, but it might converge to zero error otherwise. Alternatively, Approaches (c) and (e) offer the possibility of stable inverses, and zero error at addressed steps. One can consider using these pseudo-inverses as ILC laws for updates

each iteration. Table 5-5 presents the stability robustness results for these candidate ILC Laws. Approach (b) that simply eliminates the bad singular value part of the update space has many models that are marginally unstable with $1<|\lambda|<1.001$, and not so many that are clearly unstable with $|\lambda|>1.001$, but other methods discussed above are certainly superior. Perhaps the biggest concern is the fact that monotonic convergence is not assured for such a large percent of all the models. The number of singular values above unity, suggests that one does not want to use the pseudo-inverse as a learning law, even with the benefit of these two approaches.

5.10 Error and Control Action after Convergence

The instability of the inverse model results in: (1) zero error at the addressed time steps, (2) exponential growth of the magnitude of the control action, and (3) exponential growth of the error between addressed time steps. It is important to demonstrate for various approaches discussed here, that the control action, and the tracking error between time steps, no longer have these unstable behaviors.

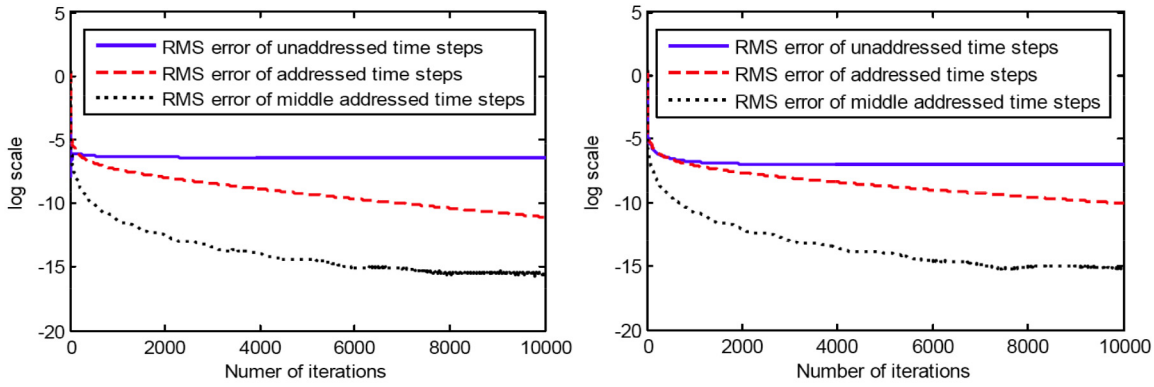

Figure 5-4. RMS error vs. iterations for Approach (c) (left) and Approach (e) (right) for $G_3(s)$ and ILC Law (3).

We demonstrate the behavior for Approaches (c) and (e). We compare approaches using ILC Law (3) and consider $P_i = P_{Wi}$. Gain ϕ is adjusted to match the "DC gain" as above. The desired trajectory is one second long with a smooth start up from zero, $[1-\cos(t)]^2$. Sample rate for the addressed time steps is 100Hz, the initial state is set to zero, the repeating disturbance $\underline{v}$ is zero, and the input for the initial run is set to zero. The left of Fig. 5-4 gives the root mean square (RMS) error for 10,000 iterations of learning. The dashed line is RMS error at the 100 addressed time step, which is still continuing to learn. The dotted line is the RMS of the error at addressed time steps with the first and last 20 time steps removed, and one sees that it reaches a Matlab numerical zero error. The errors at the beginning and the end of the trajectory learn slower than the time steps in the middle. The solid line in the left plot is the error at the single unaddressed time step that has been appended by Approach (c). That curve starts at zero on the log scale, then decreases to almost $10^{-7.65}$, then increases to about $10^{-6.4}$. The solid line in the right plot is the RMS error of the 102 extra sample times introduced by Approach (e). It starts at 2.07 or $10^{-0.32}$ which is similar to the addressed error level, and then decays to about $10^{-7.016}$ and stops improving.

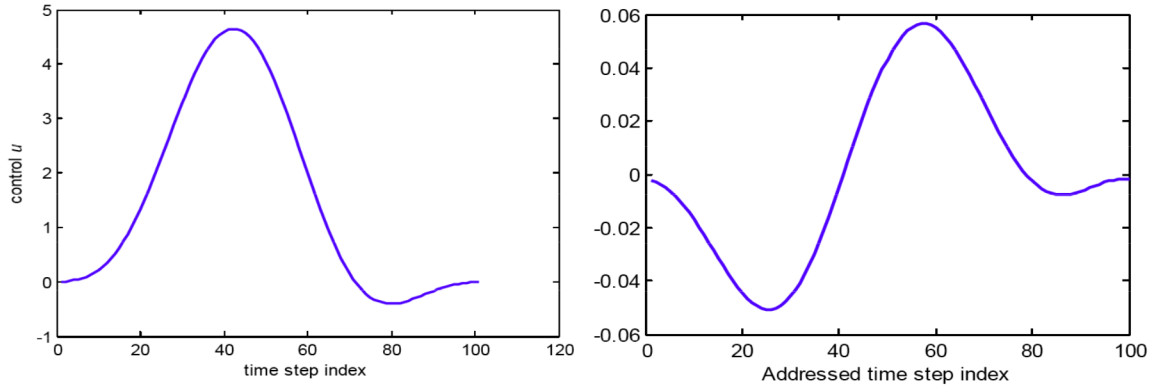

Figure 5-5. Left: the control vs. time step at iteration 10,000 for the left plot in Figure 5-4. Right: the control history for Approach (c) minus the control history at addressed steps for Approach (e) at iteration 10,000.

The left of Figure 5-5 presents the control action for Approach (c) showing that there is no exponential growth of the control magnitude with time step. The corresponding plot for Approach (e) looks very similar, so the right of Figure 5-5 presents the difference between these control action at the addressed steps. There is a difference in the second digit, but because (e) uses twice as many control actions, there is no reason that the curves should be identical. In order to demonstrate that the error between addressed time steps is also well behaved, with no exponential growth, the dashed line on the left of Figure 5-6 presents the errors half way between the addressed time steps based on the derived final error formula for Approach (c). And the solid curve is the corresponding curve for the introduced points between addressed points for Approach (e). Note that Approach (e) has between 2 and 3 orders of magnitude better error level at these intermediate times than Approach (c).

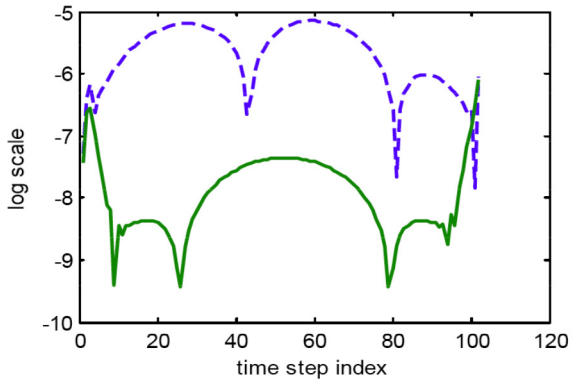

Figure 5-6. Solid curve: Final error at unaddressed time steps using Approach (e). Dashed curve: Final error at the same times, using Approach (c).

Table 5-5. Pseudo-Inverse Robustness

Pseudo-inverse			Max. Eigenvalues			Max. Singular Values		
			>1	>1.001	>1.01	>1	>1.001	>1.01
Pole excess of 3	(A) Approach (b)	1st run	213	1	0	400	212	208
		2nd run	226	1	1	400	204	203
		3rd run	205	0	0	400	216	212
	(B) Append Initial Row (c)	1st run	0	0	0	192	192	192
		2nd run	0	0	0	217	217	216
		3rd run	0	0	0	201	201	200
	(E) Every Other and Append (e)	1st run	0	0	0	192	192	192
		2nd run	0	0	0	217	217	216
		3rd run	0	0	0	201	201	200
Pole excess of 4	(A) Approach (b)	1st run	216	25	25	400	48	48
		2nd run	227	20	19	400	41	39
		3rd run	226	23	23	400	51	50
	(B) Append Initial Row (c)	1st run	24	24	23	55	55	55
		2nd run	26	26	26	46	46	45
		3rd run	16	15	15	42	41	40
	(E) Every Other and Append (e)	1st run	20	20	19	40	39	39
		2nd run	18	17	16	40	40	37
		3rd run	14	14	13	27	27	26
Pole excess of 5	(A) Approach (b)	1st run	337	38	38	400	328	325
		2nd run	354	40	37	400	327	323
		3rd run	340	35	33	400	310	309
	(B) Append Initial Rows (c)	1st run	33	33	32	315	315	314
		2nd run	40	40	39	315	315	314
		3rd run	43	43	43	321	321	319
	(E) Every Other and Append (e)	1st run	47	47	47	321	321	319
		2nd run	39	38	37	308	308	306
		3rd run	45	45	44	308	308	306

5.11 Conclusions

Robustness to model error is particularly important in ILC because it aims to converge to zero error in the world instead of in our model of the world. ILC is an inverse problem, and very often the inverse is unstable asking for exponentially growing control action and error between time steps. These two unwanted properties can easily be eliminated by appending one or more extra

time steps to the start of the desired trajectory, the number equal to the number of zeros outside the unit circle in the *z*-transfer function of the system. Appending more time steps is not usually particularly beneficial. Alternatively, one can introduce extra time steps between the addressed time steps, creating something like a generalized hold. It is shown that it is preferable to combine these two approaches. The result has substantially better error between addressed time steps than simply appending time steps to the beginning of the desired trajectory. Three main ILC control laws are studied, the Euclidean norm contraction mapping law, the partial isometry law, and a quadratic cost ILC law. The quadratic cost law is shown to have the best robustness properties, based on Monte Carlo style evaluation of models based on probability distributions of the coefficients in the model. The quadratic cost law offers the option to use a separate cost function to control intersample error.

5.12 References

[1] Y. Li and R. W. Longman, "Characterizing and Addressing the Instability of the Control Action in Iterative Learning Control," *Advances in the Astronautical Sciences*, Vol. 136, 2010, pp. 1967-1985.

[2] Y. Li and R. W. Longman, "Addressing Problems of Instability in Intersample Error in Iterative Learning Control." *Advances in Astronautical Sciences*. Vol. 129, Part 2, 2008, pp. 1571-1591.

[3] Y. Li and R. W. Longman, "Using Underspecification to Eliminate the Usual Instability of Digital System Inverse Models ." *Advances in the Astronautical Sciences*, Vol. 135, 2010, pp. 127-148.

[4] P. A. Le Voci and R. W. Longman, "Intersample Error in Discrete Time Learning and Repetitive Control," *Proceedings of the 2004 AIAA/AAS Astrodynamics Specialist Conference*, Providence, RI, August 2004.

[5] J. Bao and R. W. Longman, "Unification and Robustification of Iterative Learning Control Laws," *Advances in the Astronautical Sciences,* Vol. 136, 2010, pp. 727-745.

[6] H. S. Jang and R. W. Longman, "A New Learning Control Law with Monotonic Decay of the Tracking Error Norm," *Proceedings of the Thirty-Second Annual Allerton Conference on Communication, Control and Computing,* Monticello, IL, September, 1994, pp. 314-323.

[7] H. S. Jang and R. W. Longman, "Design of Digital Learning Controllers Using a Partial Isometry," *Advances in the Astronautical Sciences*, Vol. 93, 1996, pp. 137-152.

[8] Y. P. Hsin, R. W. Longman, E. J. Solcz, and J. deJong, "Stabilization due to Finite Word Length in Repetitive and Learning Control," *Advances in the Astronautical Sciences*, Vol. 97, 1998, pp. 817-836.

[9] Y. Shi, R. W. Longman, and M. Q. Phan, "An Algorithm for Robustification of Repetitive Control to Parameter Uncertainties," *Advances in the Astronautical Sciences*, Vol. 136, 2010, pp. 1953-1966.

[10] D.H. Owens and N. Amann, "Norm-Optimal Iterative Learning Control," *Internal Report Series of the Centre for Systems and Control Engineering*, Unviersity of Exeter, 1994.

[11] M.Q. Phan and J.A. Frueh, "System Identification and Learning Control," Chapter 15, *Iterative Learning Control: Analysis, Design, Integration, and Applications*, Z. Bien and J. Xu, Editors, Kluwer Academic Publishing, Norwell, MA, 1998, pp. 285-306.

[12] H. Elci, R. W. Longman, M. Phan, J.-N. Juang, and R. Ugoletti, "Discrete Frequency Based Learning Control for Precision Motion Control," *Proceedings of the 1994 IEEE International Conference on Systems, Man, and Cybernetics*, San Antonio, TX, Oct. 1994, pp. 2767-2773.

[13] M. Q. Phan, R. W. Longman, B. Panomruttanarug and S. C. Lee "Robustification of Iterative Learning Control and Repetitive Control by Averaging," *International Journal of Control*, Vol. 86, 2013, No. 5, 855-868.

Chapter 6

Conclusions

Feedback control systems do not do what you ask them to do. The concept of bandwidth is defined in order to describe what frequency components of a command will be executed reasonably well, and what components are not. Iterative Learning Control (ILC) is a method of making control systems that repeat the same trajectory, follow the trajectory with high precision. It has application in high speed high accuracy semiconductors manufacturing. ILC iterates with the real world aiming to converge to that command that produces zero tracking error, rather than iterating on a computer model of the world. As a result much higher accuracy motion can be obtained.

The ILC problem is an inverse problem, given the desired output of a control system, find the command that produces that output. Very often the exact solution of the inverse problem has an unstable control action. If the iterations with the world were to reach the inverse solution, the error at sample times would be zero, but the magnitude of the control grows exponentially with time step, and the error between sample times grows exponentially. Often in practice, the ILC appears to converge to a nonzero level, and perhaps disappointing error level, because too many iterations may be needed to observe the instability.

This internal instability was observed when the digital system has non-minimum phase (NMP) zeros. There are two types of them. The first type is introduced during the discretization if using a zero order hold for any system with pole excess of three or more. The second type is the image of the NMP zeros in the original continuous time system. In this study, we name the first type as sampling zeros and the second type as intrinsic zeros. Sampling zeros usually appear on the

negative real axis, and approach asymptotic locations as the sample interval goes to 0. Intrinsic zeros are connected to the locations of zeros and poles of the original continuous time system. In general, they approach 1 when sampling faster, and move away from the unit circle when sampling slower.

Because of these NMP zeros, when one plugs the desired output into the difference equation to solve for the desired input, the homogeneous solution contains growing terms. If the NMP zero is negative, the control action not only grows exponentially in magnitude, but also alternates its sign every time step.

We introduced two approaches to address this issue. The first approach suggests to append an initial step or steps to the desired trajectory without asking for zero error at this step(s). The number of time steps to append to the start of the desired trajectory is equal to the number of NMP zeros. The second approach can be thought of as using a kind of generalized hold created by having multiple zero order hold inputs between addressed time steps. This gives extra freedom that can avoid instability of the inverse problem. In this case, if desired one can ask to minimize a cost function at the newly introduced intermediate time steps.

ILC uses a matrix model to represent the input-output dynamics of the system. In the original problem asking for zero error at all time steps, the "unstable" control action produced in this finite time inverse problem is the result of ill conditioning of the matrix that must be inverted. Eliminating the requirement for zero tracking error for some initial time steps can eliminate particularly small singular values of the matrix that produce this ill conditioning. These bad singular values have the property that the longer the trajectory, the smaller the singular value, while the original remaining singular values can be related to the frequency response of the system. These bad singular values decay exponentially with length of the trajectory making the matrix

inverse produce control actions that grow exponentially in magnitude. But they alternate sign each time step so that the error can be going through zero at each sample time.

The second approach also eliminates these kinds of bad singular values, but it may introduce some new small singular values, and again they are unrelated to the frequency response of the system. The behavior of these new anomalous singular values and associated singular vectors suggest that they are not related to NMP zeros and will not cause exponentially growing control action as did the original bad singular values. In addition, by appending the right number of initial time steps to the desired trajectory, the new bad singular values can be eliminated as well.

These approaches are shown to be effective for systems with only sampling zeros. However, one may encounter difficulties when applying these rules to address the systems with intrinsic zeros. One must deleting extra rows at the beginning for both approaches. This is not desirable because there is no general rules to follow in terms of how many time steps one needs to skip at the beginning. And the number is affected not only by locations of the NMP zeros but also the poles and zeros inside the unit circle.

We developed a filter that tries to cancel poles and zeros inside the unit circle, then the remaining system only has NMP zeros. It is shown that all the rules developed for sampling zeros work again for intrinsic zeros. This extends the use of the two approaches to solve the internal instability problem for more general NMP system.

In addition, the robustness test with respect to model error suggests that both approaches are robust to model parameter error. The multiple zero order hold approach has slightly better final error level at unaddressed time steps.

There are still more things to explore in the future. We intend to examine the robustness of such strategy with respect to model parameter errors, and then extend the robustness test to systems with parasitic poles. As discussed above, the sampling zeros are introduced during the discretization, which results in instability when inverting digital systems. However, the original continuous-time system has no unstable zeros, so its inverse is stable. The most straight forward way is to obtain the inverse solution in continuous time and then convert this signal to digital. If one feeds this control action to a digital system through a zero order hold, the final error level is somewhat disappointing, indicating that the approach cannot by pass the internal instability of the digital system. Another aspect of our future work is to investigate the methods of obtaining discrete input signals from the continuous-time inverse solution, and also try to create equivalent digital learning law based on continuous-time learning laws.

Asking for zero tracking error of a feedback control system following a desired trajectory requires inverting the system. But some systems are non-minimum phase in continuous time which appears as a more fundamental issue for instability of the inverse of the system. Our work is an important contribution to solving this major problem that has plagued the field of control of non-minimum phase systems.