Local-Symmetrical-Integral-type Iterative Learning Control ⋆

Yangquan Chen, Huifang Dou and Kok Kiong Tan [a]

[a] *Dept. of Electrical Engineering,*
National University of Singapore,
10 Kent Ridge Crescent, Singapore 119260.
E-mails: yqchen@ieee.org; {eledouhf,eletankk}@nus.edu.sg

Abstract

A new iterative learning control (ILC) updating law is proposed for tracking control of continuous linear system over a finite time interval. The ILC is applied as a feedforward controller to the existing feedback controller. By using the local symmetrical integral of feedback control signal of previous iteration, the ILC updating law takes a simple form with only two design parameters: the learning gain and the range of local integration. Convergence analysis is presented together with a design procedure.

Key words: Iterative learning control; local symmetrical integration; convergence analysis; controller design and tuning.

1 Introduction

'Iterative Learning Control' (ILC) (Arimoto *et al.*, 1984) is a value-added block to enhance the feedback control performance by capitalizing the repetitiveness of system's operation. Clearly, the purpose of introducing the ILC is to utilize the system repetitions as *experience* to improve the system control performance even under incomplete *knowledge* of the system to be controlled.

Since early eighties, increasing attention has been drawn from the control community. Most of the existing work focused on the *analysis* issue of ILC schemes while the obtained convergence condition is clearly not enough for actual ILC

⋆ Corresponding author: Dr Yangquan Chen, Tel. +65-8743494. Fax: +65-7791103. E-mail: `yqchen@ieee.org`

Preprint submitted to Control Theory and Applications (China)25 February 1999

applications, e.g., (Chen *et al.*, 1998a; Xu *et al.*, 1995; Lee and Bien, 1997). Therefore, in recent years, increasing efforts have been made on the *design* issue of ILC. These can be observed from the latest book (Bien and Xu, 1998) and the dedicated ILC web server (Chen, 1998). Detailed literature review on ILC research can be found in (Moore, 1998; Chen, 1997). A recent survey on ILC *design* issue (Longman, 1998) documented various practically tested design schemes mainly for robotics applications where detailed descriptions can be found in (Elci, 1995). Other efforts in ILC design include using modern techniques like LMI (linear matrix inequality) (Doh *et al.*, 1998), H^∞ (Amann *et al.*, 1996), frequency domain LFT (linear fractional transformation) (Moon *et al.*, 1998), local ARMA (auto-regressive moving average) model approximation (Phan and Juang, 1996) etc. However, to draw the attention from the industry, the above mentioned design techniques are still not sufficiently attractive as compared to the successful use of PID (proportional-integral-derivative) controllers in industries. According to a survey (Yamamoto and Hashimoto, 1991) of the state of process control systems in 1989 conducted by the Japan Electric Measuring Instrument Manufacturer's Association, more than 90% of the control loops were of the PID type. It was also indicated (Bialkowski, 1993) that a typical paper mill in Canada has more than 2000 control loops and that 97% use PI control. Therefore, the industrialist had concentrated on PI/PID controllers and had already developed *one-button type* relay auto-tuning techniques for fast, reliable PI/PID control yet with satisfactory performance (Leva, 1993; Tan, 1994). In view of the above fact, the ILC design issue should also be attacked in a similarly rudimentary way of PI/PID controllers.

This paper makes some efforts in this direction trying to provide a PID-autotuning alike ILC design procedure with less modeling efforts. A new iterative learning control (ILC) updating law is proposed for tracking control of continuous linear system over a finite time interval. By using the local symmetrical integral (LSI) of feedback control signal of the previous iteration, the ILC updating law takes a simple form with only two design parameters: the learning gain and the range of local integration. Convergence analysis is presented together with a design procedure. Some practical considerations in the parameter tuning are also outlined.

The remainder parts of this paper are organized as follows. Sec. 2 formulates the control problem with a new learning updating scheme using local symmetrical integral of feedback signal. In Sec. 3, a convergence analysis is given. Sec. 4 details the design issues of the proposed ILC scheme. Finally, Sec. 5 concludes this paper.

2 Local-Symmetrical-Integral-type ILC

2.1 Feedforward-feedback Configuration

ILC, originally proposed as an open-loop control (Arimoto *et al.*, 1984), has been considered as a feedforward control in addition to an existing feedback controller. The feedforward-feedback configuration of ILC algorithms has already been a standard consideration in either '*analysis*' or '*design*' (Longman, 1998) work on ILC. A block-diagram is shown in Fig. 1 where **FBC** stands for "feedback controller" and y_d is the given desired output trajectory to be tracked. After the i-th iteration (repetitive operation), the feedforward control signal u^i_{ff} and the feedback control signal u^i_{fb} are to be stored in the memory bank for constructing the feedforward control signal at the next iteration, i.e., u^{i+1}_{ff}. The stored feedback control signal u^i_{fb} are to be locally symmetrically integrated (LSI) and multiplied by a learning gain γ.

Remark 2.1 System Class: *As suggested in (Longman, 1998), when ILC starts to have a substantial impact on how control is actually done in industry, it will be the linear based ILC that leads the way. In engineering practice, to design a control system, it is very fundamental to have a linear proximal model $G(s)$ for frequencies below a frequency of interest, say, ω_c. For a feedback controlled system, it is almost sure that its frequency response can be well approximated by a linear system's, i.e., $G_c(s)$, the closed-loop transfer function. Therefore, at this point, it is understood that, $G(s)$ in Fig. 1 is a linearly major part of the* plant *which may be nonlinear.*

2.2 LSI-type ILC Updating Law

The learning updating law is hence written by

$$u^{i+1}_{ff}(t) = u^i_{ff}(t) + \frac{\gamma}{2T_L}\int_{t-T_L}^{t+T_L} u^i_{fb}(\tau)\mathrm{d}\tau \tag{1}$$

while the overall control is simply that

$$u_{i+1}(k) = u^{i+1}_{ff}(k) + u^{i+1}_{fb}(k) \tag{2}$$

as shown in Fig. 1 where two parameters - γ the learning gain and T_L the width of the LSI, are to be designed and specified.

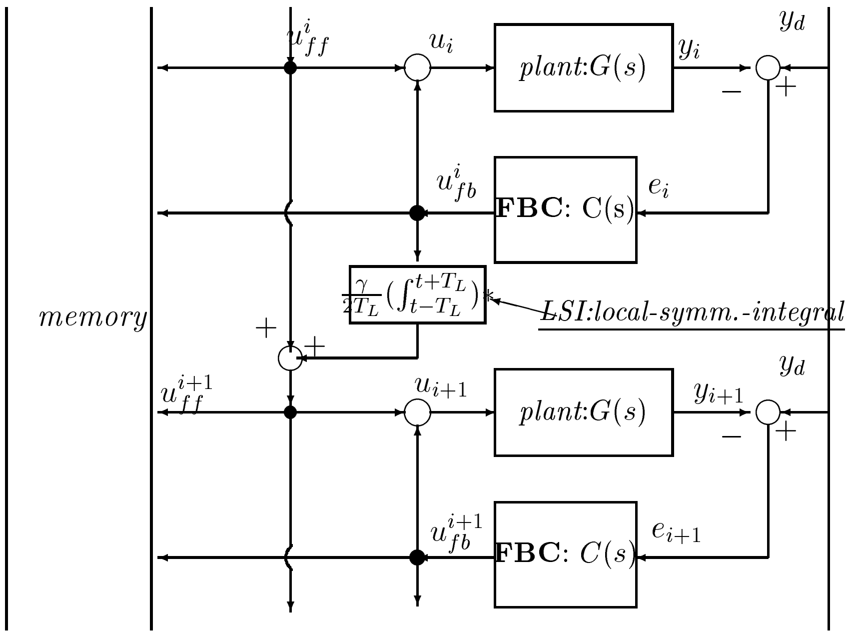

Fig. 1. Block Diagram of Local-Symmetrical-Integral-type (LSI) Iterative Learning Control

Remark 2.2 **When $T_L \to 0$**: *learning updating law (1) reduced to*

$$u_{ff}^{i+1}(t) = u_{ff}^{i}(t) + \gamma u_{fb}^{i}(t). \tag{3}$$

This was discussed in (Kuc et al.*, 1992).*

Remark 2.3 **'*Anticipatory*' Scheme**: *When the feedback controller $C(s)$ is of a PD (proportional plus derivative) type, i.e.,*

$$C(s) = K_p + K_d s,$$

(1) becomes

$$u_{ff}^{i+1}(t) = u_{ff}^{i}(t) + \gamma K_p \int_{t-T_L}^{t+T_L} e^i(\tau)\mathrm{d}\tau + \gamma K_d e^i(t+T_L) - \gamma K_d e^i(t-T_L).$$

*Clearly, this is what was noted as '*anticipatory*' scheme (Wang, 1998). It can be argued that the effectiveness of ILC method is due to the '*anticipatory*' or noncausal (Chen* et al.*, 1998b) use of the stored data of previous iterations.*

Remark 2.4 **Discrete-time Scheme**: *Consider that the sampling period is T_s. Let $T_L/T_s = M$. Using the simple rectangular formula of the numerical* *quadrature, one obtains*

$$u_{ff}^{i+1}(k) = u_{ff}^{i}(k) + \frac{\gamma}{2M+1} \sum_{j=-M}^{M} u_{fb}^{i}(k+j). \tag{4}$$

Clearly, (4) is simply an algebraic averager (Burdet et al.*, 1997) - a special type of noncausal filter discussed in (Chen* et al.*, 1998b).*

Here, our control task is to track on to the given desired output trajectory $y_d(t)$ over a fixed time interval $[0, T]$ as closely as possible. With an existing feedback controller $C(s)$, the main objective of this paper is to use a learning feed forward controller (LFFC) given by ILC updating law (1) to achieve a better tracking performance. In what follows, we will perform an analysis on ILC convergence and then present a practical procedure for ILC design.

3 Convergence Analysis

Before performing the convergence analysis of the proposed learning scheme, we should clarify the existence problem. That is, for a given $y_d(t)$, there exists a unique feedforward $u_{ff}^{\infty}(t)$ such that $y_\infty(t) \to y_d(t)$ for all $t \in [0, T]$.

The convergence of the proposed learning controller is in the sense that u_{ff}^{i} approaches to a fixed point signal as i increases and meanwhile, $y_i(t) \to y_d(t)$. This is summarized in the following theorem.

Theorem 1 *A linear system shown in Fig. 1 is controlled by a suitable feedback controller which performs a given task repeatedly. A locally-symmetrical-integral-type ILC scheme (1) is applied as a learning feedforward controller (LFFC). There exists a real constant γ and a positive $T_L (0 < T_L < T)$ such that the learning process is convergent and furthermore,*

$$\lim_{i \to \infty} U_{ff}^{i}(s) \to Y_d(s)/G(s). \tag{5}$$

where $U_{ff}^{i}(s) = \mathcal{L}[u_{ff}^{i}(t)]$ and $Y_d(s) = \mathcal{L}[y_d(t)]$. The convergence rate is given by

$$\rho(\omega, \gamma, T_L) \triangleq \mid 1 - \gamma H(\omega, T_L) G_c(j\omega) \mid < 1, \tag{6}$$

where $G_c(s)$ is the closed loop transfer function and $G_c(s) = C(s)G(s)/(1 + C(s)G(s))$.

Theorem 1 implies that the iterative learning controller is essentially applied to inverse the plant to be controlled in an iterative manner. Since linear system is considered in this paper, in the sequel, frequency domain notion is used. Using Laplace transformation, the updating law (1) becomes

$$U_{ff}^{i+1}(s) = U_{ff}^{i}(s) + \gamma H(\omega, T_L) U_{fb}^{i}(s) \tag{7}$$

where $U_{fb}^{i}(s) = \mathcal{L}[u_{fb}^{i}(t)]$, $s = j\omega$ and

$$H(\omega, T_L) = \frac{\sin(\omega T_L)}{\omega T_L}.$$

In (7), the following property of Laplace transformation is applied:

$$\mathcal{L}[\int_{t-T_L}^{t+T_L} u_{fb}(\tau)\mathrm{d}\tau/(2T_L)] = \frac{\sin(\omega T_L)}{\omega T_L} U_{fb}(s) \tag{8}$$

where the detailed derivation is given in Appendix A.

Now we proceed to present a proof of Theorem 1.

Proof: From Fig. 1, the feedback signal can be written as

$$U_{fb}(s) = -G_c(s) + G_c(s)Y_d(s)/G(s). \tag{9}$$

Learning updating law (7) becomes,

$$U_{ff}^{i+1}(s) = [1 - \gamma H(\omega, T_L)G_c(s)]U_{ff}^{i}(s) + \gamma H(\omega, T_L)G_c(s)Y_d(s)/G(s). \tag{10}$$

Iterating (10), one obtains

$$\begin{aligned} U_{ff}^{i+1}(s) = & [1 - \gamma H(\omega, T_L)G_c(s)]^{i} U_{ff}^{0}(s) \\ & + \{1 - [1 - \gamma H(\omega, T_L)G_c(s)]^{i+1}\} Y_d(s)/G(s). \end{aligned} \tag{11}$$

Since $H(\omega, T_L)$ and $G_c(s)$ are essentially with a low pass filter characteristics, it is clearly possible to choose a suitable γ such that (6) is true. In addition, T_L can be used to shape $\rho(\omega, \gamma, T_L)$ which is the convergence rate in (11). Therefore, there exists a design of γ and T_L such that (6) holds and from (11) $\lim_{i\to\infty} U_{ff}^{i}(s) \to Y_d(s)/G(s)$ and moreover, $y_i(t) \to y_d(t)$ for all $t \in [0, T]$ as $i \to \infty$. ∎

Remark 3.1 *From the above proof, it can be seen that the learning convergence is independent on* u_{ff}^0 *which means that the initial feedforward control can be chosen arbitrarily. However,* u_{ff}^0 *can practically be set to 0 because no prior knowledge on determining* u_{ff}^0 *is available.*

Remark 3.2 *It is implied in (7) that the initial condition of each iteration should be the same.*

4 Design Issues

4.1 Prior Knowledge

In this paper, it has been shown in the above that the design parameters of the proposed learning control scheme are only γ and T_L. One may argue that in the original D-type ILC scheme (Arimoto *et al.*, 1984), there is only one design parameter Γ.

To make a fair comparison between two control schemes, one must take into account many factors. Among these factors, the amount of prior knowledge assumed for the controller design is vital for the control scheme to survive. In (Arimoto *et al.*, 1984), the range of the first Markov parameter should be known *a priori*. This is an unusual and nonconventional requirement. Moreover, the derivatives of output tracking error are prone to noise amplification. As argued in (Longman, 1998), when faced to an actual system, one cannot assume zero knowledge available. In most engineering practice, it is quite common that the Nyquist curve information about the system is available. In this paper, the knowledge we used includes

- The frequency of the desired trajectory, which is less than a known frequency denoted by ω_d and $\omega_d < \omega_c$. ω_c is the systems cut-off frequency;
- An estimate of $G_c(j\omega_c)$ or $G_c(j\omega_d)$.

Clearly, the above knowledge is minimal for controller design.

4.2 Design Method for T_L

The basic intuition in choosing a T_L is that the T_L cannot be too large and meanwhile the T_L cannot be too small. Small T_L will bring in more high frequency signal components stored in the memory bank. These high frequency signal components may be accumulated due to different phase relationship from iteration to iteration. This is the major reason of the divergence for some ILC schemes which may be convergent at the initial iterations but as ILC runs, divergence can be observed in practical applications (Longman, 1998). Meanwhile, a too large T_L deteriorates the signal's low frequency components when smoothing out the high frequency components. Therefore, a suitably chosen T_L is very important.

A simple consideration is that, the signal's energy can not be attenuated by half via $H(\omega, T_L)$. One obtains

$$H(\omega_d, T_L) = \frac{\sqrt{2}}{2} \tag{12}$$

which will give an estimate of T_L by

$$\omega_d T_L \approx 1.392 \tag{13}$$

using `solve('sin(x)/x =sqrt(2)/2')` of MATLAB Symbolic Math Toolbox. Therefore,

$$T_L \geq \frac{1.392}{\omega_d}. \tag{14}$$

An alternative practical design procedure can be like this. As at the first iteration only feedback controller is commissioned, at the end of the first iteration, performing discrete Fourier transform (DFT) of the feedback signal $u_{fb}^0(t)$ gives the spectrum information of the feedback signal. A frequency ω_c' can be chosen a little bit higher than the frequency corresponding to the magnitude peak in the amplitude-frequency plot of $u_{fb}^0(t)$. Then, one can use this ω_c' to obtain a design of T_L from (14).

From the above discussions, T_L can now be designed.

Remark 4.1 *The learning scheme analyzed in (Kuc* et al.*, 1992) is actually the case when $T_L \to 0$ which caters for tracking desired trajectory with ultra high signal frequency, according to the discussion of this subsection. This is too stringent to be practically useful. Therefore, in practical use of learning control scheme like the one proposed in (Kuc* et al.*, 1992), a locally symmetrical integration is required with a suitable T_L.*

4.3 Design Method for γ

It is an open problem to get a reasonable estimate of the upper limit of γ with **less** modeling effort. However, during practice, one can always start with a smaller, conservative γ via which the learning process converges. Then fine tune of γ is still possible as discussed in Sec. 4.5. The above discussions indicate that it is an easy task to make ILC work.

Full knowledge of $G_c(j\omega), \quad \omega \leq \omega_c$ may be sometimes impractical. Therefore, it is assumed that at least the value of $G_c(j\omega_c)$ is available. In what follows, it will be shown that $G_c(j\omega_c)$ can be used to design a reasonable γ. Let $G(j\omega) = A(\omega)e^{\theta(\omega)}$. Denote $\bar{\rho} = \rho^2(\omega, \gamma, T_L)$. Then, from (6),

$$\bar{\rho} = 1 - 2\gamma H(\omega, T_L)A(\omega)\cos\theta(\omega) + \gamma^2 H^2(\omega, T_L)A^2(\omega). \tag{15}$$

Clearly, γ should be chosen to minimize $\bar{\rho}$. From (15), the best γ should be

$$\gamma = \frac{\cos\theta(\omega)}{H(\omega, T_L)A(\omega)} \tag{16}$$

by setting $\frac{\mathrm{d}\bar{\rho}}{\mathrm{d}\gamma} = 0$. At frequency ω_c,

$$\gamma = \frac{\cos\theta(\omega_c)}{H(\omega_c, T_L)A(\omega_c)}. \tag{17}$$

It should be noted that γ may be negative at certain frequency range. For most applications, ω_d is quite small and in this case γ can be given approximately by

$$\gamma \leq \sqrt{2}. \tag{18}$$

When only $G_c(j\omega_d)$ is known, γ can be design similarly according to (16). If the knowledge of $G_c(j\omega)$ within a frequency range $[\omega_L, \omega_H]$ is known. A plot of $\gamma(\omega)$ is available from (17). This plot is useful in selecting a suitable γ when different frequencies of interest are to be considered in $[\omega_L, \omega_H]$.

In any case, it is possible to tune γ to make the ILC convergence as fast as possible. However, as shown in Sec. 4.4, the ILC convergence rate has its limit governed by the closed loop system dynamics alone.

4.4 Discussion on A Limit of ILC Convergence Rate

Substituting (16) into (15), the corresponding minimal $\bar{\rho}$ is given by

$$(\bar{\rho})_{\gamma,\min} = \sin^2\theta(\omega), \tag{19}$$

i.e.,

$$\rho \geq |\sin \theta(\omega)| = \rho^*.$$

The above inequality implies that, for a given ω of interest, the ILC convergence rate cannot be faster than the limit characterized by ρ^*. This limit is independent of learning schemes applied. The only way to achieve a faster ILC convergence process is to well design the feedback controller $C(j\omega)$ such that the phase response of the closed-loop system will well behave as required in (19).

4.5 Some Heuristic Design Consideration

The following heuristic ideas may be helpful in tuning ILC parameters. When applicable, some rule-based methods could be used.

- Re-evaluate T_L at the end of every iteration. This will not cost a lot but can keep a tight monitoring of possible variations of the system dynamics and the uncertainty/disturbance.
- When ILC starts with a smaller γ, increase γ while the tracking error keep decreasing and decrease γ while the tracking error keep increasing.
- Use a cautious (larger) T_L at the beginning of ILC process. Decrease T_L when ILC converges to a stage with little improvement. In this case, smaller T_L leaves more high frequency components of the feedback control signal in the memory bank. This in turn may further improve the convergence performance.

5 Conclusions

A new iterative learning control (ILC) updating law is proposed for tracking control of continuous linear system over a finite time interval. The ILC is applied as a feedforward controller to the existing feedback controller. By using the local symmetrical integral of feedback control signal of previous iteration, the ILC updating law takes a simple form with only two design parameters: the learning gain and the range of local integration. Convergence analysis is presented together with a design procedure. Some practical considerations in the parameter tuning are also outlined. Also, a limit on the ILC convergence rate has been discussed.

References

Amann, N., Owens, D. H., Rogers, E., and Wahl, A. (1996). An H_∞ approach to linear iterative learning control design. *International Journal of Adaptive Control and Signal Processing*, **10**(6), 767–781.

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

Bialkowski, W. L. (1993). Dreams versus reality: A view from both sides of the gap. *Pulp and Paper Canada*, **94**(11).

Bien, Z. and Xu, J.-X. (1998). *Iterative Learning Control - Analysis, Design, Integration and Applications*. Kluwer Academic Publishers.

Burdet, E., Rey, L., and Codourey, A. (1997). A trivial method of learning control. In *Preprints of the 5th IFAC Symposium on Robot Control, volume 2*, Nante, France. IFAC.

Chen, Y. (1997). *High-order iterative learning control: convergence, robustness and applications*. Ph.D. thesis, Nanyang Technological University, Nanyang Avenue, Singapore 639798.

Chen, Y. (1998). Dedicated web server for Iterative Learning Control research. `http://ilc.ee.nus.edu.sg/`.

Chen, Y., Gong, Z., and Wen, C. (1998a). Analysis of a high order iterative learning control algorithm for uncertain nonlinear systems. *Automatica*, **34**(3), 345–353.

Chen, Y., Lee, T. H., Xu, J.-X., and Yamamoto, S. (1998b). Noncausal filtering based design of iterative learning control. In K. L. Moore, editor, *Proc. of the First Int. Workshop on Iterative Learning Control*, pages 63–70, Hyatt Regency, Tampa, FL, USA.

Doh, T.-Y., Jin, K. B., and Chung, M. J. (1998). An LMI approach to iterative learning control for uncertain linear systems. In *ISIAC*, pages 006.1–006.6, Albuquerque, NM, USA. TSI Press.

Elci, H. (1995). *The design and applications of learning control*. Ph.D. thesis, Columbia University.

Kuc, T.-Y., Lee, J. S., and Nam, K. (1992). An iterative learning control theory for a class of nonlinear dynamic systems. *Automatica*, **28**(6), 1215–1221.

Lee, H. S. and Bien, Z. (1997). A note on convergence property of iterative learning controller with respect to sup norm. *Automatica*, **33**(8), 1591–1593.

Leva, A. (1993). PID autotuning algorithm based on relay feedback. *IEEE Proc. Part-D*, **140**(5), 328–338.

Longman, R. W. (1998). "Designing Iterative Learning and Repetitive Controllers". In Z. Bien and Xu J.-X. eds, "*Iterative Learning Control - Analysis, Design, Integration and Application*", Kluwer Academic Publishers, pp. 107–145.

Moon, J.-H., Doh, T.-Y., , and Chung, M. J. (1998). A robust approach to iterative learning control design for uncertain systems. *Automatica*, **34**(8), 1001–1004.

Moore, K. L. (1998). Iterative learning control - an expository overview. *Applied & Computational Controls, Signal Processing, and Circuits*, **To appear**(Available at `http://132.205.2.222:888/~yqchen/ILC/Ilcrep.zip.gz`), 1–42.

Phan, M. and Juang, J. (1996). Designs of learning controllers based on an auto-regressive representation of a linear system. *AIAA Journal of Guidance, Control, and Dynamics*, **19**(2), 355–362.

Tan, K. K. (1994). *Towards autonomous process control with relay-based feedback identification*. Ph.D. thesis, National University of Singapore, 10 Kent Ridge Crescent.

Wang, D. (1998). An anticipatory iterative learning control scheme: theory and experiments. In K. L. Moore, editor, *Proc. of the First Int. Workshop on Iterative Learning Control*, pages 79–80, Hyatt Regency, Tampa, FL, USA.

Xu, J.-X., Wang, X.-W., and Lee, T. H. (1995). Analysis of continuous iterative learning control systems using current cycle feedback. In *Proc. of American Control Conference*, pages 4221–4225, Seattle, Washington, USA.

Yamamoto, S. and Hashimoto, I. (1991). Resent status and future needs: The view from japanese industry. In Arkun and Ray, editors, *Proceedings of the fourth International Conference on Chemical Process Control*, Texas. Chemical Process Control – CPCIV.

A Derivation of $\mathcal{L}[\int_{t-T_L}^{t+T_L} u_{fb}(\tau)\mathrm{d}\tau/(2T_L)]$

First, let $U_{fb}(s) \triangleq \mathcal{L}[u_{fb}(t)]$. Using the rectangular rule for integration, one obtains

$$\int_{t-T_L}^{t+T_L} u_{fb}(\tau)\mathrm{d}\tau = \lim_{m\to\infty} \sum_{i=0}^{m-1} u_{fb}(t - T_L + \frac{2T_L}{m} i)\frac{2T_L}{m}. \tag{A.1}$$

Applying Laplace transformation of both sides of (A.1) gives

$$\begin{aligned}
\mathcal{L}[\int_{t-T_L}^{t+T_L} u_{fb}(\tau)\mathrm{d}\tau/(2T_L)] &= U_{fb}(s)e^{-sT_L} \lim_{m\to\infty} \frac{1}{m} \sum_{i=0}^{m-1} e^{\frac{2T_L s}{m} i} \\
&= U_{fb}(s)\frac{e^{sT_L} - e^{-sT_L}}{2T_L s} \lim_{m\to\infty} \frac{2sT_L/m}{e^{2sT_L/m} - 1} \\
&= \frac{\sin(\omega T_L)}{\omega T_L} U_{fb}(s).
\end{aligned} \tag{A.2}$$