# IEEE ICMA 2006 Tutorial Workshop:

# – Iterative Learning Control –
# Algebraic Analysis and Optimal Design

**Presenters**:

Kevin L. Moore – Colorado School of Mines
YangQuan Chen – Utah State University

**Contributor**:

Hyo-Sung Ahn – ETRI, Korea

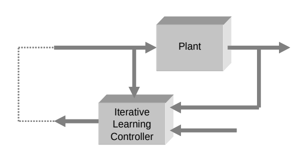
**IEEE 2006 International Conference on Mechatronics and Automation**
LuoYang, China

25 June 2006

# Outline

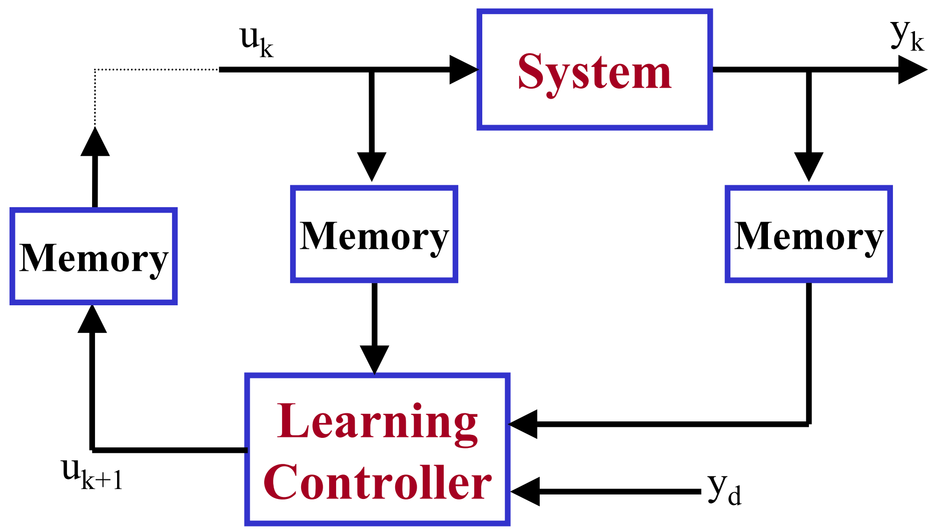
- **Iterative Learning Control (ILC)**
- Monotonic Convergence via Supervector Framework
- Current-Cycle Feedback Approach
- Non-Causal Filtering ILC Design
- Time-Varying ILC Design
- LMI Approach to ILC Design



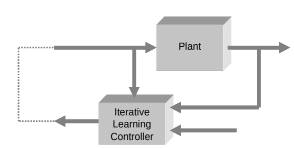


# ILC - A Control Approach Based on Intuition

- Humans gain "skill" from doing the same thing over and over.
- ILC seeks to achieve the same effect in the case when a machine performs the same task repeatedly.




- Goal is to pick next input $u_{k+1}(t)$ to improve next output response $y_{k+1}(t)$ relative to desired response $y_d(t)$, using all past inputs and outputs.
- Assume $y_d(0) = y_k(0)$ for all $k$, $t \in [0, N]$, and system is linear, discrete-time, and has relative degree one.



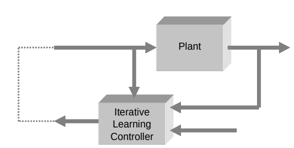
# What Information can be Included in the ILC Update?

- Most generally, we can allow:

$$\begin{aligned} u_{k+1}(t) \;=\; f\{ & u_0(t'), u_1(t'), \ldots, u_k(t'), \\ & e_1(t')), e_2(t'), \ldots, e_k(t'), \\ & \quad u_{k+1}(0), u_{k+1}(1), \ldots, u_{k+1}(t-1) \\ & \quad e_{k+1}(1), e_{k+1}(2), \ldots, e_{k+1}(t-1)\} \end{aligned}$$

  where $t' \in [0, N]$.

- That is, in general we can update $u_{k+1}(t)$ using:
  1. Information from all previous trials:
     $\Rightarrow$ Call this "**higher-order in iteration**" if more than one-trial back is used.
  2. Information from the entire time duration of any previous trial:
     $\Rightarrow$ Call this "**higher-order in time**" if filtering is done rather than using a single time instance.
     $\Rightarrow$ Note this allows non-causal signal processing – a key reason ILC works.
  3. Information up to time $t-1$ on the current trial:
     $\Rightarrow$ Call this "**current cycle feedback**."



# Higher-Order vs. First-Order

- Is there any reason to use higher-order ILC algorithms (in time or in iteration)?
- Maybe? Because of convergence? No, consider:

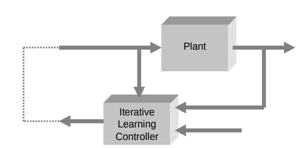
1. Classical Arimoto D-type ILC (for relative degree 1):

$$u_{k+1}(t) = u_k(t) + \gamma \frac{\mathrm{d}}{\mathrm{d}t} e_k(t)$$

2. PID-type ILC:

$$u_{k+1}(t) = u_k(t) + k_P e_k(t) + k_I \int_0^t e_k(\tau)\mathrm{d}\tau + \gamma \frac{\mathrm{d}}{\mathrm{d}t} e_k(t)$$





# Higher-Order vs. First-Order (cont.)

- For both, the convergence condition is that

$$|1 - \gamma h_1| < 1$$

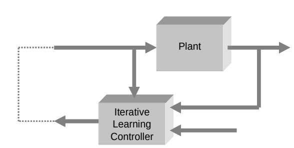
  where $h_1$ is the first Markov (non-zero) parameter. This ensures $e_k(t) \to 0$ as $k \to \infty \ \forall t$.

- Note: does not involve $k_P$, $k_I$, or with the system matrix $A$, either!

- That is, first-order in time and iteration is adequate to realize convergence.

- Something must be missing ...

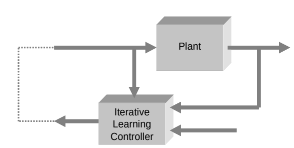
- The answer is: "how the convergence is achieved."



# Outline

- Iterative Learning Control (ILC)
- **Monotonic Convergence via Supervector Framework**
- Current-Cycle Feedback Approach
- Non-Causal Filtering ILC Design
- Time-Varying ILC Design
- LMI Approach to ILC Design





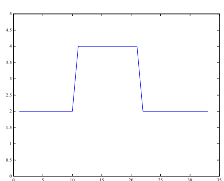
# Two Examples

- Consider two systems, each stable, minimum phase, with the same ILC update law

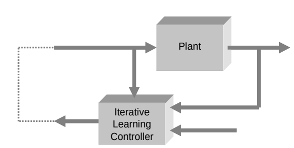
$$u_{k+1}(t) = u_k(t) + 0.9e_k(t+1)$$

1. $y_k(t+1) = -.2y_k(t) + .0125y_k(t-1) + u_k(t) - 0.9u_k(t-1)$
2. $y_k(t+1) = -.2y_k(t) + .0125y_k(t-1) + u_k(t) + 0.1u_k(t-1)$

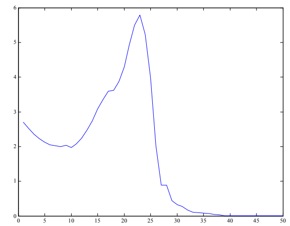
- Each is asked to track the following signal:



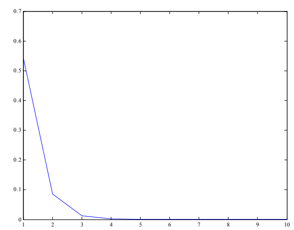
- The convergence condition guarantees that both converge, but ...



System 1 does not converge monotonically (in 2-norm):



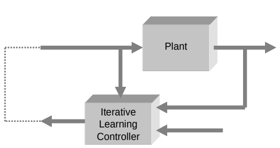
System 2 does converge monotonically (in 2-norm):



Question: Why do the two systems learn differently?

# Comments

- In the literature it has been shown that ILC achieves monotonic convergence for the $\lambda$-norm (time-weighted-norm) of the tracking error.
- However, in general the $\infty$-norm and 2-norm will often increase to a huge value before converging.
- Such ILC transients are typically not acceptable!
- It is not enough to ensure that $e_k(t) \to 0$ as $k \to \infty$. Rather, we would like the convergence to be monotonic.
- And, the norm topology should be physically meaningful.

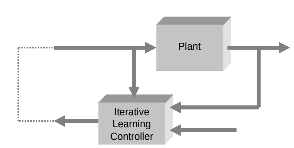


# Comments (cont.)

- Our study of convergence shows that "higher-order-in-time" algorithms, that is, proper design of the ILC update filters or algorithms, can give monotonic convergence through:

  1. Higher-order-in-time (causal) current-cycle feedback.
  2. Non-causal filtering of the error from the previous trial.
  3. Time-varying ILC gains.

- We study these problems using "supervector" notation and in terms of the system Markov parameters.



# Framework to Discuss Monotone Convergence

- Consider SISO discrete-time LTI system (relative degree 1):

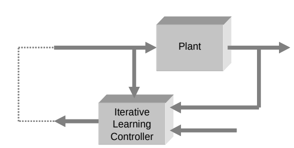
$$Y(z) = H(z)U(z) = (h_1 z^{-1} + h_2 z^{-2} + \cdots)U(z)$$

- Assume the standard ILC reset condition: $y_k(0) = y_d(0) = y_0$ for all $k$.

- Define the "supervectors:"

$$\begin{aligned} U_k &= [u_k(0), u_k(1), \cdots, u_k(N-1)]^T \\ Y_k &= [y_k(1), y_k(2), \cdots, y_k(N)]^T \\ Y_d &= [y_d(1), y_d(2), \cdots, y_d(N)]^T \\ E_k &= [e_k(1), e_k(2), \cdots, e_k(N)]^T \end{aligned}$$



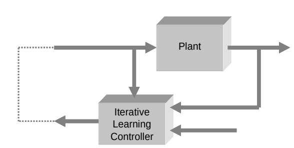


# Framework to Discuss Monotone Convergence (cont.)

- Then the system can be written as $Y_k = H_p U_k$ where $H_p$ is the matrix of Markov parameters of the plant, given by

$$H_p = \begin{bmatrix} h_1 & 0 & 0 & \ldots & 0 \\ h_2 & h_1 & 0 & \ldots & 0 \\ h_3 & h_2 & h_1 & \ldots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ h_N & h_{N-1} & h_{N-2} & \ldots & h_1 \end{bmatrix}$$

- To simplify our presentation, introduce the operator $T$ to map the vector $h = [h_1, h_2, \cdots, h_N]'$ to a lower triangular Toeplitz matrix $H_p$, i.e., $H_p = T(h)$.



## Framework to Discuss Monotone Convergence (cont.)

- Suppose we have a general higher-order ILC algorithm of the form:

$$u_{k+1}(t) = u_k(t) + L(z)e_k(t+1)$$

where $L(z)$ is a linear (possibly non-causal) filter.

- Then we can represent this ILC update law using supervector notation as:

$$U_{k+1} = U_k + LE_k$$

where $L$ is a Toeplitz matrix of the Markov parameters of $L(z)$.

- For instance, for the Arimoto-type discrete-time ILC algorithm given by

$$u_{k+1}(t) = u_k(t) + \gamma e_k(t+1)$$

where $\gamma$ is the constant learning gain, we have $L = diag(\gamma)$.

# Monotonic Convergence Condition

- For the Arimoto-update ILC algorithm, the ILC scheme converges (monotonically) if the induced operator norm satisfies:

$$\|I - \gamma H_p\|_i < 1.$$

- Likewise, a NAS for convergence is:

$$|1 - \gamma h_1| < 1.$$

- Combining these, we can show that for a given gain $\gamma$, convergence implies monotonic convergence in the $\infty$-norm if

$$|h_1| > \sum_{j=2}^{N} |h_j|.$$

- Note this condition is independent of $\gamma$, but instead puts restrictions on the plant.

# Higher-Order-in-Time Design for Monotone Convergence

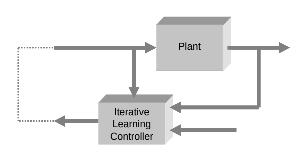
Using the monotonic convergence condition, we have derived ILC algorithm designs using higher-order time-domain filtering to achieve monotonic convergence three ways:

1. Higher-order-in-time (causal) current-cycle feedback .

2. Non-causal filtering of the error from the previous trial (optimal design of $L$ for PD-type ILC).

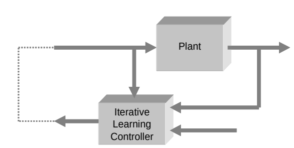
3. Time-varying ILC gain.





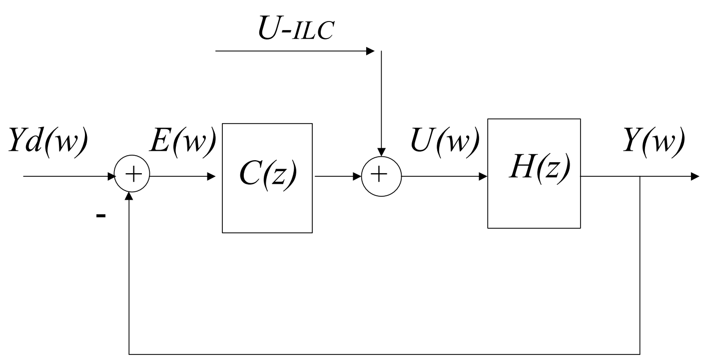
# Outline

- Iterative Learning Control (ILC)
- Monotonic Convergence via Supervector Framework
- **Current-Cycle Feedback Approach**
- Non-Causal Filtering ILC Design
- Time-Varying ILC Design
- LMI Approach to ILC Design



# Method 1: Current-Cycle Feedback

- Case A:

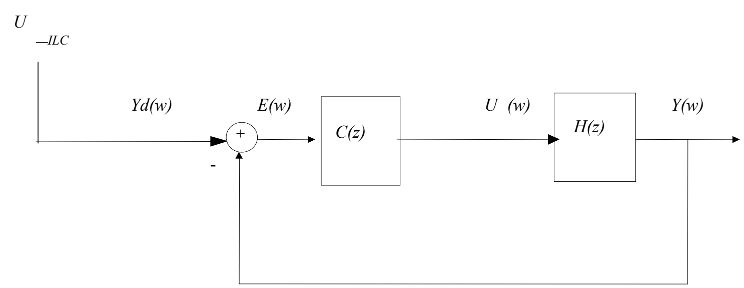



- Plant seen by the ILC algorithm:

$$H_{cl}^{A} = \frac{H(z)}{1 + C(z)H(z)}.$$

# Method 1 (cont.)

- Case B:




- Plant seen by the ILC algorithm:

$$H_{cl}^{B} = \frac{C(z)H(z)}{1 + C(z)H(z)}.$$

# FIR Approach

- Let

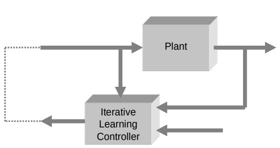
$$
\begin{aligned}
H(z) &= h_1 z^{-1} + h_2 z^{-2} + \cdots \,, \\
C(z) &= c_0 + c_1 z^{-1} + c_2 z^{-2} + \cdots \,,
\end{aligned}
$$

- For Case A the monotonic convergence condition can be shown to be:

$$
|h_1| \; > \; \sum_{i=2}^{N} |h_i - h_1 \sum_{j=1}^{i-1} h_j c_{i-1-j}|.
$$

- For Case B the monotonic convergence condition can be shown to be:

$$
|c_0 h_1| \; > \; \sum_{i=2}^{N} |\sum_{j=1}^{i} h_j c_{i-j} - h_1 \sum_{j=1}^{i-1} h_j c_{i-1-j}|.
$$



# FIR Approach (cont.)

- For both Case A and Case B a controller always exists to give a closed-loop system that satisfies the monotone convergence condition.

- For example, for Case A we can pick:

$$|h_i - h_1 \sum_{j=1}^{i-1} h_j c_{i-1-j}| = 0,$$

- That is, we solve recursively the following:

$$\begin{aligned} 0 &= |h_2 - h_1 c_0|, \\ 0 &= |h_3 - h_1(h_1 c_1 + h_2 c_0)|, \\ 0 &= |h_4 - h_1(h_1 c_2 + h_2 c_1 + h_3 c_0)|, \\ &\vdots \end{aligned}$$

- Then the system will have monotonic ILC convergence whenever ILC converges.

# FIR Approach (cont.)

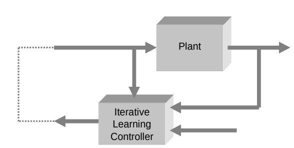
- Alternately (again for Case A), we can require:

$$|h_i - h_1 \sum_{j=1}^{i-1} h_j c_{i-1-j}| < \frac{|h_1|}{N-1},$$

- Or, equivalently, we solve the recursive equations:

$$\begin{aligned}
\frac{|h_1|}{N-1} &> |h_2 - h_1 c_0| \\
\frac{|h_1|}{N-1} &> |h_3 - h_1(h_1 c_1 + h_2 c_0)| \\
\frac{|h_1|}{N-1} &> |h_4 - h_1(h_1 c_2 + h_2 c_1 + h_3 c_0)| \\
&\vdots
\end{aligned}$$

- This approach will be more robust than the previous case.



# IIR Approach

- Now, suppose we let $C(z)$ be IIR:

$$
\begin{aligned}
H(z) &= \frac{n_h(z)}{d_h(z)} = \frac{b_1 z^{-1} + b_2 z^{-2} + \cdots + b_n z^{-n}}{a_0 + a_1 z^{-1} + a_2 z^{-2} + \cdots + a_n z^{-n}}, \\
C(z) &= \frac{n_c(z)}{d_c(z)} = \frac{\beta_0 + \beta_1 z^{-1} + \beta_2 z^{-2} + \cdots + \beta_n z^{-q}}{\alpha_0 + \alpha_1 z^{-1} + \alpha_2 z^{-2} + \cdots + \alpha_n z^{-q}}.
\end{aligned}
$$

- Now we have:

<u>Case A</u>:

$$
\begin{aligned}
H_{cl}^A &= \frac{\Gamma^A(z)}{\Delta(z)} = \frac{n_h(z)}{n_h(z)n_c(z) + d_h(z)d_c(z)}, \\
&= \frac{\gamma_1^A z^{-1} + \cdots + \gamma_{(q+n)}^A z^{-(q+n)}}{\delta_0 + \delta_1 z^{-1} + \cdots + \delta_{(q+n)} z^{-(q+n)}}, \\
&= h_1^{cl-A} z^{-1} + h_2^{cl-A} z^{-2} + \cdots .
\end{aligned}
$$

<u>Case B</u>: Similar.

# IIR Approach (cont.)

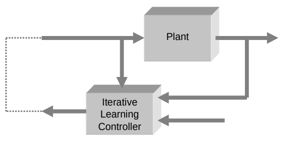
Define the following vectors:

$$
\begin{aligned}
\mathbf{a} &= (a_0, a_1, a_2, \cdots, a_n)^T, \\
\mathbf{b} &= (0, b_1, b_2, \cdots, b_n)^T, \\
\boldsymbol{\alpha} &= (\alpha_0, \alpha_1, \alpha_2, \cdots, \alpha_q)^T, \\
\boldsymbol{\beta} &= (\beta_0, \beta_1, \beta_2, \cdots, \beta_q)^T, \\
\boldsymbol{\gamma}^A &= (0, \gamma_1^A, \gamma_2^A, \cdots, \gamma_{(q+n)}^A)^T, \\
\boldsymbol{\gamma}^B &= (0, \gamma_1^B, \gamma_2^B, \cdots, \gamma_{(q+n)}^B)^T, \\
\boldsymbol{\delta} &= (\delta_0, \delta_1, \delta_2, \cdots, \delta_{(q+n)})^T.
\end{aligned}
$$

Let the appropriately-dimensioned matrices $A$ and $B$ be given as

$$
A = \begin{bmatrix} \mathbf{a} & 0 & 0 & \cdots & 0 \\ 0 & \mathbf{a} & 0 & \cdots & 0 \\ 0 & 0 & \mathbf{a} & \cdots & 0 \\ 0 & 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & 0 & \mathbf{a} \end{bmatrix}, \quad B = \begin{bmatrix} \mathbf{b} & 0 & 0 & \cdots & 0 \\ 0 & \mathbf{b} & 0 & \cdots & 0 \\ 0 & 0 & \mathbf{b} & \cdots & 0 \\ 0 & 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & 0 & \mathbf{b} \end{bmatrix}
$$





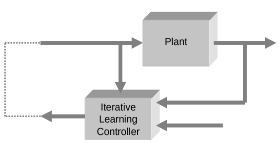
# IIR Approach

Let

$$H^{cl-A} = \begin{bmatrix} 0 & 0 & \cdots & 0 \\ h_1^{cl-A} & 0 & \cdots & \vdots \\ h_2^{cl-A} & h_1^{cl-A} & \ddots & \vdots \\ \vdots & \vdots & \ddots & 0 \\ h_{q+n+1}^{cl-B} & h_{q+n}^{cl-A} & \cdots & h_1^{cl-A} \\ \vdots & \vdots & \ddots & \vdots \\ h_N^{cl-A} & h_{N-1}^{cl-A} & \ddots & \vdots \\ \vdots & \vdots & \ddots & \vdots \end{bmatrix}.$$

A similar expression can be given for $H^{cl-B}$.





# IIR Approach (cont.)

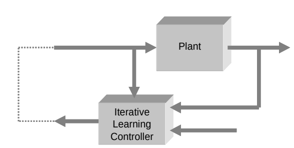
- Then we can derive

For Case A:

$$\begin{pmatrix} \mathbf{b} \\ 0 \\ \vdots \end{pmatrix} = H^{cl-A}[B|A] \begin{pmatrix} \boldsymbol{\beta} \\ \boldsymbol{\alpha} \end{pmatrix}.$$

For Case B:

$$\begin{pmatrix} \mathbf{B} & 0 \\ 0 & 0 \\ \vdots & \vdots \end{pmatrix} \begin{pmatrix} \boldsymbol{\beta} \\ \boldsymbol{\alpha} \end{pmatrix} = H^{cl-A}[B|A] \begin{pmatrix} \boldsymbol{\beta} \\ \boldsymbol{\alpha} \end{pmatrix}.$$



# IIR Approach (cont.)

- Hence, given
  - the plant, defined by the Sylvester matrix $[B|A]$ and
  - a desired closed-loop matrix of Markov parameters, $H^{cl-A}$ or $H^{cl-B}$,

  we can solve for the controller, defined by $\boldsymbol{\beta}$ and $\boldsymbol{\alpha}$.

- In general the solution of these equations is not known (they are over-determined).

- But, a solution can be possible for high enough controller order, as the null space of $[B|A]$ becomes non-trivial.

- In particular, by forcing the closed-loop system to be deadbeat a solution may be found.

# IIR Example

- Consider the second-order system:

$$Y_k(z) = \frac{z - 0.9}{z^2 + 0.2z - 0.125} U_k(z).$$

- Suppose we try a third-order controller for Case B, to give a deadbeat response.

# IIR Example (cont.)

Then

$$\boldsymbol{\delta} = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} = [A|B] \begin{pmatrix} \beta(0) \\ \beta(1) \\ \beta(2) \\ \beta(3) \\ \alpha(0) \\ \alpha(1) \\ \alpha(2) \\ \alpha(3) \end{pmatrix}$$

where the Sylvester matrix $[A|B]$ is given by

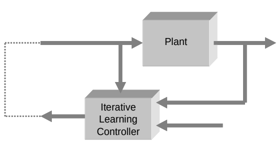
$$\begin{bmatrix} 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0.2 & 1 & 0 & 0 \\ -.9 & 1 & 0 & 0 & -0.0125 & 0.2 & 1 & 0 \\ 0 & -.9 & 1 & 0 & 0 & -0.0125 & 0.2 & 1 \\ 0 & 0 & -.9 & 1 & 0 & 0 & -0.0125 & 0.2 \\ 0 & 0 & 0 & -.9 & 0 & 0 & 0 & -0.0125 \end{bmatrix}$$

# IIR Example (cont.)

All solutions to this equation can be parameterized as

$$\begin{pmatrix}\beta(0)\\ \beta(1)\\ \beta(2)\\ \beta(3)\\ \alpha(0)\\ \alpha(1)\\ \alpha(2)\\ \alpha(3)\end{pmatrix} = \begin{pmatrix}-0.0866\\ -0.0169\\ -0.0017\\ 0.0001\\ 1.0\\ -0.1134\\ -0.0260\\ -0.0097\end{pmatrix} + w_1 \begin{pmatrix}0.4862\\ -0.1418\\ -0.0539\\ 0.003\\ 0\\ -0.4862\\ 0.6766\\ -0.2151\end{pmatrix} + w_2 \begin{pmatrix}0.3703\\ 0.6366\\ 0.1079\\ -0.007\\ 0\\ -0.3703\\ -0.2292\\ 0.5063\end{pmatrix}$$

- The first vector on the left hand side of the equation produces the deadbeat response.
- The second two vectors form a basis for the null space of the Sylvester equation.
- Thus, $w_1$ and $w_2$ parameterize all possible deadbeat responses for the closed-loop system for Case B.



# IIR Example (cont.)

Since the response is deadbeat, the numerator coefficients become:

$$
\begin{aligned}
\boldsymbol{\gamma}^B &= (\gamma_1 \gamma_2 \gamma_3 \gamma_4 \gamma_5) \\
&= (h_1^{cl-B} h_2^{cl-B} h_3^{cl-B} h_4^{cl-B} h_5^{cl-B}) \\
&= \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -.9 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -.9 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -.9 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} \beta(0) \\ \beta(1) \\ \beta(2) \\ \beta(3) \\ \alpha(0) \\ \alpha(1) \\ \alpha(2) \\ \alpha(3) \end{pmatrix}
\end{aligned}
$$

Thus

$$
\begin{pmatrix} h_1^{cl-B} \\ h_2^{cl-B} \\ h_3^{cl-B} \\ h_4^{cl-B} \\ h_5^{cl-B} \end{pmatrix} = \begin{pmatrix} -0.0866 \\ 0.0611 \\ 0.0135 \\ 0.0016 \\ -0.0001 \end{pmatrix} + w_1 \begin{pmatrix} 0.4862 \\ -0.5794 \\ 0.0737 \\ 0.0515 \\ -0.0027 \end{pmatrix} + w_2 \begin{pmatrix} 0.3707 \\ 0.3033 \\ -0.4650 \\ -0.1041 \\ 0.0063 \end{pmatrix}
$$

# IIR Example (cont.)

- If we pick $w_1 = w_2 = 1$, for example, the resulting closed-loop system seen by the ILC algorithm is

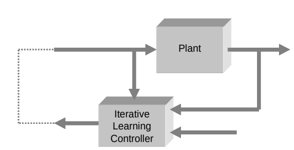
$$\begin{aligned} H_{cl}^{B} \;=\; & 0.7699z^{-1} - 0.2150z^{-2} - \\ & 0.3778z^{-3} - 0.0510z^{-4} + 0.0035^{-5}. \end{aligned}$$

  It is easily checked that this system satisfies the convergence conditions.

- Unfortunately, the method is not completely developed.

- Simply changing the zero from $z = -0.9$ to $z = -1.1$ results in an example in which it is not possible to meet the convergence conditions.

- More research is needed to understand this approach.



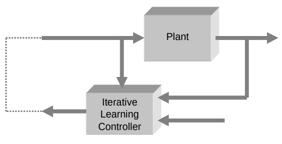
# Comments

- With classical Arimoto-type ILC algorithms, the equivalence of ILC convergence with monotonic ILC convergence depends on the characteristics of the plant.
- If a plant does not have the characteristics that ensure such monotonic convergence it is possible to "condition" the plant prior to the application of ILC using current cycle-feedback.
- Two such current-cycle feedback strategies were presented:
  - FIR design (results in high-order controller; always guaranteed, but possible robustness problems)
  - IIR design (solution not always guaranteed)
- Future work will focus on the IIR design approach.

# Outline

- Iterative Learning Control (ILC)
- Monotonic Convergence via Supervector Framework
- Current-Cycle Feedback Approach
- **Non-Causal Filtering ILC Design**
  - **Examples**
  - Optimal PD-type ILC Scheme: How to Design
  - Optimal PD-type ILC Scheme: Averaged Derivative
  - Remarks
- Time-Varying ILC Design
- LMI Approach to ILC Design





# Examples: PD-Type ILC

Simulation scenarios:

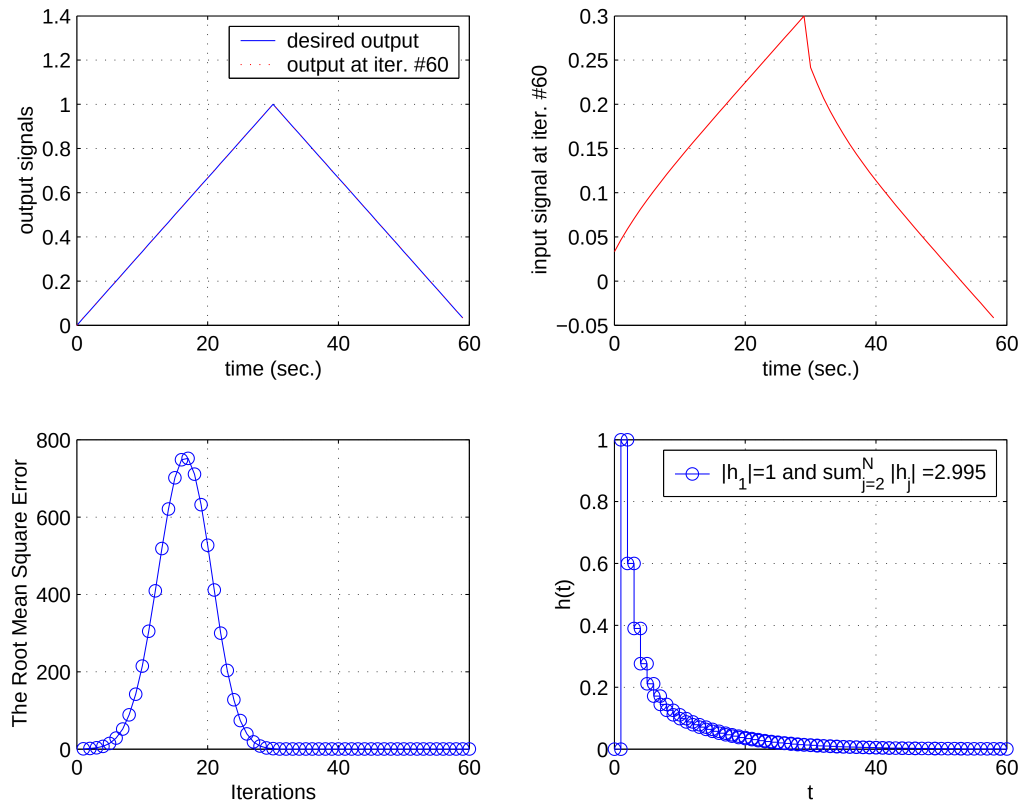
- Second order IIR models are used. All initial conditions are set to 0.
- All plants have $h_1 = 1$, so we fix $\gamma$=0.9 such that $|1 - \gamma h_1| < 1$.
- We fix $N$=60 and max number of iterations = 60.
- The desired trajectory is a triangle given by

$$y_d(t) = \begin{cases} 2t/N & , \ i = 1, \cdots, N/2 \\ 2(N-t)/N & , \ i = N/2+1, \cdots, N. \end{cases}$$

- We compare $u_{k+1}(t) = u_k(t) + \gamma e_k(t+1)$ with $u_{k+1}(t) = u_k(t) + \gamma(e_k(t+1) - \beta_1 e_k(t))$

Plant 1a. Stable lightly damped. $H_1(z) = \frac{z-0.8}{(z-0.5)(z-0.9)}$.

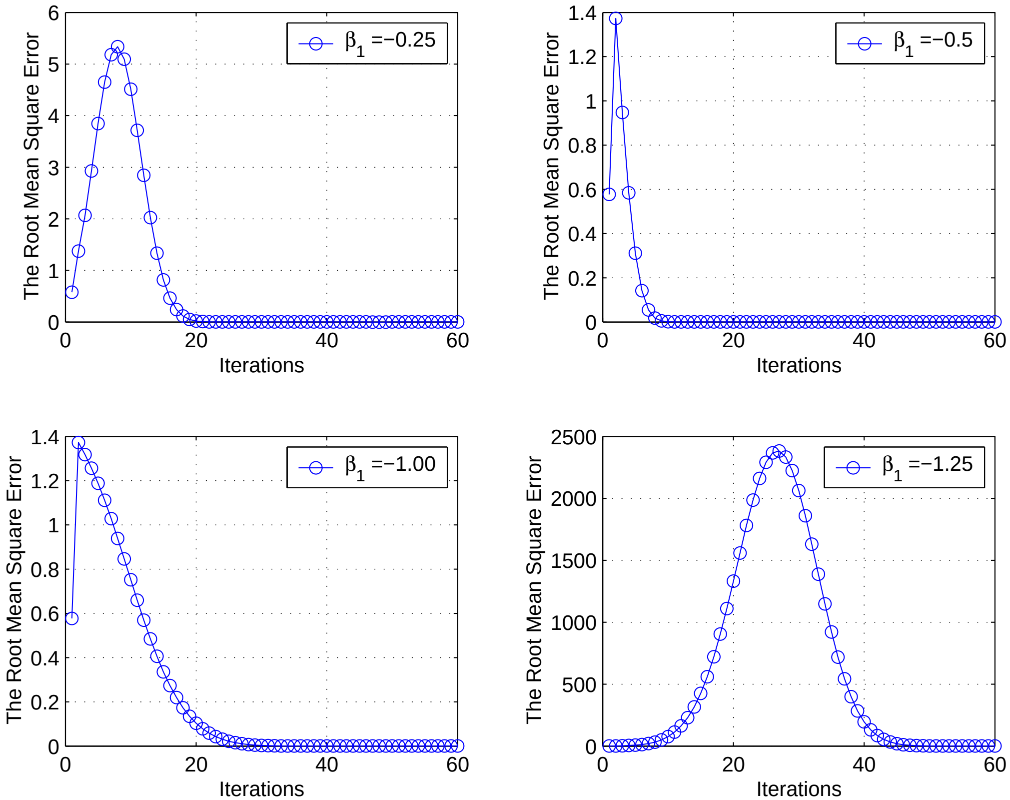
$$u_{k+1}(t) = u_k(t) + \gamma e_k(t+1),\ \gamma = 0.9$$

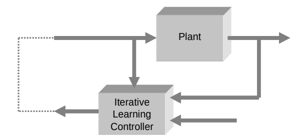


# *Part 2: Optimal Design of ILC Algorithms*

Plant 1b. Stable lightly damped. $H_1(z) = \frac{z-0.8}{(z-0.5)(z-0.9)}$.

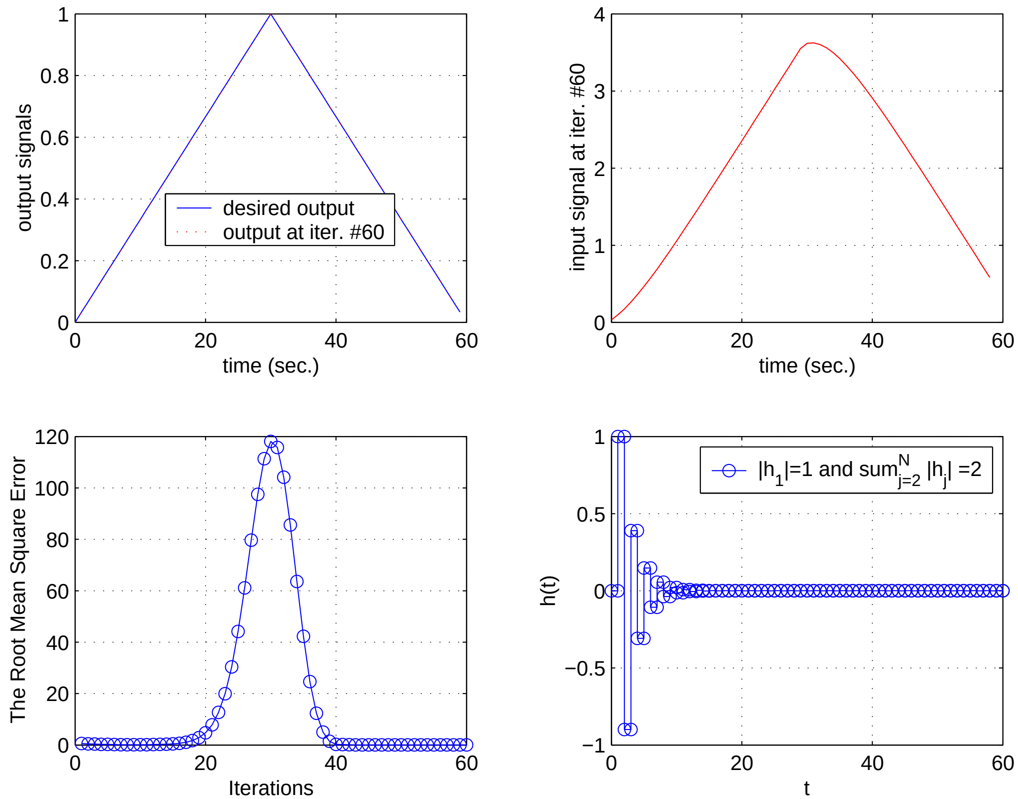
$$u_{k+1}(t) = u_k(t) + \gamma(e_k(t+1) - \beta_1 e_k(t)) \text{ with } \gamma = 0.9 \text{ fixed and } \beta_1 \text{ shown on the plots.}$$



Plant 2a. Stable oscillatory. $H_2(z) = \frac{z-0.8}{(z-0.5)(z+0.6)}$.

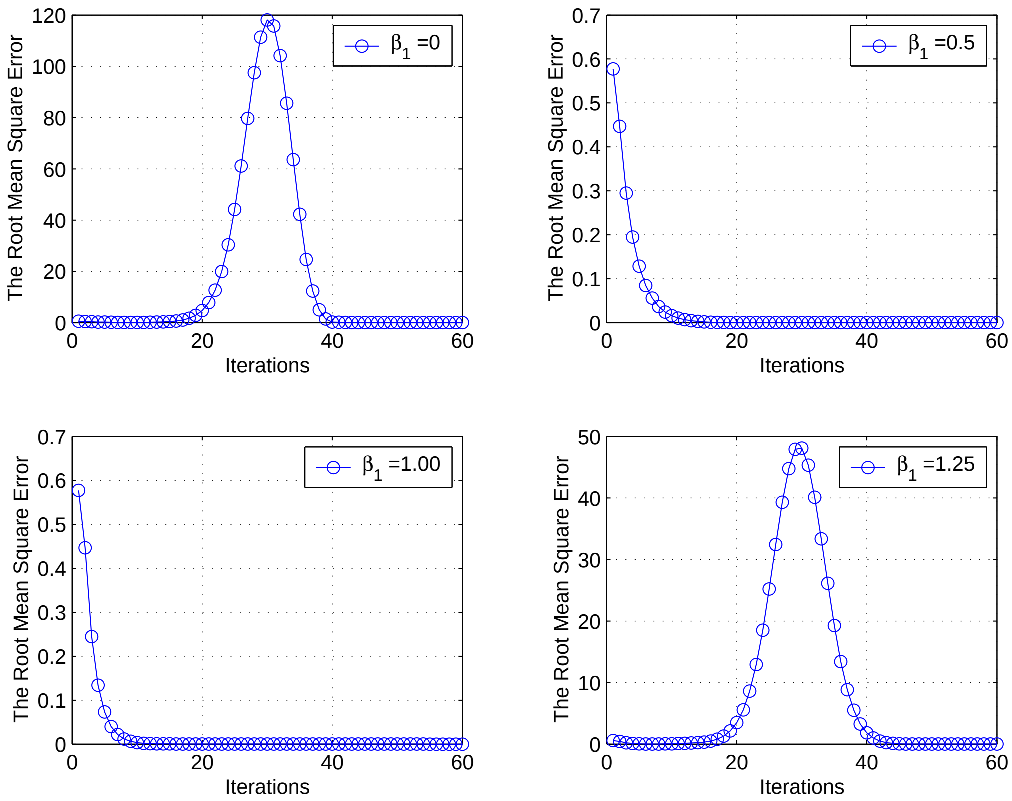
$$u_{k+1}(t) = u_k(t) + \gamma e_k(t+1),\ \gamma = 0.9$$



# Part 2: Optimal Design of ILC Algorithms

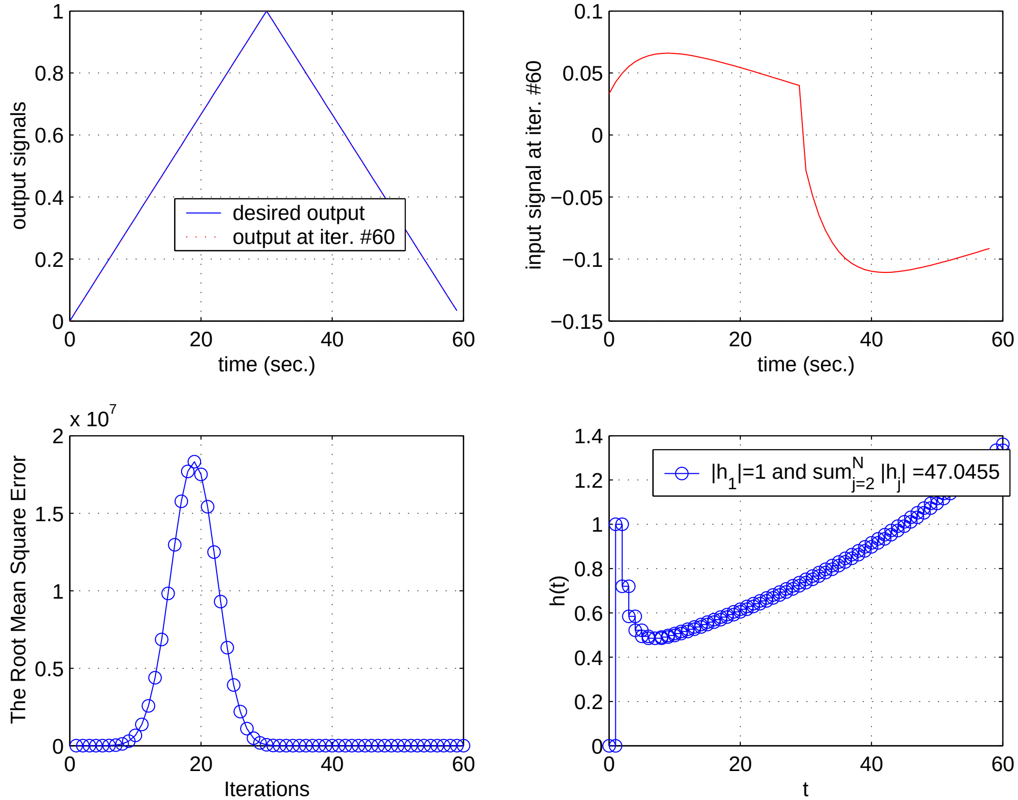
Plant 2b. Stable oscillatory. $H_2(z) = \frac{z-0.8}{(z-0.5)(z+0.6)}$.

$$u_{k+1}(t) = u_k(t) + \gamma(e_k(t+1) - \beta_1 e_k(t)) \text{ with } \gamma = 0.9 \text{ fixed and } \beta_1 \text{ shown on the plots.}$$



Plant 3a. Slightly unstable. $H_3(z) = \frac{z-0.8}{(z-0.5)(z-1.02)}$.

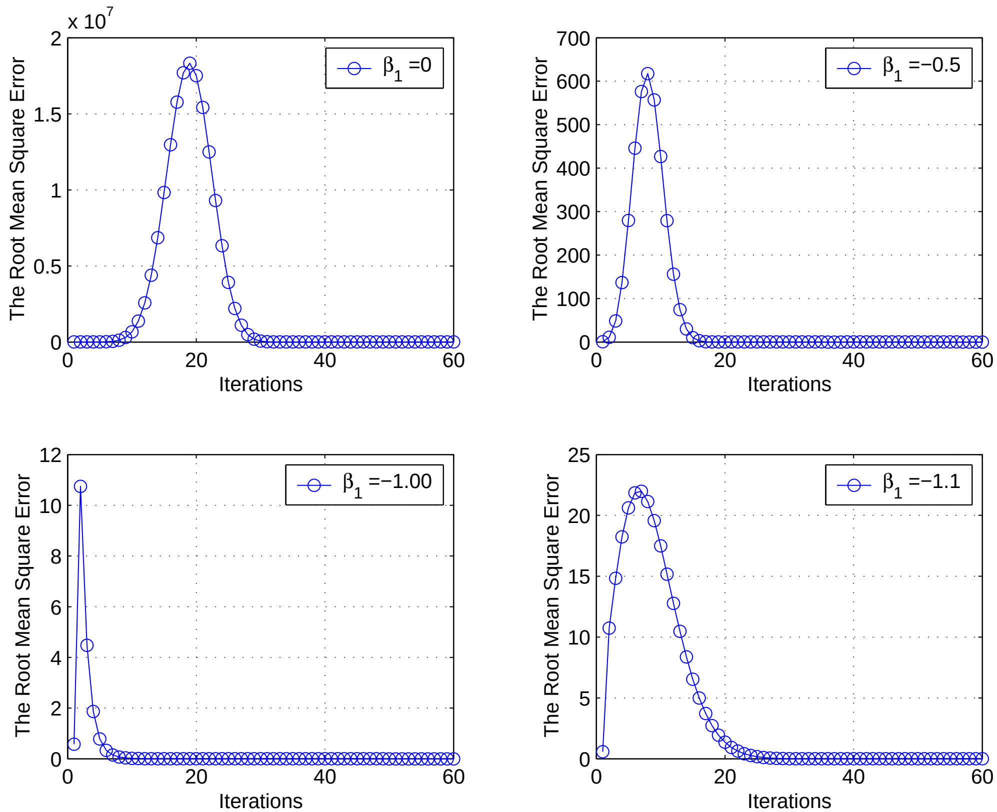
$$u_{k+1}(t) = u_k(t) + \gamma e_k(t+1),\ \gamma = 0.9$$



# Part 2: Optimal Design of ILC Algorithms

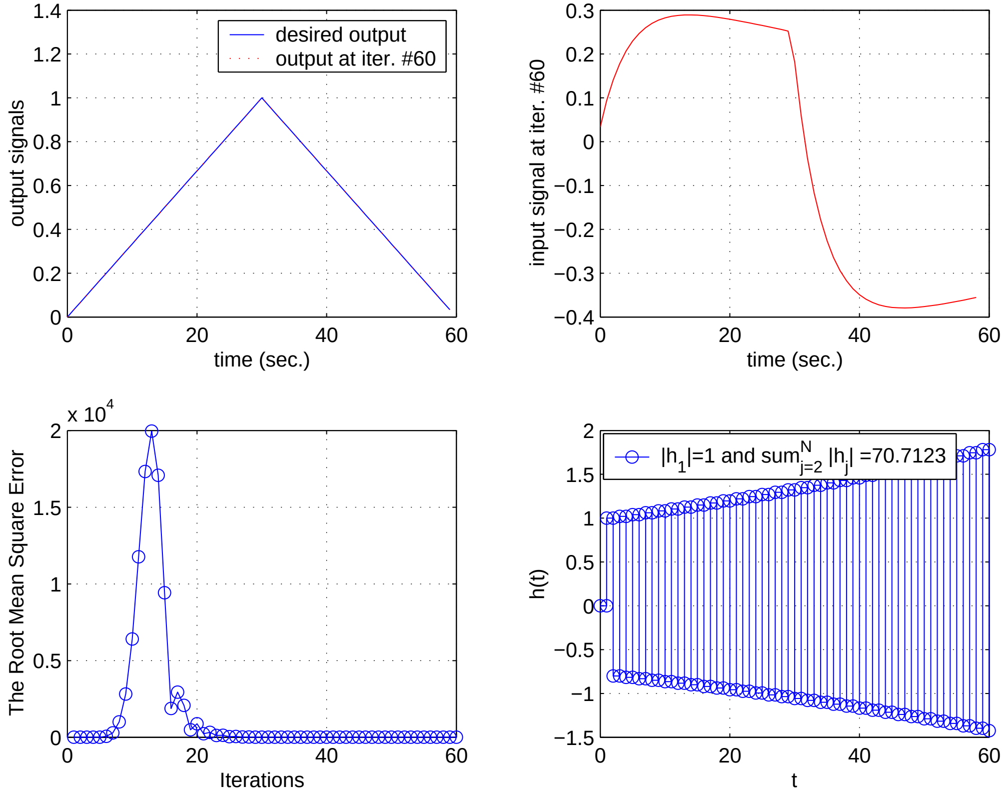
Plant 3b. Slightly unstable. $H_3(z) = \frac{z-0.8}{(z-0.5)(z-1.02)}$.

$$u_{k+1}(t) = u_k(t) + \gamma(e_k(t+1) - \beta_1 e_k(t)) \text{ with } \gamma = 0.9 \text{ fixed and } \beta_1 \text{ shown on the plots.}$$



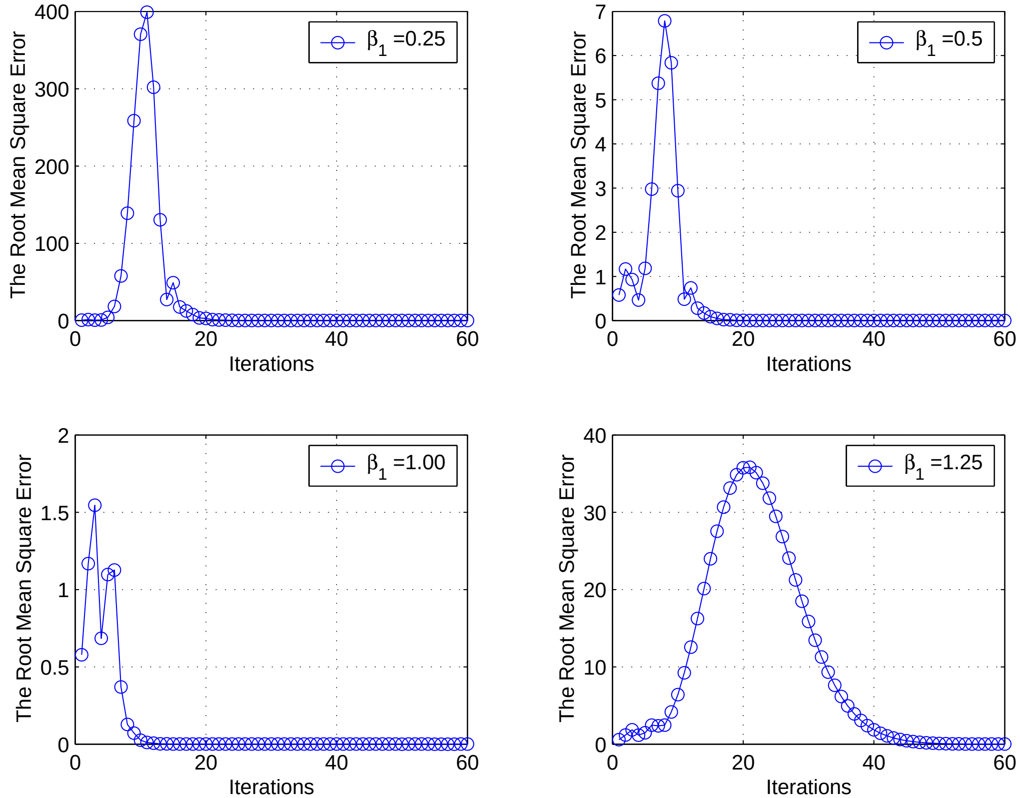
Plant 4a. Unstable oscillating. $H_4(z) = \frac{z-0.8}{(z+1.01)(z-1.01)}$.

$$u_{k+1}(t) = u_k(t) + \gamma e_k(t+1),\ \gamma = 0.9$$



Plant 4b. Unstable oscillating. $H_4(z) = \frac{z-0.8}{(z+1.01)(z-1.01)}$.

$$u_{k+1}(t) = u_k(t) + \gamma(e_k(t+1) - \beta_1 e_k(t)) \text{ with } \gamma = 0.9 \text{ fixed and } \beta_1 \text{ shown on the plots.}$$

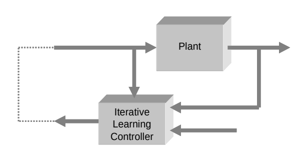


# Examples: PD-Type ILC (cont.)

- For $u_{k+1}(t) = u_k(t) + \gamma(e_k(t+1) - \beta_1 e_k(t))$ we conclude that:
  - Monotone convergence is possible for the right values of $\gamma$ and $\beta$.
  - Can relate "overshoot" in convergence for some values of $\beta$ to zeros in the iteration domain.
- In fact, further, can show:
  - Better convergence behavior is possible with $\beta < 0$.
  - How to pick the optimal $\beta$.
- In these simulations we used a simple structure. More generally, we can show how to pick a general lower triangular Toeplitz $L$ (i.e, design of $L(z)$) to find the optimal ILC filter for monotonic convergence.

# Outline

- Iterative Learning Control (ILC)
- Monotonic Convergence via Supervector Framework
- Current-Cycle Feedback Approach
- **Non-Causal Filtering ILC Design**
  - Examples
  - **Optimal PD-type ILC Scheme: How to Design**
  - Optimal PD-type ILC Scheme: Averaged Derivative
  - Remarks
- Time-Varying ILC Design
- LMI Approach to ILC Design



# Optimal PD-type ILC Scheme: How to Design - 1

- By using a one step backward finite difference as the approximation of the derivative (D) signal, the PD-type ILC is given by

$$u_{k+1}(t) = u_k(t) + k_p e_k(t) + k_d(e_k(t+1) - e_k(t)) \tag{1}$$

  where $k_p$ and $k_d$ are proportional and derivative learning gains respectively.

- Introduce the operator $T$ to map the column vector $h = [h_1, h_2, \cdots, h_N]'$ to a lower triangular Toeplitz matrix $H_p$, i.e., $H_p \triangleq T(h)$.

- For example, let $c_2 = [0, 1, 0, \cdots, 0]'$. Then, we have

$$T_2 \triangleq T(c_2) = \begin{bmatrix} 0 & 0 & 0 & \dots & 0 & 0 \\ 1 & 0 & 0 & \dots & 0 & 0 \\ 0 & 1 & 0 & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & 1 & 0 \end{bmatrix}. \tag{2}$$

# Optimal PD-type ILC Scheme: How to Design - 2

- In the sequel, we shall use a more general notion $T_i$, similar to the definition of $T_2$. Clearly, for $i = 1$, $T_i = I_N$.
- Using supervector representation, we can write

$$U_{k+1} = U_k(t) + k_p T_2 E_k + k_d (I_N - T_2) E_k \tag{3}$$

where $I_N = T_1$ is a square identity matrix of dimension $N$.

- Since $Y_k = H_p U_k$ and $E_k = Y_d - Y_k$, from (3) we have

$$E_{k+1} = H_e E_k = T(h_e) E_k \tag{4}$$

where

$$H_e = I_N - (k_p - k_d) H_p T_2 - k_d H_p \tag{5}$$

and

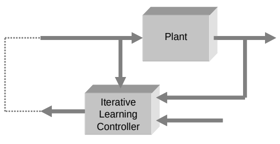
$$h_e = v_N - [\bar{h}_2, h - \bar{h}_2][k_p, k_d]'. \tag{6}$$

- In the above equation, we used the following notations:

$$v_i \triangleq [1, 0, \cdots, 0]' \in R^{i \times 1}$$

and

$$\bar{h}_2 \triangleq T_2 h = [0, h_1, h_2, \cdots, h_{N-1}]'.$$



# Optimal PD-type ILC Scheme: How to Design - 3

- The learning process is governed by (4) and the convergence condition is, analogous to

$$|h_1| > \sum_{j=2}^{N} |h_j|,$$

  that

$$\|H_e\|_i < 1. \tag{7}$$

- Clearly, if all eigenvalues of $H_e$, denoted by $\lambda(H_e) = [\lambda_1, \cdots, \lambda_N]'$, are absolutely less than one, the learning process will converge. However, $\max_i |\lambda_i| < 1$ does not imply (7). The consequence is that $\|E_k\|_i$ may not converge monotonically, which is widely recognized.

- In practice, we are more concerned with the monotonic convergence of the 1-norm, $\infty$-norm and 2-norm of $E_k$. The convergence conditions are corresponding to replacing '$i$' in (7) with '1', '$\infty$' or '2'.

# Optimal PD-type ILC Scheme: How to Design - 4

- Note that $H_e$ is a lower triangular Toeplitz matrix and

$$\|H_e\|_1 = \|H_e\|_\infty. \tag{8}$$

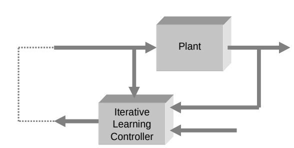
- Furthermore, $\|H_e\|_1 = \|T(h_e)\|_1 < 1$ if and only if $\|h_e\|_1 < 1$.

- So, the condition $\|h_e\|_1 < 1$ is a sufficient condition for monotonic convergence of the 1-norm, $\infty$-norm and 2-norm of $E_k$. The ILC design task becomes to optimizing $\|h_e\|_1 < 1$ with respect to $k_p$ and $k_d$.

- Thus we can define the following optimization problem for ILC design

$$J_{PD}^* = \min_{k_p,k_d} J_{PD} \stackrel{\triangle}{=} \min_{k_p,k_d} \|h_e\|_2^2.$$

Note that since $\|h_e\|_1 < \sqrt{N}\|h_e\|_2$, when $J_{PD}^*$ is small, it is possible to ensure that $\|h_e\|_1 < 1$.



# Optimal PD-type ILC Scheme: How to Design - 5

- Let $H = [\bar{h}_2, h - \bar{h}_2] \in R^{N\times 2}$ and $g = [k_p, k_d]'$. Then,

$$\begin{aligned} J_{PD} &= [v_N - Hg]'[v_N - Hg] \\ &= 1 - 2v_N' Hg + g'H'Hg. \end{aligned}$$

- Thus the optimal $g$ is simply

$$g^* = [k_p^*, k_d^*]' = (H'H)^{-1}H'v_N \tag{9}$$

and

$$J_{PD}^* = 1 - v_N' Hg^* = 1 - h_1 k_d^*. \tag{10}$$

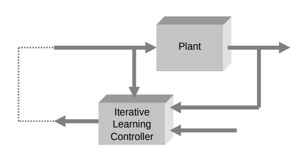
- Hence we get the following explicit design formulae:

$$k_p^* = -\frac{h_1 \bar{h}_2'(h - \bar{h}_2)}{\bar{h}_2'\bar{h}_2(h - \bar{h}_2)'(h - \bar{h}_2) - [\bar{h}_2'(h - \bar{h}_2)]^2}, \tag{11}$$

$$k_d^* = \frac{h_1 \bar{h}_2'\bar{h}_2}{\bar{h}_2'\bar{h}_2(h - \bar{h}_2)'(h - \bar{h}_2) - [\bar{h}_2'(h - \bar{h}_2)]^2} \tag{12}$$

and

$$J_{PD}^* = 1 - \frac{h_1^2 \bar{h}_2'\bar{h}_2}{\bar{h}_2'\bar{h}_2(h - \bar{h}_2)'(h - \bar{h}_2) - [\bar{h}_2'(h - \bar{h}_2)]^2}. \tag{13}$$



# Optimal PD-type ILC Scheme: How to Design - 6

**Simple Case-A**

- Set $k_d = 0$ in PD-type ILC

$$u_{k+1}(t) = u_k(t) + k_p e_k(t) + k_d(e_k(t+1) - e_k(t)) \tag{14}$$

- Then we get the *pure* P-type ILC: $u_{k+1}(t) = u_k(t) + k_p e_k(t)$.
- Using our optimal PD design formula, $J^*_{PD} = 1$.
- So, we cannot expect monotonic convergence of ILC since $J^*_{PD} = 1$. This in turn verifies that a correct time advance step, which corresponds to the system relative degree, is essential.

**Simple Case-B**:

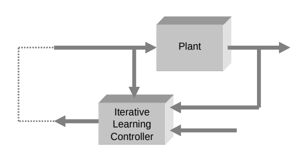
- Arimoto D-type ($k_p = k_d = \gamma$), for

$$u_{k+1}(t) = u_k(t) + \gamma e_k(t+1). \tag{15}$$

- Then using our optimal PD design formula, with $h_e = v_N - \gamma h$, gives

$$\gamma^* = h_1/(h'h), \quad J^*_P = J_P(\gamma^*) = 1 - h_1^2/(h'h). \tag{16}$$

- It is expected that for a given nominally measured $h$, $J^*_{PD} < J^*_P$.
- This means that the optimally designed PD-type ILC can be better than the optimally designed Arimoto D-type ILC in terms of monotonic convergence speed.

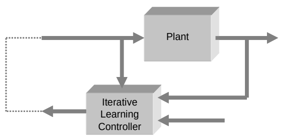


# Optimal PD-type ILC Scheme: How to Design - 7

Let's examine two simple extreme cases.

- **Extreme Case 1.** Let $h = [1, -1, 1, -1, \cdots, 1, -1]'$, i.e., the system is $z/(1+z)$ which is an extreme case for highly oscillatory systems.
  - When P-type ILC is considered, the optimal values from (16) are $\gamma^* = 1/N$ and $J_P^* = (N-1)/N$.
  - With a PD-type ILC (14), the optimal values via (11), (12) and (13) are $k_p^* = 2$, $k_d^* = 1$ and $J_{PD}^* = 0$.
  - Clearly, $J_{PD}^* < J_P^*$.

- **Extreme Case 2.** Let $h = [1, 1, 1, 1, \cdots, 1, 1]'$, i.e., the system is $z/(-1+z)$ which is an extreme case for very lightly damped systems.
  - For the P-type ILC, the optimal values are the same as in **Case 1.**
  - With a PD-type ILC (14), the optimal values are $k_p^* = 0$, $k_d^* = 1$ and $J_{PD}^* = 0$.
  - Again, $J_{PD}^* < J_P^*$.



# Outline

- Iterative Learning Control (ILC)
- Monotonic Convergence via Supervector Framework
- Current-Cycle Feedback Approach
- **Non-Causal Filtering ILC Design**
  - Examples
  - Optimal PD-type ILC Scheme: How to Design
  - **Optimal PD-type ILC Scheme: Averaged Derivative**
  - Remarks
- Time-Varying ILC Design
- LMI Approach to ILC Design

# Optimal PD-type ILC Scheme: Averaged Derivative - 1

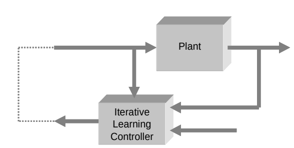
- For better noise suppression, it is a common practice to use a central difference formula.
- In this case, (14) becomes

$$u_{k+1}(t) = u_k(t) + k_p e_k(t) + k_d(e_k(t+1) - e_k(t-1))/2. \quad (17)$$

- The derivative estimate $(e_k(t+1) - e_k(t-1))/2$ can be regarded as an averaged value from two derivative estimates:
  - $e_k(t+1) - e_k(t)$
  - $e_k(t) - e_k(t-1)$
- For a more general averaged formula, we consider the following PD-type ILC scheme

$$u_{k+1}(t) = u_k(t) + k_p e_k(t) + \frac{k_d}{m}(e_k(t+1) - e_k(t-m+1)) \quad (18)$$

  where $m > 0$ is the number of averaging points.
- Clearly, (14) is a special case of (18) when $m = 1$. The value of $m$ depends on the noise suppression requirement. In practice, $m$ can be chosen between 1 to 4.



# Optimal PD-type ILC Scheme: Averaged Derivative - 2

- Starting from (4), using (18), we now have

$$H_e = I_N - k_p H_p T_2 - k_d H_p / m + k_d H_p T_m / m \tag{19}$$

and

$$h_e = v_N - [\bar{h}_2, (h - \hat{h}_m)/m][k_p, k_d]' \tag{20}$$

where $\hat{h}_m = [0_{1\times m}, h_1, h_2, \cdots, h_{N-m}]'$. Similarly, we can get

$$g^* = \begin{bmatrix} \bar{h}_2'\bar{h}_2 & \bar{h}_2'(h - \hat{h}_m)/m \\ \frac{\bar{h}_2'(h-\hat{h}_m)}{m} & \frac{(h-\hat{h}_m)'(h-\hat{h}_m)}{m^2} \end{bmatrix}^{-1} \begin{bmatrix} 0 \\ \frac{h_1}{m} \end{bmatrix}. \tag{21}$$

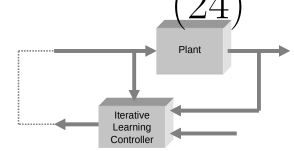
- The explicit design formulae using the averaged derivative:

$$k_p^* = -\frac{h_1 \bar{h}_2'(h - \hat{h}_m)}{\bar{h}_2'\bar{h}_2(h - \hat{h}_m)'(h - \hat{h}_m) - [\bar{h}_2'(h - \hat{h}_m)]^2}, \tag{22}$$

$$k_d^* = \frac{m h_1 \bar{h}_2'\bar{h}_2}{\bar{h}_2'\bar{h}_2(h - \hat{h}_m)'(h - \hat{h}_m) - [\bar{h}_2'(h - \hat{h}_m)]^2} \tag{23}$$

and from $J_{PD}^* = 1 - [0, h_1/m]g^*$,

$$J_{PD}^* = 1 - \frac{h_1^2 \bar{h}_2'\bar{h}_2}{\bar{h}_2'\bar{h}_2(h - \hat{h}_m)'(h - \hat{h}_m) - [\bar{h}_2'(h - \hat{h}_m)]^2}. \tag{24}$$



# Optimal PD-type ILC Scheme: Averaged Derivative - 3

- There is a trade-off between noise suppression and the rate of monotonic convergence of the ILC process. Consider $m = 2$:
  - **Extreme Case 1** : The optimal values via (22), (23) and (24) are $k_p^* = 1/(2N-3)$, $k_d^* = (2N-2)/(2N-3)$ and $J_{PD}^* = (N-2)/(2N-3)$.
  - **Extreme Case 2.** $k_p^* = -1/(2N-3)$; $k_d^*$ and $J_{PD}^*$ are the same as **Extreme Case 1**. Recall that $J_{PD}^*$ when $m = 1$ is 0.
- Clearly, the smoothing or averaging scheme for noise suppression is at the expense of slowing down the best achievable ILC monotonic convergence rate.
- This trade-off should be taken into account during ILC applications.

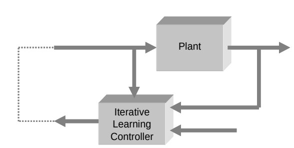
# Remarks

- We gave presented an optimal design procedure for the commonly used PD-type ILC updating law.
- Monotonic convergence in a suitable norm topology other than the exponentially weighted sup-norm is emphasized.
- For practical reasons, an averaged difference formula for the numerical derivative estimate is preferred over the conventional one-step backward difference method, as it helps in smoothing out the high frequency noise.
- Via analysis, we showed a trade-off between noise suppression and the rate of monotonic convergence of ILC process.

# Outline

- Iterative Learning Control (ILC)
- Monotonic Convergence via Supervector Framework
- Current-Cycle Feedback Approach
- Non-Causal Filtering ILC Design
- **Time-Varying ILC Design**
- LMI Approach to ILC Design





# Time-Varying ILC Gain

- Suppose we let

$$u_{k+1}(t) = u_k(t) + \lambda(t)e_k(t+1)$$

with

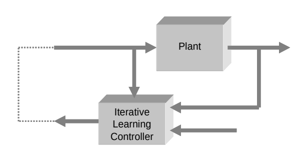
$$\lambda(t) = \gamma e^{-\alpha(t-1)}$$

- We can show that there always exists $\alpha$ and $\gamma$ so that $\|E_k\|_\infty$ and $\|E_k\|_2$ converge monotonically.

- The result also works with any general non-increasing function $\lambda(t)$.

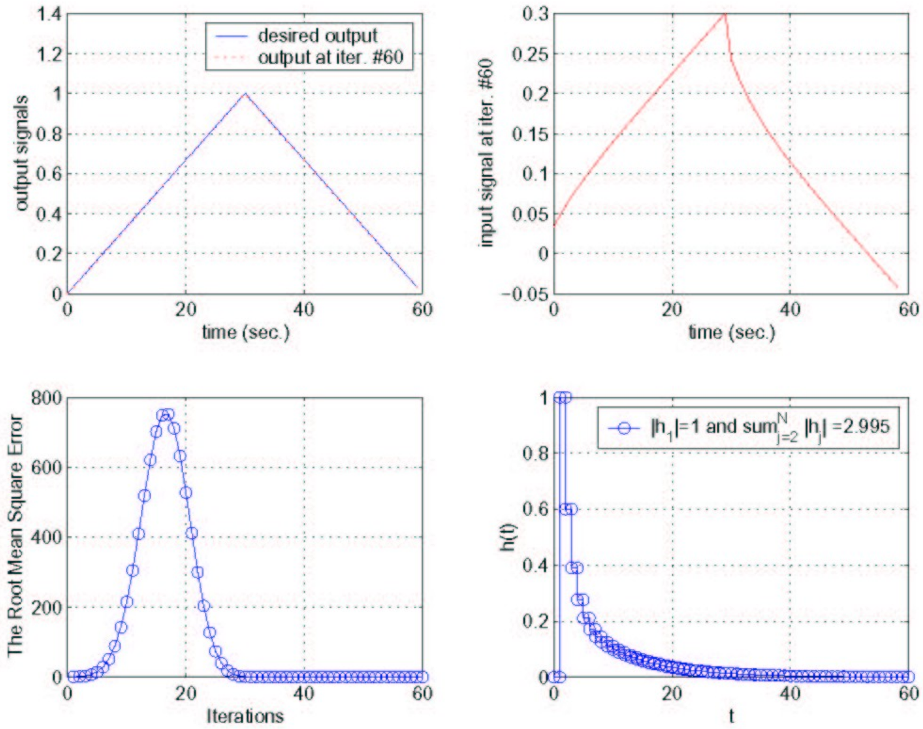
- Example: Consider the stable, lightly-damped plant

$$H_1(z) = \frac{z - 0.8}{(z-0.5)(z-0.9)}$$



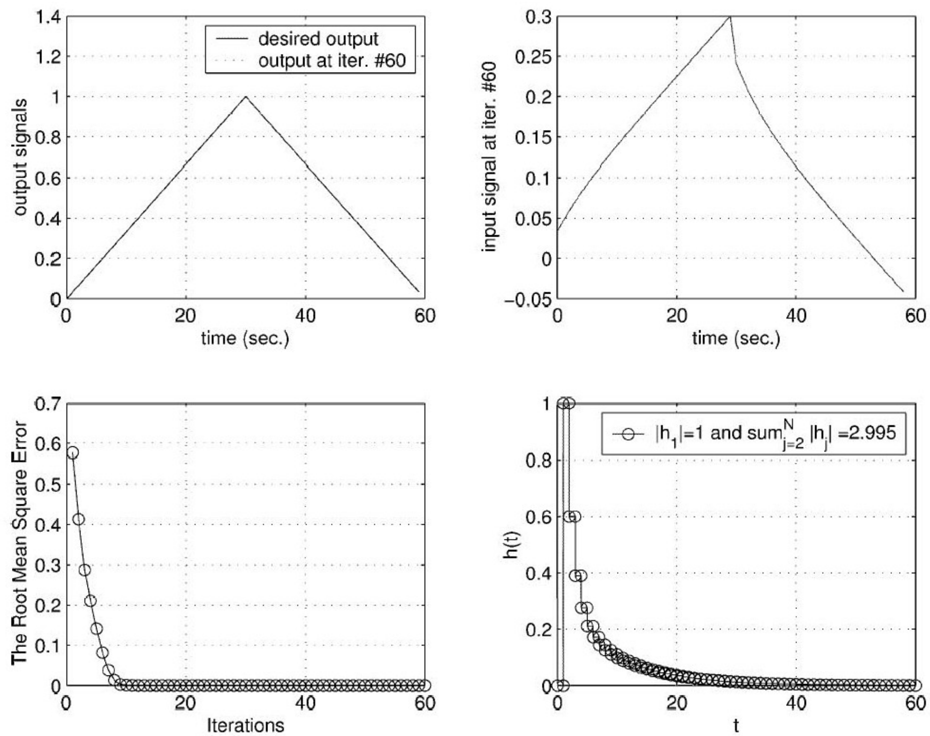
# Normal ILC

$$\gamma = 0.9, \alpha = 0$$



# ILC with a Time-Varying Gain

$$\gamma = 0.9, \alpha = 1.5/N$$



## Asymptotic Stability with a Time-Varying Learning Gain

- Using a time-varying learning gain $\lambda(t)$, the learning updating law becomes

$$u_{k+1}(t) = u_k(t) + \lambda(t)e_k(t+1).$$

- Let the varying learning gain $\lambda(t)$ be defined as follows:

$$\lambda(t) = \gamma e^{-\alpha(t-1)}$$

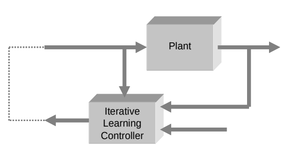
  where $\alpha$ is a suitably chosen positive real number.

- Define the $N \times N$ matrix $\Gamma$ by

$$\Gamma = \gamma \mathrm{diag}\{1, e^{-\alpha}, e^{-2\alpha}, \cdots, e^{-(N-1)\alpha}\}.$$

- **Theorem 1** *For the system $Y_k = HU_k$ and the learning control algorithm $U_{k+1} = U_k + \Gamma E_k$, the learning process converges iff*

$$\rho_1 \triangleq |1 - \gamma h_1| < 1.$$



# Recall: Monotonic Convergence Condition

- For the Arimoto-update ILC algorithm, the ILC scheme converges (monotonically) if the induced operator norm satisfies:

$$\|I - \gamma H_p\|_i < 1.$$

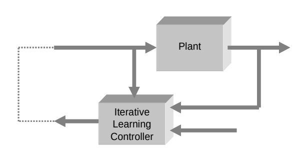
- Likewise, a NAS for convergence is:

$$|1 - \gamma h_1| < 1.$$

- Combining these, we can show that for a given gain $\gamma$, convergence implies monotonic convergence in the $\infty$-norm if

$$|h_1| > \sum_{j=2}^{N} |h_j|.$$

- Note this condition is independent of $\gamma$, but instead puts restrictions on the plant.



# Monotonic Convergence with a Time-Varying Learning Gain

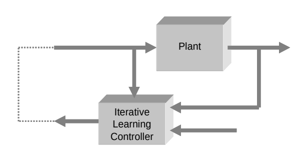
- As in the case of an Arimoto-type learning gain, the previous theorem cannot guarantee the monotonic convergence of the system with the time-varying learning gain.
- Here we will show there exists a choice of $\alpha$ such that the monotonic convergence is achievable.
- First, let $\bar{y}_k(t) = e^{-\alpha(t-1)}y_k(t)$, $\bar{y}_d(t) = e^{-\alpha(t-1)}y_d(t)$ and $\bar{e}_k(t) = e^{-\alpha(t-1)}e_k(t)$. The corresponding "supervectors" are denoted by $\bar{Y}_k = [\bar{y}_k(1), \bar{y}_k(2), \cdots, \bar{y}_k(N)]^T$ , $\bar{Y}_d = [\bar{y}_d(1), \bar{y}_d(2), \cdots, \bar{y}_d(N)]^T$, $\bar{E}_k = [\bar{e}_k(1), \bar{e}_k(2), \cdots, \bar{e}_k(N)]^T$.
- Then the transformed system can be written as

$$\bar{Y}_k = \bar{H}U_k,$$

where $\bar{H}$ is its matrix of Markov parameters given by

$$\bar{H} = \begin{bmatrix} h_1 & 0 & 0 & \dots & 0 \\ e^{-\alpha}h_2 & e^{-\alpha}h_1 & 0 & \dots & 0 \\ e^{-2\alpha}h_3 & e^{-2\alpha}h_2 & e^{-2a}h_1 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ e^{-(N-1)\alpha}h_N & e^{-(N-1)\alpha}h_{N-1} & e^{-(N-1)\alpha}h_{N-2} & \dots & e^{-(N-1)\alpha}h_1 \end{bmatrix}$$

and ILC update rule becomes $U_{k+1} = U_k + \Gamma E_k = U_k + \gamma\bar{E}_k$.



# Monotonic Convergence (cont.)

- Simple manipulations yield

$$\bar{E}_{k+1} = (1 - \gamma \bar{H})\bar{E}_k.$$

We can then derive the following theorem:

- **Theorem 2** *For the system $\bar{Y}_k = \bar{H}U_k$ and the learning control algorithm $U_{k+1} = U_k + \gamma \bar{E}_k$, there exist a $\gamma$ and an $\alpha > 0$ such that*

$$\sum_{j=2}^{N} e^{-(j-1)\alpha}|h_j| < |h_1|,$$

*and*

$$\gamma h_1 \in (0, 1).$$

*Thus, the monotonic convergence of $\|\bar{E}_k\|_\infty$ is guaranteed.*

# Monotonic Convergence (Cont.)

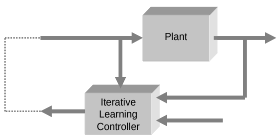
- **Remark** *Note that* $\bar{e}_k(t) = e^{-\alpha(t-1)}e_k(t)$*. From the fact that* $\max_{t\in[1,N]} |\bar{e}_{k+1}(t)| < \max_{t\in[1,N]} |\bar{e}_k(t)|$ *for all* $k$*, one cannot conclude that* $\max_{t\in[1,N]} e^{\alpha(t-1)}|\bar{e}_{k+1}(t)| < \max_{t\in[1,N]} e^{\alpha(t-1)}|\bar{e}_k(t)|$*. Therefore, the previous theorem does not guarantee the monotone convergence of* $\|E_k\|_\infty$*. Moreover, monotone convergence of* $\|\bar{E}_k\|_\infty$ *does not, in general, imply monotone convergence of* $\|\bar{E}_k\|_1$ *and* $\|\bar{E}_k\|_2$*.*

- However, we can show that there exists an $\alpha$ such that monotone convergence of $\|\bar{E}_k\|_1$ and $\|\bar{E}_k\|_2$ can be ensured.

- First, however, we need the following intermediate result.

**Theorem 3** *There exists an* $\alpha$ *such that for all* $k$ *and* $t$

$$|\bar{e}_{k+1}(t)| \leq |\bar{e}_k(t)|.$$



# Monotonic Convergence (cont.)

- With Theorem 3, we can immediately conclude that there exists an $\alpha$ such that the convergence of $\|\bar{E}_k\|_1$ and $\|\bar{E}_k\|_2$ can be ensured to be monotonic, i.e.,

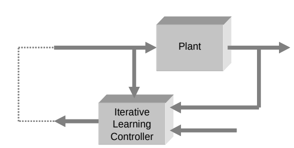
$$\sum_{t=1}^{N} |\bar{e}_{k+1}(t)| - \sum_{t=1}^{N} |\bar{e}_k(t)| \leq 0,$$

$$\sum_{t=1}^{N} |\bar{e}_{k+1}(t)|^2 \leq \sum_{t=1}^{N} |\bar{e}_k(t)|^2 \text{ and } \sqrt{\sum_{t=1}^{N} |\bar{e}_{k+1}(t)|^2} - \sqrt{\sum_{t=1}^{N} |\bar{e}_k(t)|^2} \leq 0.$$

- Finally, from the monotonicity of $\|\bar{E}_k\|_1$ and $\|\bar{E}_k\|_2$ we can conclude the monotonicity of $\|E_k\|_1$ and $\|E_k\|_2$:

$$\begin{aligned}
\|E_{k+1}\|_1 - \|E_k\|_1 &= \sum_{t=1}^{N} e^{\alpha(t-1)} |\bar{e}_{k+1}(t)| - \sum_{t=1}^{N} e^{\alpha(t-1)} |\bar{e}_k(t)| \\
&= \sum_{t=1}^{N} e^{\alpha(t-1)} \left(|\bar{e}_{k+1}(t)| - |\bar{e}_k(t)|\right) \leq 0, \\
\|E_{k+1}\|_2^2 - \|E_k\|_2^2 &= \sum_{t=1}^{N} e^{2\alpha(t-1)} \left(|\bar{e}_{k+1}(t)|^2 - |\bar{e}_k(t)|^2\right) \leq 0.
\end{aligned}$$

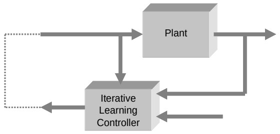




# Outline

- Iterative Learning Control (ILC)
- Monotonic Convergence via Supervector Framework
- Current-Cycle Feedback Approach
- Non-Causal Filtering ILC Design
- Time-Varying ILC Design
- **LMI Approach to ILC Design**





# LMI Approach to ILC Design

- Consider again the SISO discrete-time system $Y_k(z) = H(z)U_k(z)$ with transfer function

$$H(z) = h_1 z^{-1} + h_2 z^{-2} + \cdots$$

- For trial length $N$ and desired output $y_d(t)$, lift the time-domain signals to form the super-vectors:

$$\begin{aligned} U_k &= (u_k(0), u_k(1), \cdots, u_k(N-1)) \\ Y_k &= (y_k(1), y_k(2), \cdots, y_k(N)) \\ Y_d &= (y_d(1), y_d(2), \cdots, y_d(N)) \end{aligned}$$

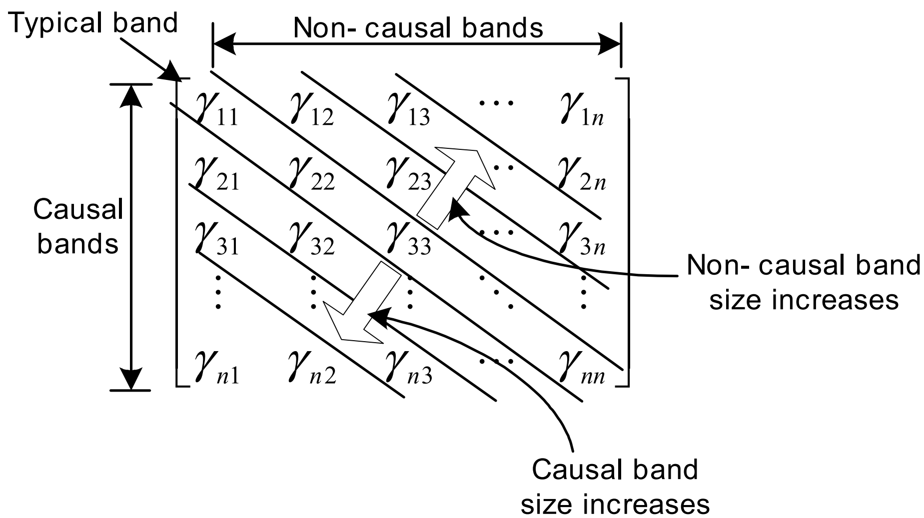
- Then write $Y_k = HU_k$, where $H$ is given by:

$$H = \begin{bmatrix} h_1 & 0 & \cdots & 0 \\ h_2 & h_1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ h_N & h_{N-1} & \cdots & h_1 \end{bmatrix}$$

- Also, let the ILC update law be given as $u_{k+1}(t) = u_k(t) + L(z)(y_d(t+1) - y_k(t+1))$, which can also be written as $U_{k+1} = U_k + \Gamma E_k$, where $\Gamma$ could be upper- or lower-triangular (Toeplitz or not), band-diagonal, or fully-populated, depending on the algorithm.

# LMI Approach to ILC Design (cont.)

- Define "bands" in $\Gamma$ as follows:




- In this section we use LMI techniques to design $\Gamma$ for different band sizes and structure in $\Gamma$.
- Recall, the LMI techniques solves the problem of minimizing or maximizing a convex objective function $J(x)$ subject to the constraint

$$F(x) \equiv F_0 + \sum_{i=1}^{m} x_i F_i \geq 0,$$

where $x \in \Re^m$ is the decision variable, $F_i = F_i^T, i = 1, \cdots, m$, are given symmetric matrices, and the constraint $\geq 0$ means positive semidefinite (i.e., nonnegative eigenvalues).

## Definitions

- $\Gamma$ is a *linear time-invariant* (LTI) ILC gain matrix if all the learning gain components in each diagonal are fixed as the same value.

- $\Gamma$ is a *linear time-varying* (LTV) ILC gain matrix if the learning gain components in each diagonal are different from each other.

- The system is *asymptotically stable* if every finite initial state excites a bounded response, and the error ultimately approaches 0 as $k \to \infty$.

- The system is *monotonically convergent* if $\|e_{k+1}\| < \|e_k\|$, and ultimately approaches 0 as $k \to \infty$.

# Basic Results

- When Arimoto or causal-only gains are used, the asymptotic stability condition is defined as:

$$|1 - \gamma_{ii} h_1| < 1, i = 1, \cdots, n$$

- When non-causal gains are used in the ILC learning gain matrix the asymptotic stability condition becomes:

$$\rho(I - H\Gamma) < 1$$

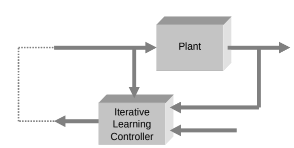
  where $\rho$ represents the spectral radius of $(I - H\Gamma)$.

- The condition for monotonic convergence is the same for all types of gain and requires:

$$\|I - H\Gamma\|_i < 1$$

  where $\|\cdot\|_i$ represents the induced operator norm in the topology of interest.

- In this section we will consider the standard $l_1$ and $l_\infty$ norm topologies.



# Basic Results (cont.)

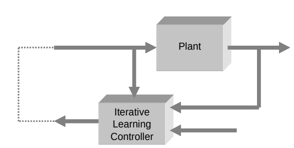
Consider four different cases:

1. Arimoto gains with causal LTI gains.
2. Arimoto gains with causal LTV gains.
3. Arimoto gains with both causal and non-causal LTI gains.
4. Arimoto gains with causal and non-causal LTV gains.

- *Lemma 1*: In Case 1, the minimum of $\| I - H\Gamma\|_1$ and $\| I - H\Gamma\|_\infty$ occurs if and only if $\Gamma$ is exactly equal to the inverse of $H$.
- *Lemma 2*: In Case 2, Case 3, and Case 4, the minimum of $\| I - H\Gamma\|_1$ and the minimum of $\| I - H\Gamma\|_\infty$ are zero if and only if $\Gamma$ is exactly equal to the inverse of $H$.
- Thus, we conclude that the best structure of $\Gamma$ is the inverse of $H$. This is a necessary and sufficient condition.
- However, it is unrealistic to assume that we know $H$ exactly and it is not advisable to use the inverse of $H$ as it can be ill-conditioned.
- Therefore, we seek to optimize $\Gamma$ when it has a fixed structure.

# More Definitions and Basic Results

- An LTI learning gain matrix with fixed band size is denoted as $\Gamma_{LTI}$, and an LTV learning gain matrix with the same band size as $\Gamma_{LTI}$ is denoted as $\Gamma_{LTV}$.
- When $\Gamma$ is fixed as $\Gamma_{LTI}$, the minimum of $\| I - H\Gamma_{LTI} \|$ is denoted by $J_I^*$; and when $\Gamma$ is fixed as $\Gamma_{LTV}$, the minimum of $\| I - H\Gamma_{LTV} \|$ is denoted by $J_V^*$.
- *Theorem*: If the same band size ILC gain matrices are used in $\Gamma_{LTI}$ and $\Gamma_{LTV}$, the following inequality is satisfied:
$$J_V^* \leq J_I^*$$
- *Corollary*: If the same band size is used in causal ILC and non-causal/causal ILC, then
$$J_N^* \leq J_C^*,$$
where $J_N^*$ is the minimum value using causal, Arimoto, and non-causal learning gains; and $J_C^*$ is the minimum value using only causal and Arimoto gains.
- In summary, we conclude that
  - The best gain matrix is just the inverse of $H$ with respect to convergence in the $l_1$ and $l_\infty$ norms.
  - When the band size is fixed, LTV is better than LTI
  - Including non-causal terms is more optimal than using Arimoto- or causal-only terms.



# LMI Design Technique

- We wish to satisfy the monotonic convergence condition $\min[\overline{\sigma}(I-H\Gamma)] < 1$ (i.e., we wish to minimize the maximum (indicated by the overbar notation) singular value of the map $(I - H\Gamma)$).

- Now, because

$$\sigma[I - H\Gamma] \equiv \lambda([I - H\Gamma][I - H\Gamma]^T)$$

(where $\sigma$ denotes singular value and $\lambda$ denotes eigenvalue) and because:

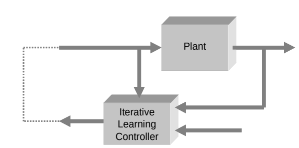
$$\lambda([I - H\Gamma][I - H\Gamma]^T) \leq \| [I - H\Gamma][I - H\Gamma]^T \|$$

then by minimizing $\| [I - H\Gamma][I - H\Gamma]^T \|$, we can limit the upper bound of $\overline{\sigma}(I - H\Gamma)$.

- Thus, because

$$\min(\| [I - H\Gamma][I - H\Gamma]^T \|)$$

is a typical matrix inequality problem, the ILC design problem can be solved by an LMI.



# LMI Design for General $\Gamma$

- By minimizing $\|[I - H\Gamma][I - H\Gamma]^T\|$, we can limit the upper bound of $\overline{\sigma}(I - H\Gamma)$.
- The optimization problem, $\min(\|[I - H\Gamma][I - H\Gamma]^T\|)$, can be changed to an matrix inequality problem given by:

$$\min\{x_1^2\}$$

subject to

$$x_1^2 I > [I - H\Gamma][I - H\Gamma]^T.$$

- Then, to express the learning gain matrix $\Gamma$ in a linear form, we convert this to the following inequality:

$$\begin{bmatrix} x_1 I & [I - H\Gamma] \\ [I - H\Gamma]^T & x_1 I_{N\times N} \end{bmatrix} > 0_{2N\times 2N}$$

leading to the following:

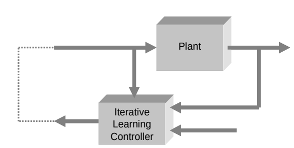
- **Suggestion** *Design a general* $\Gamma$ *by solving the LMI*

$$\max\{-x_1^2\}$$

*subject to*

$$-x_1^2 I_{2N\times 2N} - \begin{bmatrix} 0_- & I \\ I & 0_- \end{bmatrix} + \begin{bmatrix} H \\ 0_- \end{bmatrix} \Gamma \begin{bmatrix} 0_- & I \end{bmatrix} + \begin{bmatrix} 0_- \\ I \end{bmatrix} \Gamma^T \begin{bmatrix} H^T & 0_- \end{bmatrix} < 0_{2N\times 2N}$$

*where* $0_-$ *is* $N \times N$ *zero matrix.*



# LMI Design for Fixed Band-Size LTI $\Gamma$

- Consider a structure-fixed learning gain matrix such as:

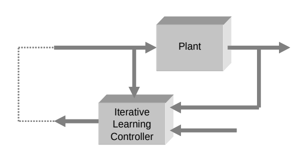
$$\Gamma = \begin{bmatrix} \gamma_p & \gamma_N^1 & \gamma_N^2 & \cdots & \gamma_N^{N-1} \\ \gamma_C^1 & \gamma_p & \gamma_N^1 & \cdots & \gamma_N^{N-2} \\ \gamma_C^2 & \gamma_C^1 & \gamma_p & \cdots & \gamma_N^{N-3} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \gamma_C^{N-1} & \gamma_C^{N-2} & \gamma_C^{N-3} & \cdots & \gamma_p \end{bmatrix},$$

where subscript $N$ denotes the noncausal gains, $C$ denotes the causal gains, and the diagonal terms are fixed at a same value, (e.g., Toeplitz gain matrix denoting LTI learning algorithm).).

- The algorithm for this case is described by:

Table 1: Markov matrices for LTI ILC

**for** $j = 1 : 1 : N - 1$
$\quad H_C^j(:, 1 : N - j) = H(:, j + 1 : N)$
$\quad H_C^j(:, N - j + 1 : N) = 0_-$
$\quad H_N^j(:, j + 1 : N) = H(j, 1 : N - j)$
$\quad H_N^j(:, 1 : j) = 0_-$
**end**



# LMI Design for Fixed Band-Size LTI $\Gamma$ (cont.)

- **Suggestion**: *For a fixed band-size, LTI update law, the following LMI can be used to find* $\Gamma$*:*

$$\max\{-x_1^2\}$$

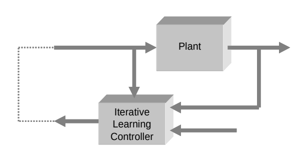
*subject to*

$$-x_1^2 I_{2N\times 2N} - \begin{bmatrix} 0_- & I \\ I & 0_- \end{bmatrix} + M_1 + M_2 + M_3 < 0_{2N\times 2N}, \tag{25}$$

*with*

$$\begin{aligned}
M_1 &= \begin{bmatrix} 0_- & H_p \\ 0_- & 0_- \end{bmatrix} \gamma_p + \gamma_p \begin{bmatrix} 0_- & 0_- \\ H_p^T & 0_{N\times N} \end{bmatrix}, \\
M_2 &= \begin{bmatrix} 0_- & H_C^1 \\ 0_- & 0_- \end{bmatrix} \gamma_C^1 + \gamma_C^1 \begin{bmatrix} 0_- & 0_- \\ (H_C^1)^T & 0_- \end{bmatrix} + \cdots + \begin{bmatrix} 0_- & H_C^{N-1} \\ 0_- & 0_- \end{bmatrix} \gamma_C^{N-1} + \gamma_C^{N-1} \begin{bmatrix} 0_- & 0_- \\ (H_C^{N-1})^T & 0_- \end{bmatrix}, \\
M_3 &= \begin{bmatrix} 0_- & H_N^1 \\ 0_- & 0_- \end{bmatrix} \gamma_N^1 + \gamma_N^1 \begin{bmatrix} 0_- & 0_- \\ (H_N^1)^T & 0_- \end{bmatrix} + \cdots + \begin{bmatrix} 0_- & H_N^{N-1} \\ 0_- & 0_- \end{bmatrix} \gamma_N^{N-1} + \gamma_N^{N-1} \begin{bmatrix} 0_- & 0_- \\ (H_N^{N-1})^T & 0_- \end{bmatrix},
\end{aligned}$$

*where* $H_p = H$*, and* $H_C^i$ *and* $H_N^i$ *are calculated from the algorithms in Table 1.*



# LMI Design for Fixed Band-Size LTI $\Gamma$ (cont.)

- Proof:
  - Expand $I - H\Gamma$ as

$$I - [\gamma_C^{N-1} H_C^{N-1} + \cdots + \gamma_C^1 H_C^1 + \gamma_p H_p + \gamma_N^1 H_N^1 + \cdots + \gamma_N^{N-1} H_N^{N-1}], \tag{26}$$

    where $H_C^k, k = 1, \cdots, N-1$ are Markov matrices corresponding to causal gains; $H_p$ is a Markov matrix corresponding to Arimoto-like gains; and $H_N^k, k = 1, \cdots, N-1$ are Markov matrices corresponding to non-causal gains.
  - These Markov matrices can be calculated by expanding $I - H\Gamma$ as shown in Table 1.
  - The matrix inequality problem is then changed to the optimization problem:

$$\min\{x_1^2\}$$

    subject to

$$\begin{bmatrix} x_1 I & [I - H\Gamma] \\ [I - H\Gamma]^T & x_1 I_{N\times N} \end{bmatrix} > 0_{2N\times 2N}. \tag{27}$$

  - By inserting (26) into (27), we have (25).
  - Therefore, since each learning gains are expressed in a linear form, LMI optimization can be used.

# LMI Design for Fixed Band-Size LTV $\Gamma$

- Now consider the LTV case. The following learning gain matrix is used, assuming a fixed band size:

$$\Gamma = [\gamma_{ij}]$$

- **Suggestion** *The optimization problem is designed as*

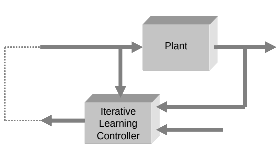
$$\max\{-x_1^2\}$$

*subject to*

$$-x_1^2 I_{2N\times 2N} - \begin{bmatrix} 0_- & I \\ I0_- & \end{bmatrix} + \sum_{j=1}^{N}\sum_{i=1}^{N}[H_u\gamma_{ij} + \gamma_{ij}H_l] < 0_{2N\times 2N},$$

*where*

$$H_u = \begin{bmatrix} 0_- & H_{ij} \\ 0_- & 0_- \end{bmatrix};\ H_l = \begin{bmatrix} 0_- & 0_- \\ H_{ij}^T & 0_- \end{bmatrix}.$$



# LMI Design for Fixed Band-Size LTV $\Gamma$ (cont.)

- $H_{ij}$ are Markov matrices corresponding to $\gamma_{ij}$, which is calculated by expanding $I - H\Gamma$ as:

$$\begin{aligned} I \;-\; & [H_{11}\gamma_{11} + \cdots + H_{1N}\gamma_{1N} \\ & \qquad\qquad \vdots \\ & H_{N1}\gamma_{N1} + \cdots + H_{NN}\gamma_{NN}], \end{aligned}$$

  where $H_{kl}$ is a matrix composed of one column vector beginning from $k^{th}$ row and $l^{th}$ column such as:

$$H_{kl} = \begin{bmatrix} {}^{11}0 & \dots & {}^{1l}0 & \dots & {}^{1N}0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ {}^{k1}0 & \dots & {}^{kl}h_1 & \dots & {}^{kN}0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ {}^{N1}0 & \dots & {}^{Nl}h_{N-k} & \dots & {}^{NN}0 \end{bmatrix},$$

  where left superscript represent $k^{th}$ row and $l^{th}$ element of matrix $H_{kl}$; ${}^{ij}0$ means zero at $i^{th}$ row and $j^{th}$ column; and $h_i$ are Markov parameters.

- When the band size is fixed as $m$, the algorithms in Table 2 and Table 3 are used, where $\Sigma_1$ $\Sigma_2$ are summed to make LMI constraints given by

$$-x_1^2 I_{2N\times 2N} - \begin{bmatrix} 0_- & I \\ I & 0_- \end{bmatrix} + \sum\nolimits_1 + \sum\nolimits_2 < 0_{2N\times 2N}.$$

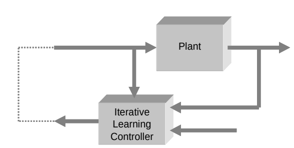
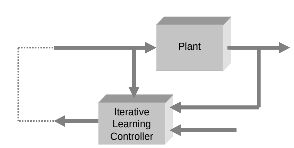


Table 2: : Markov matrices for LTV ILC

**for** $i = 1 : 1 : m$
  **for** $j = 1 : 1 : i$
    **for** $k = 1 : 1 : N - j + 1$
      $l = k + j - 1$
      $\gamma' = \gamma_{kl}$
      $R(1 : N, 1 : N) = 0_{-}$
      $R(1 : N, l) = H(1 : N, k)$
      $\Sigma_1 = \Sigma_1 + \begin{bmatrix} 0_{-} & R \\ 0_{-} & 0_{-} \end{bmatrix} \gamma' + \gamma' \begin{bmatrix} 0_{-} & 0_{-} \\ R^T & 0_{-} \end{bmatrix}$
    **end**
  **end**
**end**

Table 3: Markov matrices for LTI ILC (cont.)

**for** $i = 1 : 1 : m$
  **for** $j = 1 : 1 : i - 1$
    **for** $k = j + 1 : 1 : N$
      $l = k - j$
      $\gamma' = \gamma_{kl}$
      $R(1 : N, 1 : N) = 0_{-}$
      $R(1 : N, l) = H(1 : N, k)$
      $\Sigma_2 = \Sigma_2 + \begin{bmatrix} 0_{-} & R \\ 0_{-} & 0_{-} \end{bmatrix} \gamma' + \gamma' \begin{bmatrix} 0_{-} & 0_{-} \\ R^T & 0_{-} \end{bmatrix}$
    **end**
  **end**
**end**



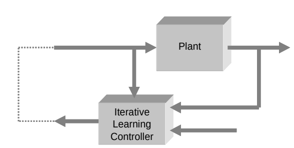
# Simulation Illustration

- Consider the following unstable system:

$$x_{k+1} = \begin{bmatrix} -0.50 & 0.00 & 0.00 \\ 1.00 & 2.04 & -1.20 \\ 0.00 & 1.20 & 0.00 \end{bmatrix} x_k + \begin{bmatrix} 1.0 \\ 0.0 \\ 0.0 \end{bmatrix}$$

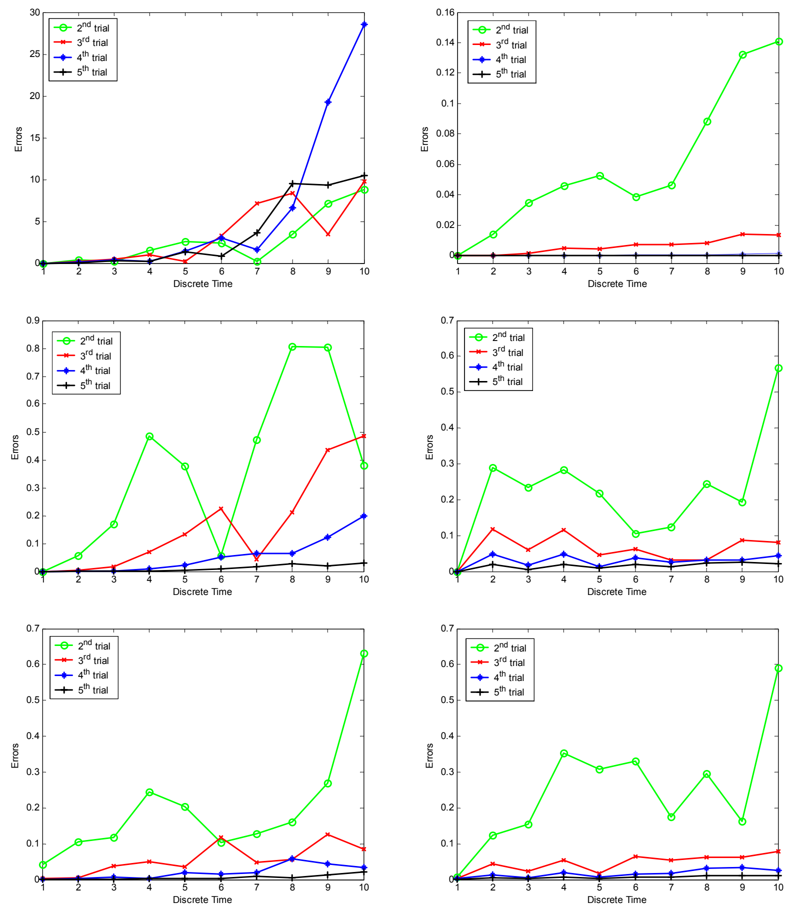
$$y_k = \begin{bmatrix} 1.0 & 2.5 & -1.5 \end{bmatrix} x_k,$$

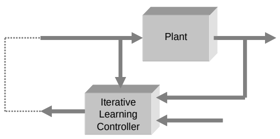
- A sinusoidal reference signal was used, with a trial length of ten time steps.
- For LMI solutions, the free online Matlab software $SeDuMi$ and $SeDuMiInt$ were used.
- We consider six cases:
  1. Arimoto only gain, fixed at $\gamma = 0.5$
  2. Unstructured learning gain matrix
  3. Causal LTI ILC with fixed band size
  4. Noncausal LTI ILC with fixed band size
  5. Causal LTV ILC with fixed band size
  6. Noncausal LTV ILC with fixed band size
- It is interesting to note that the LMI solution for Case 2 was in fact $H^{-1}$.
- Also, we see that monotonic convergence was improved by the use of non-causal gains.



# Simulation Illustration (cont.)

**Upper-left**: no LMI; **Upper-right**: using $H^{-1}$
**Middle-left**: causal LTI with band size = 3; **Middle-right**: causal LTV with band size = 3
**Bottom-left**: non-causal LTI with band size = 3; **Bottom-right**: non-causal LTV with band size = 3



# Comments about Monotonic ILC

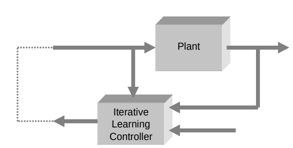
- Guaranteeing monotonic convergence of an ILC system is practically important and is theoretically desirable.

- Both higher-order-in-time and first-order-in-iteration have been analyzed with respect to monotonic convergence.

- We found that time-varying learning gains could be used for monotonic convergence. This is practically important because without using causal and noncausal bands, the monotonic convergence can be achieved.

- If we just consider the time domain, it is very difficult to guarantee the monotonic condition, while in the iteration domain, the monotonic condition can be achieved relatively easily.

- Various monotonic convergence conditions under various ILC algorithms have been studied. In particular, we have shown that the LMI tool box can be used to design monotonically-convergent ILC algorithms.



# Outline

- Iterative Learning Control (ILC)
- Monotonic Convergence via Supervector Framework
- Current-Cycle Feedback Approach
- Non-Causal Filtering ILC Design
  - Examples
  - Optimal PD-type ILC Scheme: How to Design
  - Optimal PD-type ILC Scheme: Averaged Derivative
  - Remarks
- Time-Varying ILC Design
- LMI Approach to ILC Design