Iterative Learning Control and Design II

Professor Eric Rogers

Electronics and Computer Science
University of Southampton
Southampton, SO17 1BJ, UK
etar@ecs.soton.ac.uk
http://www.ecs.soton.ac.uk/
Contents

- PD ILC Design
- Plant Inversion ILC Design
- $H_\infty$ ILC Design
- Optimal Control
- Norm Optimal ILC (NOILC)

Acknowledgement: The material in this section, apart from norm optimal approach, largely follows that in the following paper.

D. A. Bristow, M. Tharayil and A. G. Alleyne
A Survey of Iterative Learning Control
Basics

• In practical terms, ILC design aims to generate an open-loop signal that approximately inverts the plant’s dynamics and track the reference and reject repeating disturbances.

• Ideally, ILC only learns repeating disturbances and ignores noise and nonrepeating disturbances.

• In essence ILC is open-loop control but combing with feedback action has advantages — see earlier.

• In this section of the course, we consider 4 design methods and illustrate with experimentally measured results.

• We start with PD-type designs with tuning that can be applied to a system without extensive modeling and analysis.
PD ILC Design
PD ILC Design

• Arimoto’s original algorithm was PD-type. In discrete-time the corresponding law is

\[ u_{k+1}(p) = u_k(p) + k_p e_k(p + 1) + k_d [e_k(p + 1) - e_k(p)] \quad (1) \]

• An alternative is to use \( e_k(p) \) instead of \( e_k(p + 1) \) in the second term on the right-hand side of (1).

• From the analysis of the previous section, an ILC system with PD-type applied is AS if and only if,

\[ |1 - (k_p + k_d)p_1| < 1 \quad (2) \]

• If \( p_1 \) is known then it is always possible to find \( k_p \) and \( k_d \) such that AS holds.
Monotonic trial-to-trial error convergence is not always possible with ILC PD-type.

The most commonly used approach to achieving monotonic convergence is to modify the law to include a low-pass $Q$ filter. Such a filter can be used to block learning at high frequencies and this is of use in satisfying the robust monotonic error condition (last theorem in the previous set of notes). The $Q$ filter also has benefits in terms of added robustness and high-frequency noise filtering.

The most common method for selecting the PD gains is by tuning.

When a $Q$ filter is used the filter type — Butterworth, Chebyshev etc — and order are specified and the filter bandwidth used as the tuning variable.
PD ILC Design

• Unlike feedback PID (or three term control) Ziegler Nichols type tuning rules are not available for ILC.
• Intuitive guidelines for tuning to achieve good learning transients and low error are as follows.
• For each set of gains, the learning is reset and run for sufficient trials to determine transient behaviour and $e_\infty$. The gains $k_p$ and $k_d$ and filter bandwidth are set low.
• After stable baseline transient performance and error performance have been achieved, the gains and bandwidth can be increased.
• Depending on the application, the learning gains influence the rate of convergence and the $Q$ filter the converged error performance.
PD ILC Design

• Increasing the $Q$ filter bandwidth decreases robustness but improves performance. Decreasing the bandwidth has the opposite effect.

• As noted in the last section, large transients can arise very quickly!!

• A two-step method is possible if (by experiments) the system frequency response is available.

• This is based on satisfying the AS and monotonic convergence criterion of the previous section (last theorem), written as

$$|1 - e^{j\theta} L(e^{j\theta}) G(e^{j\theta})| < \frac{1}{|Q(e^{j\theta})|}$$

for all $\theta \in [-\pi, \pi]$. 
PD ILC Design

- Using a Nyquist plot of $e^{j\theta} L(e^{j\theta}) G(e^{j\theta})$ the learning gains can be tuned to maximize the range $\theta \in [0, \theta_c]$ over which this plot lies inside the unit circle with center 1 and the $Q$ filter bandwidth selected to satisfy the stability condition.
Plant Inversion ILC Design
Plant Inversion ILC Design

- As its name suggests, this class of methods use models of the inverse plant dynamics as the learning function. In discrete-time the resulting law is

\[ u_{k+1}(p) = u_k(p) + \hat{G}^{-1}(q)e_k(p) \]  

(4)

- or

\[ u_{k+1}(p) = u_k(p) + q^{-1}\hat{G}^{-1}(q)e_k(p + 1) \]  

(5)

The learning function is

\[ L(q) = q^{-1}\hat{G}^{-1}(q) \]  

(6)

which is causal and of zero relative degree (same number of zeros and poles).
Plant Inversion ILC Design

• If $\hat{G}(q)$ is an exact model of the plant then convergence occurs in one trial and $e_\infty = 0$.

• Fact: The inverse of a non-minimum phase plant is unstable.

• The alternative is to use a stable inversion approach, which will give a non-causal learning function.

• In any case, the success of the plant inversion method ultimately depends on the accuracy of the plant model.

• Consider again the uncertainty description of the previous section, i.e.,

$$G(q) = \hat{G}(q)[1 + W(q)\Delta(q)]$$

(7)

where $\hat{G}(q)$ is the nominal model, $W(q)$ is known and stable and $\Delta(q)$ is unknown but stable with $\|\Delta(z)\|_\infty < 1$. 

Professor Eric Rogers
ECS, University of Southampton, UK
Plant Inversion ILC Design

• Monotonic convergence with a rate better than $\gamma^*$ holds if

$$|W(e^{j\theta})| < \gamma^*, \ \theta \in [-\pi, \pi]$$  \hspace{1cm} (8)

• Consider a given $\theta_0$. Then $|W(e^{j\theta_0})| \geq 1$, i.e., an uncertainty greater than 100%, means robust monotonic convergence is not possible.

• Model uncertainty of greater than 100% is common in applications at high frequencies.

• To avoid(?) the resulting poor transients a low-pass $Q$ filter is often used.

• If the $Q$ filter cutoff frequency is set sufficiently low, the uncertainty causing the problem is blocked.
$H_\infty$ ILC Design
$H_\infty$ ILC Design

- $H_\infty$ based ILC design is also possible — find the learning function $L(q)$ that offers the fastest convergence for a given $Q$ filter.
- Alternatively — solve the model matching problem

$$L^*(z) = \arg\min_L \|Q(z)(I - z L(z)G(z))\|_\infty$$  \hspace{1cm} (9)

With the notation $(z)$ dropped for ease of presentation, the above problem can be equivalently written as a lower linear fractional transform

$$Q(I - zL G) = G_{11} + G_{12} L(I - G_{22} L)^{-1} G_{21} = \mathcal{F}_L(G, L)$$  \hspace{1cm} (10)

where

$$\begin{bmatrix} G_{11} & G_{12} \\ G_{21} & G_{22} \end{bmatrix} = \begin{bmatrix} Q & Q \\ -zG & 0 \end{bmatrix}$$
$H_\infty$ ILC Design

- $H_\infty$ synthesis tools can now be employed.
- The analysis extends to allow models with known uncertainty bounds.
- This is now a robust performance problem to which $\mu$ synthesis tools can be applied.
- Finding a solution with $\mu < 1$ guarantees robust monotonic convergence.
- There are many other variations and approaches.

S. Skogestad and I. Postlethwaite
Optimal Control
Optimal Control

- A natural model based approach is to control law design is to formulate as an optimal control problem.
- We first consider $Q$ optimal design in the lifted setting.
- The cost function is of the form

$$J(u_{k+1}) = e^T_{k+1} Q_L Q e_{k+1} + u^T_{k+1} R_L Q u_{k+1}$$
$$+ \delta^T_{k+1} S_L Q \delta_{k+1}$$

$$\delta_{k+1} = u_{k+1} - u_k$$

- $Q_L$ is an $T \times T$ positive-definite matrix.
- $R_L$ and $S_L$ are $T \times T$ positive semi-definite matrices.
Optimal Control

• Solution

\[ Q_{OPT} = \left(G^T Q_{LQ} G + R_{LQ} S_{LQ}\right)^{-1} \left(G^T Q_{LQ} G + S_{LQ}\right) \]  (12)

• Optimal value of cost function

\[ I_{OPT} = \left(G^T Q_{LQ} G + S_{LQ}\right)^{-1} G^T Q_{LQ} \]  (13)

• The converged error is

\[ E_{\infty, opt} = \left[I - G \left(G^T Q_{LQ} G + R_{LQ}\right)^{-1} G^T Q_{LQ}\right] \left(Y_r - d\right) \]  (14)

• The weighting on the change in the control input \( S_{LQ} \), which affects the speed of convergence has no effect on the asymptotic error but the weighting on the control \( R_{LQ} \) does.

• If \( R_{LQ} = 0 \) then the error converges to zero but \( R_{LQ} \neq 0 \) may be useful to limiting control action (and hence prevent damage to the actuators).
Norm Optimal ILC (NOILC)
There is an alternative solution of the optimal control problem.
This approach simultaneously yields the ILC law and a feedback controller.
Deriving this algorithm in an abstract setting using Hilbert spaces also gives systems theoretic insight to the problem.
The general ILC problem is easily seen to be equivalent to finding the minimizing input $u_\infty$ for the optimization problem:

$$\min_u \{ ||e||^2 : e = r - y, \; y = Gu \}$$

Gradient based ILC algorithms generate $u_{k+1}$ as

$$u_{k+1} = u_k + \epsilon_{k+1} G^* e_k$$

$G^*$ is the adjoint operator of $G$ and $\epsilon_{k+1}$ is a step length to be chosen on each trial.
This approach suffers from the need to choose $\varepsilon_{k+1}$ and its feedforward structure takes no account of current trial effects, e.g., disturbances, plant modelling errors.

Improved approach — NOILC.

NOILC has the following important properties:

- Automatic choice of step size.
- Potential for improved robustness through use of causal feedback (current trial error data) and feedforward of data from previous trials.
Chose current trial input as

\[ u_{k+1} = \arg \min_{u_{k+1} \in \mathcal{U}} \{ J_{k+1}(u_{k+1}) \} \]

where

\[ J_{k+1}(u_{k+1}) = \| e_{k+1} \|_{\mathcal{Y}}^2 + \| u_{k+1} - u_k \|_{\mathcal{U}}^2 \]

and

\[ e_{k+1} = r - y_{k+1} \]

\[ y_{k+1} = Gu_{k+1} \]

Initial control — \( u_0 \in \mathcal{U} \) can be arbitrary but a good first guess at a solution would be better. \( \mathcal{U} \) and \( \mathcal{Y} \) are appropriately chosen Hilbert spaces. The designations of these spaces are deleted from this point onwards for ease of notation.
NOILC — Analysis

This approach determines $u_{k+1}$ with the properties that:
- the tracking error is reduced in an optimal manner;
- the new control input does not deviate too much from the control input used on the previous trial.

The relative weighting of these two objectives can be absorbed into the definitions of the norms on $\mathcal{Y}$ and $\mathcal{U}$.

Immediate benefit (interlacing result)

$$\|e_{k+1}\|^2 \leq J_{k+1}(u_{k+1}) \leq \|e_k\|^2$$

- The choice $u_{k+1} = u_k$ is suboptimal and therefore at optimality

$$\|e_{k+1}\|^2 + \|u_{k+1} - u_k\|^2 \leq \|e_k\|^2$$

Hence

$$\|e_{k+1}\|^2 \leq \|e_k\|^2$$
NOILC — Analysis

- The algorithm generates bounded (in norm) sequences but it does not immediately imply that the error converges to zero.
- The abstract solution is

\[ u_{k+1} = u_k + G^* e_{k+1} \]  \hspace{1cm} (17)

and the formal error evolution is

\[ e_{k+1} = (I + GG^*)^{-1} e_k \]  \hspace{1cm} (18)

- Questions
  - Does this sequence converge to zero or any other signal?
  - How can the convergence rate be influenced?
Suppose that
\[ \langle v, G G^* v \rangle \geq \sigma^2 \| v \|^2, \quad \forall v \in \mathcal{Y}, \quad \sigma > 0 \]  

Then
\[ \| e_{k+1} \| \leq \frac{1}{1 + \sigma} \| e_k \| \]  

Hence convergence to zero error is guaranteed and is geometric.

Technical point: In the case of continuous-time dynamics, the result is ‘almost’ geometric convergence to zero error.
NOILC — Analysis

• The abstract solution

\[ u_{k+1} = u_k + G^* e_{k+1} \]

is noncausal and hence cannot be implemented.

• An equivalent implementable formulation exists for linear systems

• Plant model continuous-time case — state space triple \{A, B, C\}

• Cost Function

\[
J_{k+1}(u_{k+1}) = \frac{1}{2} \int_0^T \{e_{k+1}^T(t)Qe_{k+1}(t) + (u_{k+1}(t) - u_k(t))^T \times R(u_{k+1}(t) - u_k(t))\} dt + \frac{1}{2} e_{k+1}^T(T)Fu_{k+1}(t)
\]
NOILC — Analysis

Update equation

\[ u_{k+1}(t) = u_k(t) + R^{-1}B^T \left[ -K(t)(x_{k+1}(t) - x_k(t)) + \zeta_{k+1}(t) \right] \]

Riccati Equation

\[ \dot{K}(t) = -A^T K(t) - K(t)A + K(t)BR^{-1}B^T K(t) - C^T QC \]
\[ K(T) = C^T FC \]

Predictive Component

\[ \dot{\zeta}_{k+1}(t) = -(A - BR^{-1}B^T K(t))^T \zeta_{k+1}(t) - C^T Qe_k(t) \]

with terminal boundary condition

\[ \zeta_{k+1}(T) = C^T F \zeta_k(T) \]
Predictive NOILC

\[ J(u_{k+1}, \lambda) = \sum_{i=1}^{M} \lambda^{i-1} (\|e_{k+i}\|^2 + \|u_{k+i} - u_{k+i-1}\|^2) \quad (21) \]

- \( M \)-‘step’ ahead prediction.
- This form of predictive control requires \( M - 1 \) copies of the plant model in implementation.
- The convergence rate of the algorithm increases with \( M \) and for \( \lambda > 1 \).

\[ \|e_{k+1}\| \leq \frac{1}{\lambda} \|e_k\| \quad (22) \]

and hence a plant independent convergence rate.
Parameter-Optimal ILC

• This gives a simpler implementation.
• The algorithm is as follows and has a fixed (feedforward) structure

\[ u_{k+1} = u_k + \gamma_{k+1}Ke_k \]  (23)

where \( K \) is a fixed operator and \( \gamma_{k+1} \) is a variable adaptive learning gain/parameter.

• The resulting error evolution equation is

\[ e_{k+1} = (I - \gamma_{k+1} GK)e_k \]  (24)

• Hence

\[
\|e_{k+1}\|^2 = \|e_k\|^2 - 2\gamma_{k+1}e_k^T GKe_k \\
+ \gamma_{k+1}^2 e_k^T K^T G^T GKe_k
\]  (25)
Parameter-Optimal ILC

• For monotonic convergence the term $-2\gamma_{k+1} e_k^T G K e_k$ must be sign-definite.

• ‘Natural’ choices for $K$ are

\[
K = I, \quad K = G^{-1}, \quad K = G^T
\]  

(26)

• In the case of $K = I$ and sampled data systems, we can think of $G$ as a matrix and $e$ as a time series vector of point-wise error values and in this case

\[
-2\gamma_{k+1} e_k^T G K e_k = -\gamma_{k+1} e_k^T (G + G^T) e_k
\]  

(27)

• A necessary condition for monotonic convergence is that $G + G^T$ is a positive-definite matrix.
Parameter-Optimal ILC

- **Fact:** Let $zG(z)$ be a transfer-function that generates $G$. Then if $zG(z)$ is a positive-real system, $G + G^T$ is a positive-definite matrix.
- Positivity is a benefit but also a constraint on the system dynamics.
- Positivity can be achieved by, as one method, low sampling frequency.
- For monotonic convergence, $\gamma_{k+1}$ has to be tuned properly.

\[
u_{k+1} = \arg \min_{u_{k+1} \in \mathcal{R}} \left\{ J_{k+1}(\gamma_{k+1}) \right\}
\]

where

\[
J_{k+1}(\gamma_{k+1}) = \|e_{k+1}\|^2 + w\gamma_{k+1}^2
\]
Parameter-Optimal ILC

- The optimal learning gain is

\[ \gamma_{k+1} = \frac{e^T G e_k}{w + e^T G^T G e_k} \]  \hspace{1cm} (28)

- Convergence properties under the positivity assumption

\[ \lim_{k \to \infty} \gamma_k = 0, \quad ||e_{k+1}|| < ||e_k||, \quad e_k \neq 0, \quad \gamma_{k+1} \neq 0 \]

\[ \lim_{k \to \infty} e_k = 0 \]  \hspace{1cm} (29)
Parameter-Optimal ILC

- If the plant is not positive, the algorithm will still converge monotonically, but possibly to a non-zero vector.
- The set of all such limit vectors is given by the solutions of equations of the form

\[ e^T(G + G^T)e = 0, \quad ||e|| \leq ||e||_0 \]  

(30)

- Similar results exist for \( K = G^{-1} \) and \( K = G^T \).
- Bonus of more complex choices: the positivity of \( G \) is not required for monotonic convergence to zero.
- Similar results exist for HOILC

\[ u_{k+1} = u_k + \sum_{i=1}^{M_1} \alpha_{k+1,i} u_{k-i} + \sum_{i=1}^{M_2} \beta_{k+1,i} e_{k+1-i} \]  

(31)
NOILC References

The theory of the NOILC approach is given in the following papers.

1. N. Amann, D. H. Owens and E. Rogers
   Iterative learning control using optimal feedback and feedforward actions.

2. N. Amann, D. H. Owens and E. Rogers
   Iterative learning control for discrete time systems with exponential rates of rate convergence.

3. N. Amann, D. H. Owens and E. Rogers
   Predictive optimal iterative learning control.