Copyright © 2002 IFAC
15th Triennial World Congress, Barcelona, Spain

ON THE ILC DESIGN AND ANALYSIS FOR A HDD SERVO SYSTEM [1]

Jian-Xin Xu, Tong Heng Lee, Heng-Wei Zhang

Department of Electrical and Computer Engineering
National University of Singapore
elexujx@nus.edu.sg

Abstract: In this paper, several types of iterative learning control (ILC) schemes for hard disk drive (HDD) control, including previous cycle learning (PCL), current cycle learning (CCL) and their combination, are designed and analyzed under the sampled-data environment. The feedback loop of the CCL stabilizes the system in time domain while the PCL guarantees the convergence along the iteration axis. Therefore, the combination of these two schemes is applied and the effectiveness of the scheme is illustrated through simulations and experiments.

Keywords: Iterative Learning Control, Sampled-data, Hard Disk Drives, Disturbance rejection

1. INTRODUCTION

Iterative Learning Control (ILC) was first introduced and applied by Arimoto and his coworkers in 1984 (Arimoto and Miyazaki, 1984). More and more intention has been given to this new method and many ILC algorithms have been proposed hitherto. In general, ILC is a technique for improving the performance of systems or processes that operate repetitively over a fixed time interval. In practice, this defines a broad class of systems to which the technique can be applied (Moore, 1993).

Hard disk drives (HDDs), serve as an important data storage medium for data processing systems, require high precision control and complete rejection of disturbances. Among plenty of research work in this area, repeatable run-out (RRO) compensation attracts much attention. Since RRO is a repeatable signal with finite interval and it has consistent initial condition, RRO compensation is actually under a repetitive control environment. Therefore, ILC schemes are the best solution for such systems in practical because they could achieve perfect tracking performance in the finite time interval (Tae-Jeong Jang and Ahn, 1995), (S.M. Zhu and Low, n.d.), (Jung-Ho Moon and Chung, 1996). Due to the more and more common applications of computer control, the ILC has been investigated in discrete time for a long time (Kurek and Zaremba, 1993), (Saab, 1995b), (Bien and Xu, 1998). Most ILC schemes can be classified into previous cycle learning (PCL), current cycle learning (CCL) or their combinations. The essential difference between PCL and CCL is the introduction of feedback loop in CCL. The latter becomes robust to the tracking error. However, CCL is not suitable for learning with sampled delay and does not guarantee the learning convergence. On the other hand, PCL is a noncausal system. It can compensate the sampled delay easily and therefore guarantee the learning convergence. The PCL is open loop system with no robustness. A combination scheme, namely previous and current cycle learning (PCCL) scheme, is therefore introduced. The PCCL guarantees the learning convergence while keep the system robust. In this work, design and analysis of CCL, PCL and PCCL will be conducted for HDD RRO compensation.

[1] This work was supported in part by National University of Singapore under research project 'Development of Advanced Run-out Compensation Techniques for High precision HDD Control' (RP3972694)

The paper is organized as follows. In Section 2, for a given HDD model, CCL and PCL schemes are discussed in sampled data systems. In Section 3, the PCCL scheme is designed and analyzed based on the system stability and learning convergence. To illustrate the effectiveness of the schemes, simulations and experiments are conducted. Finally, the conclusion is drawn in Section 4.

2. SAMPLED DATA ILC

2.1 *Sampled HDD*

A simplified HDD model is essentially a double integrator. Its open loop discrete time transfer function is

$$G_o = \gamma \frac{T_s^2}{2} \frac{z+1}{(z-1)^2} \quad (1)$$

where, γ is the system gain of the HDD, T_s is the sampling period. The z-domain transfer function is obtained by zero order hold (ZOH) method.

2.2 *Sampled data CCL*

It is known that the CCL scheme best fits the RRO compensation problem for HDD in the continuous time (Jian-Xin Xu and Zhang, 2001) for it can eliminate the RRO disturbance effectively with relative degree 1. In sampled data systems, the convergence condition of CCL is however different . Suppose that the discrete updating law of CCL scheme at $i+1^{th}$ iteration is

$$u_{i+1}(k) = u_i(k) + K_p e_{i+1}(k) + T_d \eta_{i+1}(k), \quad (2)$$

where, in z-domain,

$$\eta_{i+1} = \frac{N(z-1)}{z - e^{-NT_s}} e_{i+1}$$

When the sampling rate is high (say, $T_s = 1/6000(sec)$), the system is approximately the same as in continuous time and the CCL scheme still works properly at the first several iterations, as shown in Fig. 1.

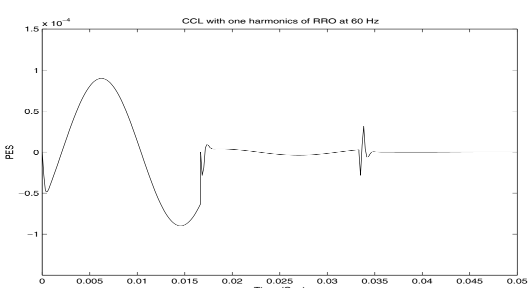

Fig. 1. The first 3 iterations under CCL scheme

However, when learning carries on, the system sampling delay will gradually degrade the learning performance. As shown in Fig. 2, the controller signal

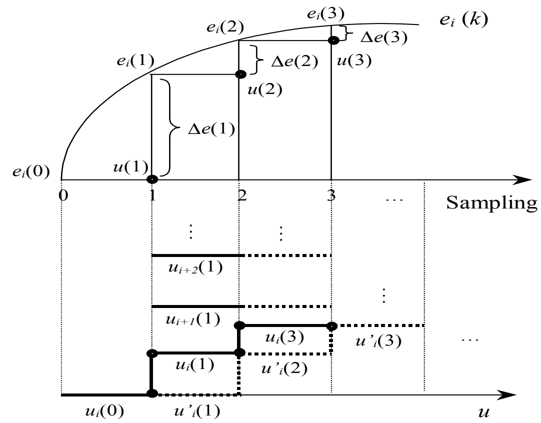

Fig. 2. Initial errors in one iteration

obtain at k^{th} sample can only function at the period between the k^{th} and $k+1^{th}$ sample. Applying resetting condition, $e(0) = 0$, $u(0) = 0$, the controller signal between the 0^{th} and 1^{st} sample is zero. At 1^{st} sampled instant, the error is no longer zero (which is $\Delta e(1)$ shown in Fig. 2, denoted as initial error). We should note that this initial error $\Delta e(1)$ is constant since we assume the tracking error is periodic signal and the learning is an iterative procedure. Therefore, the updated controller signal will be summed up at the 1^{st} sampled instant and diverging to infinity along the iteration axis. Since the CCL scheme is a causal system, it could never make up this initial error caused by sampling delay.

Suppose at i^{th} iteration, the initial error is $\Delta e_i(1)$. According to the updating law of CCL in (2),

$$\begin{aligned} u_i(1) &= u_{i-1}(1) + K_p e_i(1) + T_d \eta_i(1) \\ &= u_0(1) + \sum_{j=1}^{i} (K_p e_j(1) + T_d \eta_j(1)). \end{aligned} \quad (3)$$

At the first iteration, there is nothing in the memory, $u_0(1) = 0$. Hence, the controller signal at the 1^{st} sampled instant will go divergence because of the integral action of the ILC law (3) along the iteration axis whenever $K_p e_j(1) + T_d \eta_j(1) =$ constant $\neq 0$. Fig. 3 shows the divergence of the CCL in sampling time domain and Fig. 4 shows the controller signals up to first 5 samples and first four iterations (which are u_1, u_2, u_3 and u_4 in the figure). Note that the divergence of controller signal at the 1^{st} sampled instant will influence the following several sampled instants.

Considering the system control delay in a practice system whereby in computing $u_i(k)$ only $e_i(k-1)$ is available. The controller signal will go divergence due to the same reason at the 2^{nd} sampling instant $k = 2$ (shown as the dotted curve of $u'_i(k)$ in Fig. 2).

2.3 *Sampled data PCL*

To solve the sampled data delay problem, it needs a non-causal compensation scheme. The PCL scheme is

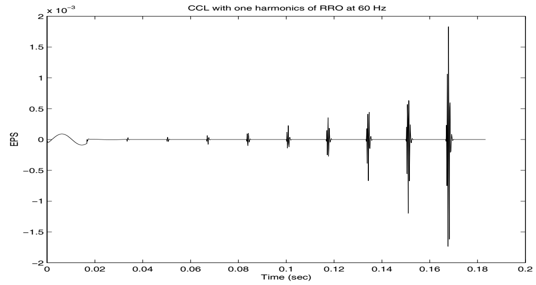

Fig. 3. CCL scheme in sampled data system

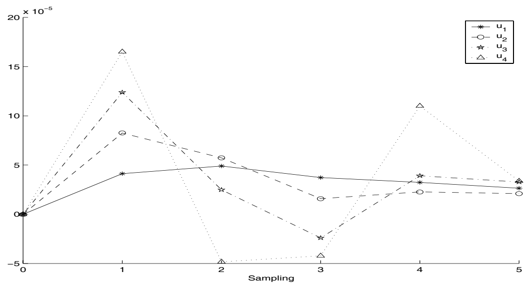

Fig. 4. The controller signal at 1^{st} 5 iterations

hence employed. It is well known that the discrete-time PCL scheme can compensate for any relative degrees (Xu, 1997), (Hwang and Oh, 1991). Since the discrete-time HDD has relative degree one (from (1)), the updating law of PCL in discrete system is

$$u_{i+1}(k) = u_i(k) + K_p e_i(k+1) + \eta_i(k+1) \quad (4)$$

Note that the error information of one sample ahead from previous iteration is used to update the controller input signal. A drawback of PCL is its open loop nature. When it is applied to the HDD system, which is essentially a double integrator, the PCL could be unstable within each iteration, i.e., diverges, in the time domain. The simulation result of PCL is shown in Fig. 5

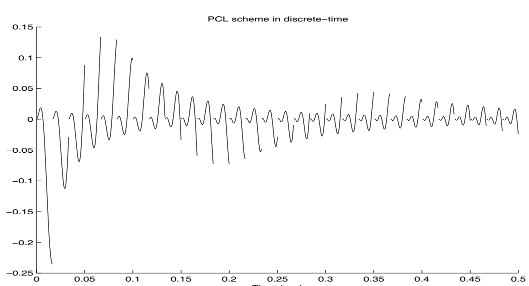

Fig. 5. PCL scheme in sampled data system

As the result shown, the PCL tends to diverge in each iteration. This slows down the learning convergence speed along the iteration axis.

3. SAMPLED DATA PCCL SCHEME

Since the CCL scheme constructs a causal system, it could not eliminate the initial error caused by sampled data delay in a discrete-time system. However, the feedback loop in this scheme can stabilize the HDD system in time domain. On the other hand, the PCL scheme is a non-causal system, which means the data at any sampling point from the previous iterations are available. In this way, the initial error can be eliminated. From the properties of these two schemes, a combination scheme, namely the previous and current cycle learning (PCCL) scheme, is therefore introduced in discrete time.

For a HDD system shown in (1), the design is conducted in two steps. First, a proper PD controller is chosen to stabilize the system. Then, the previous information is used to improve the learning performance. The formulation is shown as

$$u_{i+1}(k) = u_{i+1,\ PCL}(k) + u_{i+1,\ PD}(k)$$
$$u_{i+1,\ PCL}(k) = u_{i,\ PCL}(k) + u_{i,\ PD}(k),$$

where, $C_{PCL}(k)$ indicates the PCL scheme portion and $C_{CCL}(k)$ indicates the CCL scheme portion. The block diagram of PCCL scheme is illustrated in Fig. 6. Note that for the computer control system to an analogue HDD plant, zero-order-hold (ZOH) modules are used as AD/DA converters. In the diagram, T_s is the sampling period and T_f is the learning iteration period.

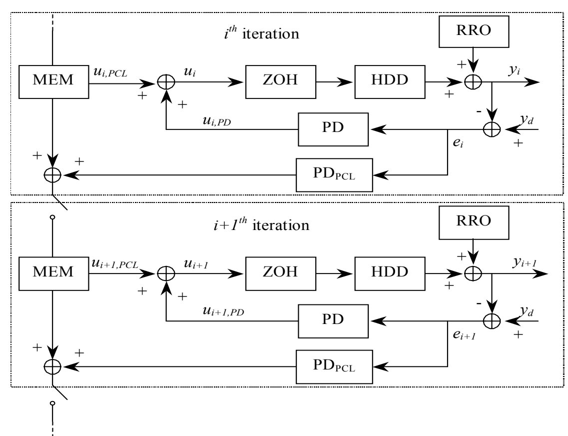

Fig. 6. Block diagram illustration of PCCL scheme

3.1 *Closed-loop design in time domain*

First, we need to prove this new scheme is stable with the additional PD control feedback loop. For the convenience in the analysis, we transfer the approximated differential expression in s-domain, $\frac{N\ s}{N+s}$, to z-domain, $\frac{N(z-1)}{z-e^{-NT_s}}$, by ZOH method, where, N is a constant factor chosen as 100 and T_s is the sampling period. Therefore, we have the close-loop transfer function of the HDD plant with a PD controller feedback, in z-domain, which is

$$
\begin{aligned}
G_c(z) &= \frac{Y(z)}{U(z)} \\
&= \frac{\gamma \frac{T_s^2}{2} \frac{z+1}{(z-1)^2}}{1 + \gamma \frac{T_s^2}{2} \frac{z+1}{(z-1)^2} [K_{ccl} + T_{ccl} \frac{N(z-1)}{z-e^{-NT_s}}]},
\end{aligned} \tag{5}
$$

where, $Y(z)$ and $U(z)$ are the output and input of the HDD plant with the PD feedback, respectively. γ is the gain of HDD, K_{ccl} and T_{ccl} are the proportional and derivative gain of PD controller, respectively. The characteristic function of (5) is

$$
1 + \gamma \frac{T_s^2}{2} \frac{z+1}{(z-1)^2} [K_{ccl} + T_{ccl} \frac{N(z-1)}{z - e^{-NT_s}}] = 0. \tag{6}
$$

Substituting the actual parameters $\gamma = 1.1467 \times 10^8$, $N = 100$ and $T_s = 1/6000$ into the characteristic function (6) yields

$$
\begin{aligned}
z^3 + (1.5926K_{ccl} + 159.2639T_{ccl} - 2.9835)z^2 \\
+(0.02632K_{ccl} + 2.9669)z \\
-(1.5663K_{ccl} + 159.2639T_{ccl} + 0.9835) = 0
\end{aligned}
$$

The values of K_{ccl} and T_{ccl} can be adjusted to guarantee the stability of the system, i.e., all the roots of characteristic function (6) are in the unit circle. The following figures (Fig. 7 - 9) show the absolute values of three roots $|z_1|$, $|z_2|$ and $|z_3|$ w.r.t. K_{ccl} and T_{ccl}.

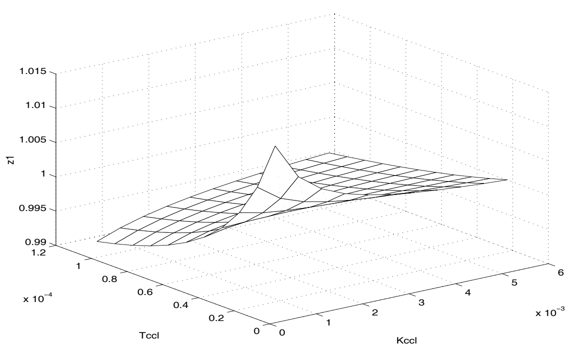

Fig. 7. $|z_1|$ of the characteristic function

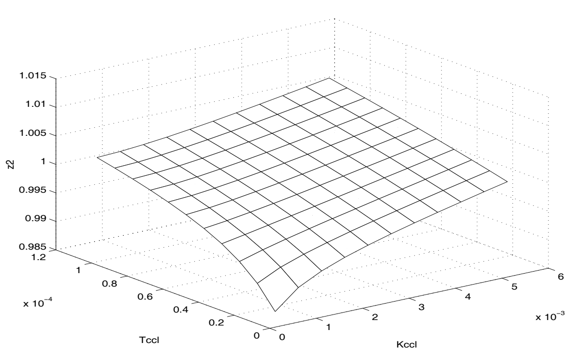

Fig. 8. $|z_2|$ of the characteristic function

The figures show that there is a feasible region for K_{ccl} and T_{ccl} to ensure the absolute values of all three roots of z strictly less than 1 simultaneously (i.e., all the poles of (6) are within unit circle).

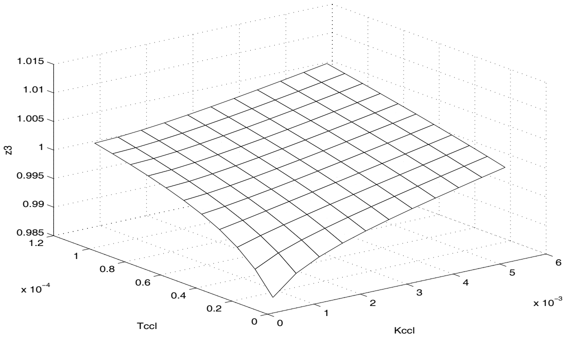

Fig. 9. $|z_3|$ of the characteristic function

3.2 *Learning design in iteration domain*

The next step in PCCL design is to show the PCL part is convergent along the iteration axis.

The transfer function of the HDD plant shown in (1) has a relative degree of 1. Considering there is one system control delay in the system, the updating law of the PCL should be

$$
u_{i+1}(k) = u_i(k) + P_l e_i(k+2), \tag{7}
$$

where,

$$
P_l = K_{pcl} + T_{pcl} \frac{N(z-1)}{z - e^{-NT_s}}
$$

is the z-domain PD-controller transfer function of PCL. Referring to Fig. 2, at $k = 0$ and $i+1^{th}$ iteration, $e_{i+1}(0) = \dot{e}_{i+1}(0) = 0$ by resetting condition. The controller signal is

$$
u_{i+1}(0) = u_i(0) + P_l e_i(1).
$$

Note that now in computing control signal at $k = 0$, we use the error signal at $k = 1$. This plays a role as one step prediction advanced in iteration domain such that the one step system sampling delay can be compensated.

Let us now verify the convergence condition of the PCL scheme. Define the CCL close-loop transfer function as G_c and the PCL scheme transfer function as P_l. Thus, in sampled data system,

$$
\begin{aligned}
y_d &= G_c u_d \\
y_{i+1}(k+1) &= G_c u_{i+1}(k) \\
e_{i+1}(k+1) &= y_d - y_{i+1}(k+1) \\
\delta u_{i+1}(k) &= u_d - u_{i+1}(k) \\
\Rightarrow \quad e_{i+1}(k+1) &= G_c \delta u_{i+1}(k)
\end{aligned} \tag{8}
$$

According to the updating law of PCL,

$$
\begin{aligned}
u_{i+1}(k) &= u_i(k) + P_l e_i(k+2) \\
\Rightarrow \quad \delta u_{i+1}(k) &= \delta u_i(k) - P_l e_i(k+2)
\end{aligned} \tag{9}
$$

Applying z-transform to (8) and (9) and assuming the zero initial conditions, we have the following equations in frequency domain

$$
\begin{aligned}
\delta u_{i+1}(z) &= \delta u_i(z) - P_l z \, e_i(z) \\
&= \delta u_i(z) - P_l G_c z \delta u_i(z) \\
&= (\mathbf{I} - P_l G_c z) \delta u_i(z)
\end{aligned} \tag{10}
$$

The PCCL scheme converges if all the magnitude of eigenvalues of $\mathbf{I} - P_l(z)G_c(z)z$ are less than 1. In this case, $|1 - P_l(z)G_c(z)z| \leq \rho < 1$ can ensure the convergency of the scheme, where, ρ is a positive real number strictly less than 1. Once K_{ccl} and T_{ccl} are chosen to make the PCCL stable in time domain, K_{pcl} and T_{pcl} can be tuned accordingly so that the learning control system converges (Refer to the section Appendix for the detail calculations of choosing K_{pcl} and T_{pcl}).

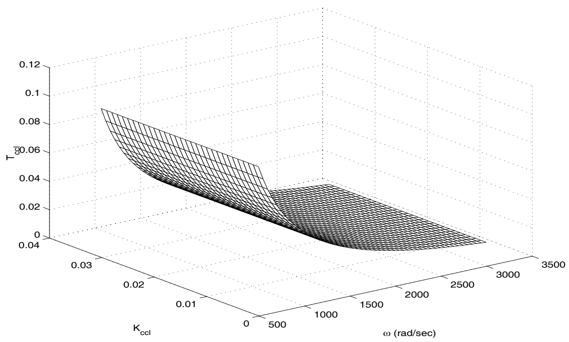

Fig. 10. Convergence region of PCCL

Fig. 10 shows the convergence region when choosing $K_{pcl} = K_{ccl}$ and $T_{pcl} = T_{ccl}$ and $\rho = 0.9$. Note that the system will converge when K_{pcl} and T_{pcl} are chosen below this region.

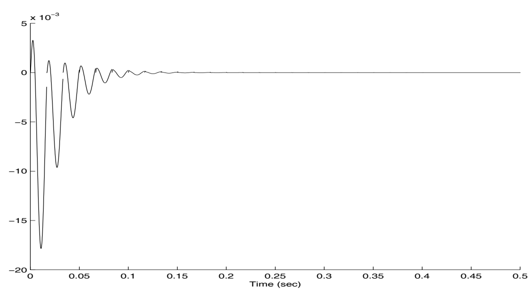

Fig. 11. PCCL scheme for RRO fundamental harmonics compensation

The simulation for RRO fundamental harmonics compensation is shown in Fig. 11. The frequency of the RRO fundamental harmonics $\omega_0 = 60$ Hz. In the simulation $K_{pcl} = K_{ccl} = 1.0 \times 10^{-3}$ and $T_{pcl} = T_{ccl} = 0.8 \times 10^{-4}$. The PCCL scheme is shown stable and convergence for these values chosen. The system converges in iteration domain and meanwhile improves performance in the time domain as well.

3.3 *Experiments*

The experiment set contains a Maxtor hard disk prototype (Model 51536U3), a Laser Doppler Vibrometer (LDV) and a TMS320 Digitial Signal Processor (DSP). In the experiment, the computer generated control signals and disturbance signals (RRO) are sent to the VCM in the HDD head positioning circuit by the DSP. The R/W head is driven by the motor and the head positioning signals are measured by LDV and sent back to the computer through DSP. In the experiment, we use computer generated RRO rather than the RRO in a real HDD in order to simplify the system disturbance, so that we can focus on the RRO elimination and consider either single or multiple harmonics. Due to the environment noise, the experimental results were analyzed in frequency domain.

Fig. 12 shows the Fast Fourier Transform (FFT) of a single RRO harmonics at 60 Hz. The amplitude of RRO is about 0.3 tracks. Fig. 13 shows the system output in frequency domain after 2 iterations learning. The RRO harmonics shown in Fig. 13 was reduced to 1/30 of its original amplitude.

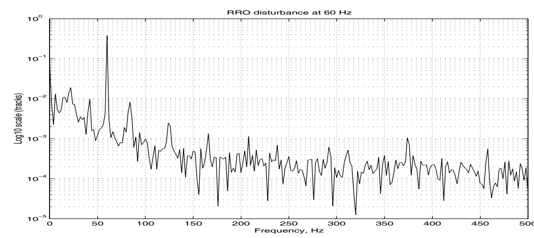

Fig. 12. RRO disturbance at 60 Hz

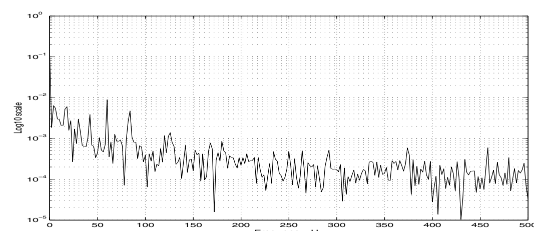

Fig. 13. After 2nd iteration learning

4. CONCLUSION

Stability and learning convergence are two most concerned problems when applying iterative learning control (ILC) schemes to HDD RRO compensations. In this paper, several types of ILC schemes have been analyzed. It is observed that the feedback loop of current cycle learning (CCL) scheme stabilizes the HDD system. The previous cycle learning (PCL) scheme can guarantee the convergence condition for RRO compensations. Therefore, a combination scheme, previous and current cycle learning (PCCL) scheme, has been employed. Its stability and convergency properties are verified through both the theoretical analysis and experiments.

5. APPENDIX

According to (10), define

$$1 - P_l(z)G_c(z)z = \frac{num(z)}{den(z)}$$

where,

$$
\begin{aligned}
\mathbf{den(z)} &= (z-1)^2(z-e^{-NT_s}) + \frac{\gamma T_s^2}{2}(z \\
&\quad + 1)[K_{ccl}(z-e^{-NT_s}) + T_{ccl}N(z-1)] \\
&= z^3 + (\frac{\gamma T_s^2}{2}K_{ccl} + \frac{\gamma T_s^2}{2}N\,T_{ccl} - 2 - e^{-NT_s})z^2 \\
&\quad + [\frac{\gamma T_s^2}{2}(1-e^{-NT_s})K_{ccl} + 1 + 2e^{-NT_s}]z \\
&\quad - (\frac{\gamma T_s^2}{2}e^{-NT_s}K_{ccl} + \frac{\gamma T_s^2}{2}N\,T_{ccl} + 2e^{-NT_s}); \\
\mathbf{num(z)} &= den(z) - \frac{\gamma T_s^2}{2}(z+1)[K_{pcl}(z \\
&\quad - e^{-NT_s}) + T_{pcl}N(z-1)]z \\
&= [1 - \frac{\gamma T_s^2}{2}(K_{pcl} + T_{pcl}N)]z^3 \\
&\quad + [\frac{\gamma T_s^2}{2}K_{ccl} + \frac{\gamma T_s^2}{2}N\,T_{ccl} \\
&\quad - 2 - e^{-NT_s} - \frac{\gamma T_s^2}{2}(1-e^{-NT_s})K_{pcl}]z^2 \\
&\quad + [\frac{\gamma T_s^2}{2}(1-e^{-NT_s})K_{ccl} + 1 \\
&\quad + 2e^{-NT_s} + \frac{\gamma T_s^2}{2}(e^{-NT_s}K_{pcl} + T_{pcl}N)z \\
&\quad - (\frac{\gamma T_s^2}{2}e^{-NT_s}K_{ccl} + \frac{\gamma T_s^2}{2}N\,T_{ccl} + 2e^{-NT_s}).
\end{aligned}
$$

Substituting values of N, T_s and γ, we have

$$
\begin{aligned}
&den(z) \\
&= z^3 + (1.5926K_{ccl} + 159.2639T_{ccl} - 2.9835)z^2 \\
&\quad + (0.02632K_{ccl} + 2.9669)z \\
&\quad - (1.5663K_{ccl} + 159.2639T_{ccl} + 0.9835) \\
&num(z) \\
&= (1 - 1.5926K_{pcl} - 159.2639T_{pcl})z^3 \\
&\quad + (1.5926K_{ccl} + 159.2639T_{ccl} - 2.9835 \\
&\quad - 0.02632K_{pcl})z^2 \\
&\quad + (0.02632K_{ccl} + 2.9669 + 1.5663K_{pcl} \\
&\quad + 159.2639T_{pcl})z \\
&\quad - (1.5663K_{ccl} + 159.2639T_{ccl} + 0.9835).
\end{aligned}
$$

Substituting $z = e^{j\omega T_s}$ and let

$$
\begin{aligned}
B2 &= 1.5926K_{ccl} + 159.2639T_{ccl} - 2.9835 \\
B1 &= 0.02632K_{ccl} + 2.9669 \\
B0 &= -(1.5663K_{ccl} + 159.2639T_{ccl} + 0.9835) \\
A3 &= -1.5926K_{pcl} - 159.2639T_{pcl} \\
A2 &= -2.9835 - 0.02632K_{pcl} \\
A1 &= 1.5663K_{pcl} + 159.2639T_{pcl}
\end{aligned}
$$

$$
\begin{aligned}
num(\omega) &= (1+A3)e^{j3\omega T_s} + (B2+A2)e^{j2\omega T_s} \\
&\quad + (B1+A1)e^{j\omega T_s} + B0 \\
den(\omega) &= e^{j3\omega T_s} + B2e^{j2\omega T_s} + B1e^{j\omega T_s} + B0
\end{aligned}
$$

Applying the convergence condition,
$|1 - P_l(z)G_c(z)z| \le \rho$,

$$
\begin{aligned}
& |num(z)|^2 \le \rho^2|den(z)|^2 \\
\Rightarrow\ & [(1+A3)\cos 3\omega T_s + (B2+A2)\cos 2\omega T_s \\
& + (B1+A1)\cos\omega T_s + B0]^2 \\
& + [(1+A3)\sin 3\omega T_s + (B2+A2)\sin 2\omega T_s \\
& + (B1+A1)\sin\omega T_s]^2 \\
\le\ & \rho^2(\cos 3\omega T_s + B2\cos 2\omega T_s + B1\cos\omega T_s + B0)^2 \\
& + \rho^2(\sin 3\omega T_s + B2\sin 2\omega T_s + B1\sin\omega T_s)^2
\end{aligned}
\tag{11}
$$

6. REFERENCES

Arimoto, S., S. Kawamura and F. Miyazaki (1984). Better operation of robots by learning. *Robotic Systems* **1(2)**, 123–140.

Bien, Z and J X Xu (1998). *Iterative Learning Control – Analysis, Design, Integration and Applications*. Kluwer Academic Press. Boston, USA.

Hwang, D.H., Z. Bien and S.R. Oh (1991). Iterative learning control method for discrete-time dynamic systems. *IEE Proc. Part-D* **138(2)**, 139–144.

Jian-Xin Xu, Tong Heng Lee and Heng-Wei Zhang (2001). Comparative studies on repeatable runout compensation using iterative learning control. *Proceedings of American Control Conference* pp. 2834–2839.

Jung-Ho Moon, Moon-Noh Lee and Myung Jin Chung (1996). Track-following control for optical disk drives using an iterative learning scheme. *IEEE Transactions on Consumer Electronics*.

Kurek, J.E. and M.B. Zaremba (1993). Iterative learning control synthesis based on 2-d system theory. *IEEE Trans. Automat. Controls.* **AC-38(1)**, 121–125.

Moore, Kevin L. (1993). *Iterative Learning Control for Deterministic Systems. - (Advances in Industrial Control Series)*. Springer-Verlag London Limited. London.

Saab, S.S. (1995b). A discrete-time learning control algorithm for a class of linear time-invariant systems. *IEEE Trans. of Automat. Control.* **AC-40(6)**, 1138–1141.

S.M. Zhu, F.H. Choo and K.S. Low (n.d.). Iterative learning control algorithm synthesis and its application to optical disk drive track-following control.

Tae-Jeong Jang, Chong-Ho Choi and Hyun-Sik Ahn (1995). Iterative learning control in feedback systems. *Automatica* **31 (2)**, 243–248.

Xu, J.X. (1997). Analysis of iterative learning control for a class of nonlinear discrete-time systems. *Automatica* **33(10)**, 1905–1907.