Development of Robust Learning Control and Application to Motion Control

Meng-Shiun Tsai, Chung-Liang Yen and Hong-Tzong Yau
Department of Mechnaical Engineering and Advanced, Institute of Manufacturing with High-tech Innovations, National Chung-Cheng Unviersity, 168, University Rd, MinHsiung, Chiayi, Taiwan

Keywords: Iterative Learning Control, Motion Control, Nurbs Curves, Robust H_∞ Control.

Abstract: In this paper, the error dynamic equation of the ILC algorithm is derived with consideration of parameter uncertainties and noise. The H_∞ frame work is utilized using the derived error dynamics to design the robust learning controller. The proper learning gain is designed based on an optimization process to ensure that both tracking performance and convergence condition can be achieved. Simulation and experiments are conducted to validate the robust learning algorithm and the system is stable ever under high payload uncertainty.

1 INTRODUCTION

Iterative learning control (ILC) is a technique to control the system when it operate same tasks repetitively. The ILC can be applied to robot manipulators (Tayabi and Islam, 2006), chemical batch process (Lee and Lee, 2007), and so on. Many schemes of the ILC including the 2D theory method (Geng et al., 1990), stochastic method (Wang and Afshar, 2009), inverse system (Ye and Wang, 2005), and feedback learning operators (Goldsmith, 2002; Chin et al., 2004) have been proposed. Technical review on the methodologies and applications of the ILC is referred to (Ahn et al., 2007).

System robustness is generally a major concern in the implementation of ILC to either linear or nonlinear systems. The adaptive iterative learning control was proposed (French and Rogers, 2000). The Lyapunov method was adopted to prove the convergence of the algorithm. Other adaptive ILC algorithms were proposed to handle system with time-varying parameters using a positive-definite Lyapunov-like sequence (Kuc et al., 1991). Another approach to ensure system robustness is to utilize the H_∞ theory to formulate the general design framework for the ILC algorithm (Padieu and Su, 1990). In these papers, only the performance and robustness analysis of ILC schemes are considered.

In this paper, two steps design process is proposed. The first step is to design the H∞ controller without consideration of the system uncertainty. But the noise effect is included in the design process. The second step is to iterate the learning gain such that the convergence condition is satisfied even under large system uncertainty. The learning gain served as the performance weighting which is the loop optimization variable to further minimize system performance. Simulations and experiments are conducted to demonstrate the design philosophy.

2 MODELLING OF SERVO CONTROL SYSTEM

In this paper, the command-based ILC is applied to a CNC milling machine tool (Tsai et al., 2006). The general servo control system as shown in Fig. 1 which includes the linear dynamic model of the servo system, the velocity and position loops with a velocity feedforward controller F(s). The function $F(s)$ is designed as $K_f s$ where K_f is the feedforward gain. The J_s, B_s, K_t and h_p are the moment of inertia, viscosity, torque constant and pitch of lead screw. The parameters K_{vp} and K_{vi} in the velocity loop can be designed by specifying the damping ratio and bandwidth of the closed-loop transfer function of the velocity loop. The position gain K_{pp} in the position loop is determined by the designed bandwidth of the position loop. The

Tsai M., Yen C. and Yau H..
Development of Robust Learning Control and Application to Motion Control.
DOI: 10.5220/0004008601480152
In *Proceedings of the 9th International Conference on Informatics in Control, Automation and Robotics* (ICINCO-2012), pages 148-152
ISBN: 978-989-8565-21-1

feedforward gain K_f is designed by using an optimization approach, which minimizes the sum of the magnitudes of the error transfer function at various frequencies. According to Fig. 1, the transfer functions between the output $y_k(s)$ and the reference command $r_k(s)$, and the sensor noise $n_k(s)$ are givens as:

$$y_k(s) = G_r(s)r_k(s) + G_n(s)n_k(s) \tag{1}$$

Because the learning process is implemented in discrete-time domain, Eq. (1) is converted to the discrete-time model using the zero-order hold technique and is given as:

$$y_k(z) = G_r(z)r_k(z) + G_n(z)n_k(z) \tag{2}$$

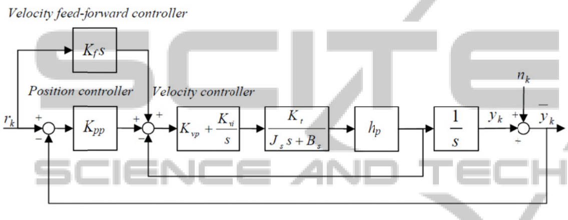

Figure 1: Architecture of servo control system with measurement noise.

3 OPTIMIZATION PROCESS

In the previous section, the general model of the servo control system is developed. In this section, the error dynamics equation is derived and design methodology is proposed.

3.1 Error Dynamics Equation

Figure 2 illustrates the architecture of command-based ILC where r_k and r_{k+1} are the input commands at the k^{th} and the $(k+1)^{th}$ iteration, respectively. y_d is the desired trajectory, $n_k(z)$, $\bar{y}_k(z)$ and $\bar{e}_k(z)$ denote the measurement noise, the output signal and the tracking error with measurement noise. The tracking error $\bar{e}_k$ processed by a learning control $L(z)$ and learning gain Φ is added to the reference command r_k to obtain the new updated command r_{k+1}. The complete learning process shown in Fig. 2 can be represented by the following equations.

$$r_{k+1}(z) = r_k(z) + \Phi L(z)\bar{e}_k(z) \tag{3}$$

$$e_k(z) = y_d(z) - y_k(z) \tag{4}$$

After derivation, the error dynamic equation can be simplified as:

$$e_{k+1} = (I - \Phi LG_r)e_k + \Phi LG_r n_k \tag{5}$$

Equation (3) indicates that the tracking performance is strongly influenced by the transfer function ΦLG_r. Using the multiplicative uncertainty representation, the G_r can be further represented by the nominal plant $G_{r0}(z)$ and the weighting function $W(z)$ and given as:

$$G_r(z) = G_{r0}(z)(1 + W(z)\triangle(z)) \tag{6}$$

Here $\triangle(z)$ is the small perturbation normalized to be $\|\Delta\|_\infty < 1$. The errors dynamics can be further expressed as:

$$e_{k+1} = [I - \Phi LG_{r0}(I + W\Delta)]e_k + \Phi LG_{r0}[I + W\Delta]n_k \tag{7}$$

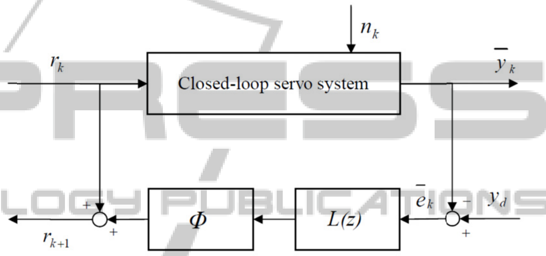

Figure 2: Architecture of command-based ILC with measurement noise.

3.2 Robust ILC using H∞ Control

After developing the error dynamics, the H_∞ control can be designed by first transforming the error dynamics into a linear fraction transformation (LFT) form (Zhou et al., 1996). Figure 3 shows the block diagram corresponding to the error dynamic equation (12). The block diagram can be represented by the following LFT form given as:

$$\begin{bmatrix} y_J \\ e_{k+1} \\ y_L \end{bmatrix} = \begin{bmatrix} 0 & -\Phi G_{r0} & \Phi G_{r0} & 0 \\ 0 & 0 & I & -I \\ W & -\Phi G_{r0} & \Phi G_{r0} & 0 \end{bmatrix} \begin{bmatrix} u_J \\ n_k \\ e_k \\ u_L \end{bmatrix} = P_A \begin{bmatrix} u_J \\ n_k \\ e_k \\ u_L \end{bmatrix} \tag{8}$$

where P_A is the augmented plant, Φ is the learning gain which is the designed variable for the optimization process, y_J, y_L, u_J and u_L are the output of the augmented plant, input to the learning function $L(z)$, input to the augmented plant and output of the learning function $L(z)$, respectively. Equation (6) can be further represented by the following form:

$$\begin{bmatrix} \tilde{Z} \\ y_L \end{bmatrix} = \begin{bmatrix} P_{11} & P_{12} \\ P_{21} & P_{22} \end{bmatrix} \begin{bmatrix} \tilde{W} \\ u_L \end{bmatrix} \tag{9}$$

The relation between the y_L and u_L is given as $u_L = L(z)y_L$. The augmented system can be expressed as a standard form for designing the H_∞ controllers shown below:

$$\tilde{z} = \begin{bmatrix} y_J \\ e_{k+1} \end{bmatrix} = P_{11}\tilde{W} + P_{12}u_L \tag{10}$$

$$y_L = P_{21}\tilde{W} + P_{22}u_L \tag{11}$$

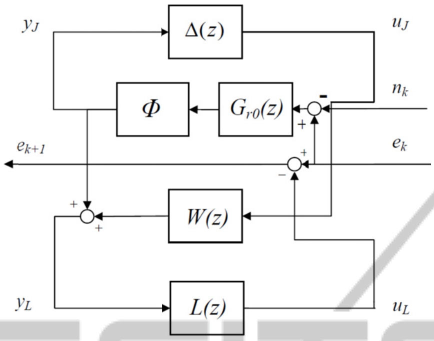

Figure 3: Architecture of ILC synthesis problem with considering measurement noise

With the developed models by Eqs. (8) and (9), the perturbed system model can be used for designing the robust H_∞ controller can be designed by solving the two Riccati equations. After designing the learning function L, the closed loop between the $\tilde{Z}$ and $\tilde{W}$ can be represented as the following:

$$T_{\tilde{Z}\tilde{W}} = \begin{bmatrix} 0 & -\Phi G_{r0} & \Phi G_{r0} \\ -LW & \Phi LG_{r0} & 1-\Phi LG_{r0} \end{bmatrix} \tag{12}$$

It is noted that the following equation provides a design criteria in selecting the learning gain Φ .

$$\|T_{\tilde{Z}\tilde{W}}\|_\infty \geq \|\Phi G_{r0}\|_\infty \tag{13}$$

And the sufficient convergence condition of the ILC algorithm (Roover, and Bosgra, 2000) is to ensure that

$$\|T_{\tilde{Z}\tilde{W}}\|_\infty \leq 1 \tag{14}$$

Equation (12) implies that the condition of $\Phi < 1$ should be set in order to satisfy the convergence condition.

3.3 Optimization Process

By observing the closed loop transfer function, it is found that the stability and performances of the ILC algorithm are strongly influenced by the uncertainties. Before introducing the optimization process, the H_∞ controller designed process without considering the plant uncertainties is given as follows.

$$\begin{bmatrix} \tilde{Z}_1 \\ y_L \end{bmatrix} = \begin{bmatrix} P_{11} & P_{12} \\ P_{21} & P_{22} \end{bmatrix} \begin{bmatrix} \tilde{W}_1 \\ u_L \end{bmatrix} \tag{15}$$

where $P_{A2} = \begin{bmatrix} P_{11} & P_{12} \\ P_{21} & P_{22} \end{bmatrix}$ is the augmented plant. The augmented system can be expressed as a standard form for H_2/H_∞ controller design as shown below:

$$\tilde{z}_L = \begin{bmatrix} y_J \\ e_{k+1} \end{bmatrix} = P_{11}\tilde{W}_1 + P_{12}u_L \tag{16}$$

With the designed controller, the closed loop transfer function can be represented as the following:

$$\begin{aligned} T_{\tilde{Z}_1\tilde{W}_1} &= P_{11} + P_{12}(I - LP_{22})^{-1}LP_{21} \\ &= [\Phi LG_{r0} \quad 1-\Phi LG_{r0}] \end{aligned} \tag{17}$$

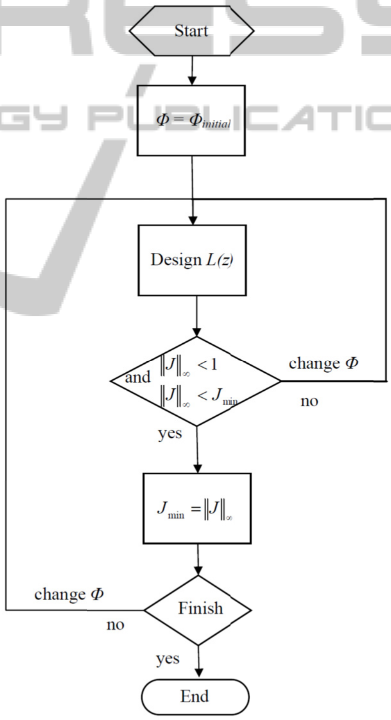

Figure 4: The flow chart of designing learning gain and learning controller.

The design process is to start by selecting the learning gain Φ and ensure the stability condition is satisfied. The procedure starts by first choosing an initial value of Φ and then apply the H_∞ theory to design the controller $L(z)$ based on the augmented plant. The second step is to ensure that the designed controller $L(z)$ should satisfy the sufficient condition of the convergence where the plant uncertainty is included.

4 EXPERIMENTAL VALIDATION

To validate the design process, simulation analysis and experiments are performed using a three-axis machine tool system. The x, y, and z axes of the machine tool are driven by Panasonic MHMD042S1S servo motors and MHMD042S1S servo drives. The resolution of the linear scale for each axis is equal to 1μm. The real-time extension (RTX) (VenturCom Inc., 2006) is used to ensure the control system with real-time performance. The output data from the linear scale are processed by the Advantech 1784 encoder card. The plant parameters are identified using a frequency domain approach. The nominal plant $G_{r0}(s)$ is measured with a linear swept-frequency signal at the magnitude of -1 Volt to 1 Volt and the frequencies from 0.1 Hz to 1000 Hz, and identified by using the Empirical Transfer Function Estimate (ETFE) method (Ljung, 1999). The damping ratio and the bandwidth of the velocity closed-loop transfer function are chosen as 1.0 and 502.65 rad/sec, respectively. The bandwidth of position closed-loop transfer function is chosen as 251.33 rad/sec and the velocity feedforward gain K_f is selected to 0.95.

To ensure that the proposed algorithm can be applied to a complex trajectory, a butterfly NURBS curve is used as a working example as shown in Fig.5. Three payloads given as 0, 10, and 20 Kg are tested in the moving platform. The 10 and 20 Kg payload changes can cause the moment of inertial J_s to deviate about 8% and 16%, respectively. The corresponding weighting function $W(s)$ to cover the uncertainty bounds is given as:

$$W(s) = \frac{0.5s}{s+200} \tag{18}$$

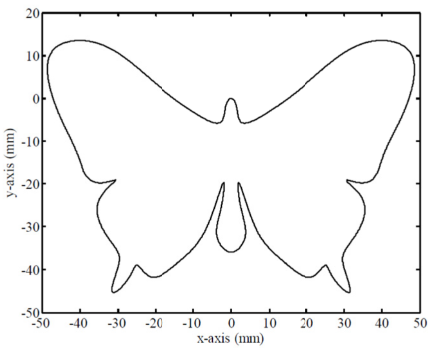

Figure 5: The butterfly curve.

After applying the design process shown in Fig. 4, it is found that the learning gain Φ should be in the range of 0.3 and 0.53 to satisfy the convergence condition. The optimal learning gain is determined to be 0.53.

Experiments are conducted on the machining tool with the payload equal to 0, 10 kg and 20 kg, respectively. The comparisons of tracking error between the cases of without learning and the learning after the 10th iteration are shown in Figs. 6 and 7. From the results, the tracking performances of the x-axis and the y-axis are much improved under the conditions of different loadings. The RMS values of x-axis output tracking errors under the conditions of that without loading, loading 10kg weight and loading 20 kg weight reduce 99.40%, 99.40% and 99.41% after 10 iterations. The RMS values of y-axis output tracking errors under the conditions of that without loading, loading 10kg weight and loading 20 kg weight reduce 99.33%, 99.30% and 99.12% after 10th iteration. The maximum values of the x-axis output tracking errors under the conditions of that without loading, loading 10kg weight and loading 20 kg weight reduce 98.16%, 98.71% and 98.77% after 10th iteration. The results illustrate that not only the controller satisfies the system robustness but also the trends of output tracking errors are similar under different loadings.

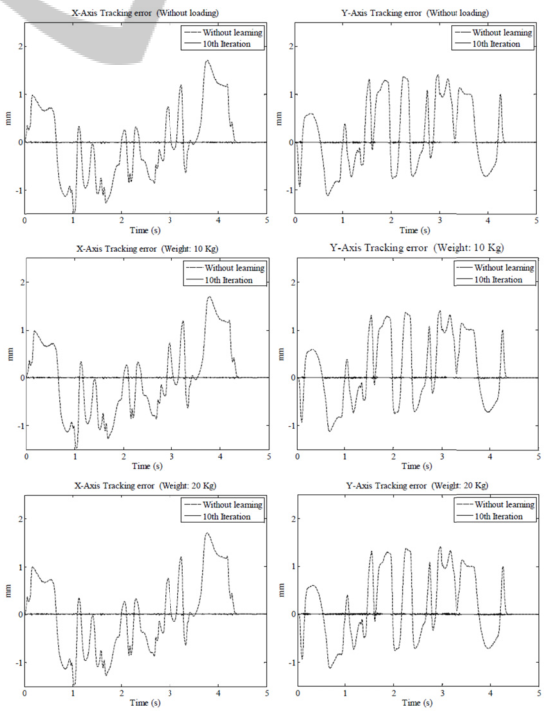

Figure 6: The comparisons of tracking error between the cases of without learning and the learning.

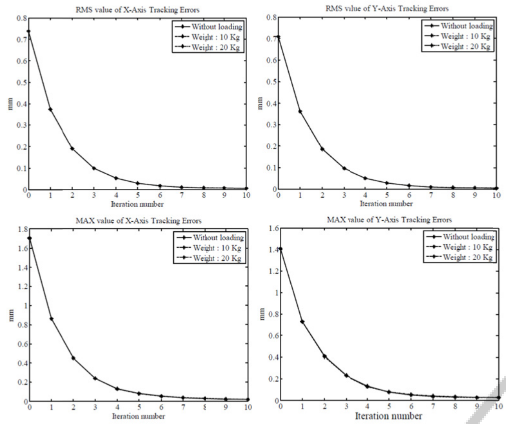

Figure 7: The trends for the RMS and maximum values of the tracking errors.

5 CONCLUSIONS

A modified H_∞ optimization process is proposed in this paper to provide a systematic methodology in choosing the learning gain and designing the learning function. According the proposed methodology, the appropriate learning gain and learning function can be designed simultaneously. Experiments are conducted on the machining tools with different loading conditions. It is shown that the reduction of tracking error is over 97% after 10 iterations for the nominal plant. Furthermore, the comparisons between the plants with different payloads demonstrate that system robustness can be achieved.

ACKNOWLEDGEMENTS

The authors are grateful to the Ministry of Economic Affairs, R. O. C. for the financial supports provided to this study under contract 98 – EC – 17 – A – 05 – SE - 101, and 100-EC-17-A-05-S1-189, respectively.

REFERENCES

Ahn, H. S., Chen, Y. Q., and Moore, K. L., 2007, Iterative learning control: brief survey and categorization, *IEEE Trans. on Systems, Man, and Cybernetics*, 37, 1099-1121.

French, M., and Rogers, E., 2000, Nonlinear iterative learning by an adaptive Lyapunov technique. *International Journal of Control*, 73, 840-850.

Geng, Z., Carroll, R., and Xie, J., 1990, Two-dimensional model and algorithm analysis for a class of iterative learning control systems. *Int. J. Control*, 52, 833-862.

Kuc, T. Y., Nam, K., and Lee, J. S., 1991, An iterative learning control of robot manipulators. *IEEE Trans. Robot. Autom.*, 7, 835–841.

Lee, J. H., and Lee, K. S., 2007, Iterative learning control applied to batch processes: An overview. *Control Engineering Practice*, 15, 1306-1318.

Ljung, L., 1999, System Identification: Theory for the User. *Prentice Hall*.

Padieu, F., and Su, R., 1990, H_∞ approach to learning control systems, *Int. J. Adaptive Contr. Signal Processing*, 4, 465–474.

Tayabi, A, and Islam, S., 2006, Adaptive iterative learning control for robot manipulators: Experimental results. *Control Engineering Practice*, 14, 843-851.

Tsai, M. S., Lin, M. T., and Yau, H. T., 2006, Development of command-based iterative learning control algorithm with consideration of friction, disturbance, and noise effects. *IEEE Trans. Control Syst. Technol.*, 14, 511-518.

VenturCom Inc., 2006, VenturCom RTX 6.5 SDK Documentation. *Cambridge*.

Wang, H., and Afshar, P., 2009, ILC-Based Fixed-Structure Controller Design for Output PDF Shaping in Stochastic Systems Using LMI Techniques. *IEEE Trans. on Automatic control*, 54, 760-773.

Ye, Y. Q., and Wang, D. W., 2005, Clean system inversion learning control law. *Automatica*, 41, 1549–1556.

Zhou, K., Doyle, J. C., and Glover, K., 1996, Robust and Optimal Control. New Jersey, *Prentice Hall*.