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
Part 4: Robust ILC

Outline

- Review and Summary

- Interval ILC
  - Problem 1: Analysis of Maximum Allowable Perturbation
  - Problem 2: Design for Maximum Allowable Perturbation
  - Problem 3: Convergence Analysis
  - Problem 4: Design for Convergence (with Interval Model Conversion)

- Iteration-Domain $H_\infty$ ILC Design
Review and Summary

- Iterative learning control (ILC) is a control strategy for systems that execute the same trajectory, motion, or operation over and over.

- Consider the following single-input, single-output (SISO) 2-dimensional plant:

  \[
  x_k(t + 1) = A x_k(t) + B u_k(t) \\
  y_k(t) = C x_k(t)
  \]

  where \( t \) represents the discrete time point along the time axis (assumed finite) and the \( k \) represents the iteration trial number along the iteration axis.

- Taking \( z \)-transforms in time and defining \( G(z) = C(zI - A)^{-1}B + D \), the plant can be written as

  \[
  Y(z) = G(z)U(z) = (h_m z^{-m} + h_{m+1} z^{-(m+1)} + h_{m+2} z^{-(m+2)} + \cdots)U(z),
  \]

  where \( m \) is the relative degree of the system, \( z^{-1} \) is the standard delay operator in time, and the parameters \( h_i \) are the standard Markov parameters of the system \( G(z) \).

- Further, introduce the input update equation

  \[
  u_{k+1}(t) = u_k(t) + L(z)(y_d(t + m) - y_k(t + m)),
  \]

  where \( L(z) \) is the learning filter.
Review and Summary (cont.)

- Define the lifted super-vectors as:

\[
U_k = (u_k(0), u_k(1), \ldots, u_k(N - 1)),
\]
\[
Y_k = (y_k(m), y_k(m + 1), \ldots, y_k(N - 1 + m)),
\]
\[
Y_d = (y_d(m), y_d(m + 1), \ldots, y_d(N - 1 + m)),
\]
\[
E_k = Y_d - Y_k = (E_k(m), E_k(m + 1), \ldots, E_k(N - 1 + m))
\]

- This gives the relationships \( Y_k = HU_k \) and \( U_{k+1} = U_k + \Gamma E_k \), where \( H \) and \( \Gamma \) are lower-triangular Toeplitz matrices representing the system and the learning filter \( L(z) \), respectively.

- The error propagation equation becomes \( E_{k+1} = (I - H\Gamma)E_k \), where

\[
H = \begin{bmatrix}
  h_m & 0 & 0 & \ldots & 0 \\
  h_{m+1} & h_m & 0 & \ldots & 0 \\
  h_{m+2} & h_{m+1} & h_m & \ldots & 0 \\
  \vdots & \vdots & \vdots & \ddots & \vdots \\
  h_{m+N-1} & h_{m+N-2} & h_{m+N-3} & \ldots & h_m
\end{bmatrix}
\]
\[
\Gamma = \begin{bmatrix}
  \gamma_{11} & \gamma_{12} & \gamma_{13} & \ldots & \gamma_{1N} \\
  \gamma_{21} & \gamma_{22} & \gamma_{23} & \text{noncausal} & \gamma_{2N} \\
  \gamma_{31} & \gamma_{32} & \gamma_{33} & \ldots & \gamma_{3N} \\
  \vdots & \vdots & \vdots & \ddots & \vdots \\
  \gamma_{N1} & \gamma_{N2} & \gamma_{N3} & \ldots & \gamma_{NN}
\end{bmatrix}
\]

- Adding iteration-varying reference, disturbance, and noise signals, the possibility of an iteration-varying plant, and uncertainty in the plant model gives the complete ILC framework.
Part 4: Robust ILC

Complete Framework

- The resulting design problem is to determine the learning algorithm so the “closed-loop system” in the iteration domain is optimally convergent (asymptotically or monotonically) along the iteration axis in an appropriate norm topology, subject to specific assumptions on the noises, disturbances, and uncertainty in the system.

- This gives rise to a categorization of problems to consider.

- In the remainder of this presentation we consider robust learning controller design for two types of uncertainty problems: interval ILC and iteration-domain $H_{\infty}$ ILC.
### Categorization of Problems

<table>
<thead>
<tr>
<th>$Y_d$</th>
<th>$D$</th>
<th>$N$</th>
<th>$C$</th>
<th>$H$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$Y_d(z)$</td>
<td>0</td>
<td>0</td>
<td>$\Gamma(w - 1)^{-1}$</td>
<td>$H_p$</td>
</tr>
<tr>
<td>$Y_d(z)$</td>
<td>0</td>
<td>0</td>
<td>$\Gamma(w - 1)^{-1}$</td>
<td>$H_p(w)$</td>
</tr>
<tr>
<td>$Y_d(z)$</td>
<td>$D(w)$</td>
<td>0</td>
<td>$C(w)$</td>
<td>$H_p$</td>
</tr>
<tr>
<td>$Y_d(w)$</td>
<td>$D(w)$</td>
<td>0</td>
<td>$C(w)$</td>
<td>$H_p$</td>
</tr>
<tr>
<td>$Y_d(w)$</td>
<td>$D(w)$</td>
<td>$N(w)$</td>
<td>$C(w)$</td>
<td>$H_p(w) + \Delta(w)$</td>
</tr>
<tr>
<td>$Y_d(z)$</td>
<td>$D(z)$</td>
<td>0</td>
<td>$C(w)$</td>
<td>$H(w) + \Delta(w)$</td>
</tr>
<tr>
<td>$Y_d(z)$</td>
<td>0</td>
<td>$w(t), v(t)$</td>
<td>$\Gamma(w - 1)^{-1}$</td>
<td>$H_p$</td>
</tr>
<tr>
<td>$Y_d(z)$</td>
<td>0</td>
<td>$w(t), v(t)$</td>
<td>$C(w)$</td>
<td>$H_p$</td>
</tr>
<tr>
<td>$Y_d(z)$</td>
<td>0</td>
<td>0</td>
<td>$\Gamma(w - 1)^{-1}$</td>
<td>$H^I$</td>
</tr>
<tr>
<td>$Y_d(z)$</td>
<td>$D(z)$</td>
<td>$w(t), v(t)$</td>
<td>$C(w)$</td>
<td>$H(z) + \Delta H(z)$</td>
</tr>
<tr>
<td>$Y_d(z)$</td>
<td>$D(w)$</td>
<td>$w(k,t), v(k,t)$</td>
<td>$C(w)$</td>
<td>$H(w) + \Delta H(w)$</td>
</tr>
<tr>
<td>$Y_d(z)$</td>
<td>0</td>
<td>$w(k,t), v(k,t)$</td>
<td>$\Gamma(k)$</td>
<td>$H_p + \Delta H(k)$</td>
</tr>
<tr>
<td>$Y_d(z)$</td>
<td>0</td>
<td>$\tilde{H}$</td>
<td>$\Gamma$</td>
<td>$H_p$</td>
</tr>
<tr>
<td>$\vdots$</td>
<td>$\vdots$</td>
<td>$\vdots$</td>
<td>$\vdots$</td>
<td>$\vdots$</td>
</tr>
</tbody>
</table>
Part 4: Robust ILC

Interval ILC

- Prehistory of automatic control
- Primitive period
- Classical control
- Modern control
- Classic control
- Nonlinear control
- Estimation
- Robust control
- Optimal control
- Adaptive control
- Intelligent control
- $H_{\inf}$
- Interval
- Fuzzy
- Neural Net
- ILC
- ...
Interval Definitions

• A scalar \(a\) is called an **interval parameter** if it lies between two extreme boundaries according to \(a \in [a, \bar{a}]\), where \(a\) is the minimum value of \(a\) and \(\bar{a}\) is the maximum value of \(a\). The superscript \(I\) is used to represent an interval variable, e.g., \(a \in a^I := [a, \bar{a}]\) denotes an interval scalar.

• An **interval matrix** \((A^I)\) is defined as:

\[
A^I = \left\{ A : A = \left[ a_{ij} \in [a_{ij}, \bar{a}_{ij}] \right] , i, j = 1, \cdots, n \right\},
\]

where \(\bar{a}_{ij}\) is the maximum extreme value of the \(i^{th}\) row and \(j^{th}\) column element of the uncertain plant, and \(a_{ij}\) is the minimum extreme value of the \(i^{th}\) row and \(j^{th}\) column element of the uncertain plant.

• The **upper bound matrix** \((\bar{A})\) is a matrix whose elements are \(\bar{a}_{ij}\). The **lower bound matrix** \((A)\) is a matrix whose elements are \(a_{ij}\). The **vertex matrices** \((A^v)\) are defined by:

\[
A^v = \left\{ A : A = \left[ a_{ij} \in \{a_{ij}, \bar{a}_{ij}\} \right] , i, j = 1, \cdots, n \right\}
\]

• If the Markov parameters of a system are intervals such as: \(h_i \in [\underline{h}_i, \bar{h}_i]\), then the system is said to have interval uncertainties. The **interval Markov matrix** of such a system is denoted as \(H^I\) and its **interval vertex Markov matrices** are denoted as \(H^v\).
Part 4: Robust ILC

Interval ILC

- The interval ILC problem is concerned with the analysis and design of the ILC system when the system to be controlled is subjected to interval uncertainties.

- In this section we consider this problem, assuming:
  
  - No disturbances and no noise.
  
  - First-order ILC, where \( C(w) = \Gamma \).
Interval ILC Problems

- We consider four problems:

  1. Given a Markov matrix $H$ and a gain matrix $\Gamma$, how much perturbation $\Delta H$ can the system tolerate and still be stable?
    - This analysis question was addressed at the 2005 ACC.

  2. Given a nominal Markov matrix $H_0$, find $\Gamma$ to maximize the perturbation $\Delta H$ that can be allowed while still allowing convergence.
    - This design question was addressed at the 2005 ACC.

  3. Given an interval Markov matrix $H^I$ and a gain matrix $\Gamma$, what are the stability and convergence properties of the closed-loop system?
    - This analysis question was addressed at the 2005 IFAC.

  4. Given an interval Markov matrix $H^I$, design $\Gamma$, so as to achieve desired stability and convergence properties of the closed-loop system.
    - This design question was addressed at the 2005 ISIC and in a TAC paper.
    - An intermediate step in this problem is to solve for $[h_i, \bar{h}_i]$ given the interval plant $A^I$. 

Part 4: Robust ILC

Reminder: ILC Stability/Convergence Conditions

- When Arimoto-like gains and purely causal gains are used, the asymptotic stability condition is defined as:
  \[ |1 - \gamma_{ii} h_1| < 1, \ i = 1, \ldots, n, \]
  where \( h_1 \) is the first non-zero Markov parameter.

- When non-causal gains are used, the asymptotic stability condition is defined as:
  \[ \rho(I - H\Gamma) < 1 \]
  where \( \rho \) is the spectral radius of \((I - H\Gamma)\), and \( H \) is the Markov matrix.

- Monotonic convergence is defined in an appropriate norm topology as follows:
  
  - If \( \|I - H\Gamma\|_1 < 1 \), then \( \|E_k\| \) is monotonically convergent to zero in the \( l_1 \)-norm topology.
  
  - If \( \|I - H\Gamma\|_\infty < 1 \), then \( \|E_k\| \) is monotonically convergent to zero in the \( l_\infty \)-norm topology.
Part 4: Robust ILC

Outline

• Review and Summary

• Interval ILC
  – Problem 1: Analysis of Maximum Allowable Perturbation
  – Problem 2: Design for Maximum Allowable Perturbation
  – Problem 3: Convergence Analysis
  – Problem 4: Design for Convergence (with Interval Model Conversion)

• Iteration-Domain $H_\infty$ ILC Design
Part 4: Robust ILC

Problem Setup

- Assume the plant is $Y_k = HU_K$
- Assume first-order ILC with an update equation $U_{k+1} = U_k + \Gamma E_k$
- This gives the following evolution of the error vector in iterations:

$$E_{k+1} = (I - H\Gamma)E_k,$$

where $E_k = Y_d - Y_k$ and $\Gamma$ is the learning gain matrix defined as

$$\Gamma = \{\gamma_{ij}\}, i, j = 1, \cdots, n.$$

- The ILC algorithm is called Arimoto-like when the ILC gains are $\gamma_{ij} = 0, i \neq j$ and $\gamma_{ij} = \gamma, i = j$.
- The gains $\gamma_{ij}$ are called causal ILC gains for $i > j$ and non-causal ILC gains for $i < j$.
- If the gains do not exhibit Toeplitz-like symmetry we call the learning algorithm time-varying.
- We also refer to the band size of $\Gamma$. For example,
  - If band size is 1, only a diagonal line composed of Arimoto-like gains is used.
  - If band size is 2, then one causal diagonal line and one noncausal diagonal line are used in addition to Arimoto-like gains.
Some Definitions

- The *interval radius matrix* ($\Delta H^r$) is defined as $\Delta H^r = \frac{\bar{H} - H}{2}$
- For convenience, the symbols $T$ and $T^I$ are introduced, where
  \[ T \equiv I - H\Gamma, \quad T^I \equiv I - H^I\Gamma \]
- Then, the following notation is defined:
  \[ \Delta T = I - H\Gamma - (I - H^I\Gamma) = (H^I - H)\Gamma = \Delta H\Gamma, \]
  where $\Delta T$ is the interval uncertainty of the iterative learning control system, and $\Delta H$ is the interval uncertainty of the nominal Markov matrix.
- Also define
  \[ T^I_s = \begin{bmatrix} 0 & (I - H^I\Gamma)^T \\ (I - H\Gamma) & 0 \end{bmatrix}; T_s = \begin{bmatrix} 0 & (I - H\Gamma)^T \\ (I - H\Gamma) & 0 \end{bmatrix}. \]
- We introduce the symbol, $\langle \cdot \rangle$ to represent the bigger norm value between a matrix and its transpose:
  \[ \langle \Delta T \rangle \equiv \max\{\|\Delta T\|, \|\Delta T^T\|\}, \]
  where $\| \cdot \|$ denotes any kind of matrix norm.
Asymptotic Stability Condition of Interval ILC

**Theorem 1** Given \( \Gamma \) designed for the nominal plant \( H \), if there exists a symmetric, positive definite matrix \( P \) that satisfies the constraint

\[
(I - H \Gamma)^T P (I - H \Gamma) - P = -I,
\]

then the maximum allowable interval uncertainty (AIU), \( \Delta H \), for which \( (I - (H \pm \Delta H) \Gamma) \) is guaranteed to have asymptotic stability is bigger than \( \Delta H \) that satisfies

\[
\langle \Delta H \rangle \equiv \frac{-\langle I - H \Gamma \rangle + \sqrt{\langle I - H \Gamma \rangle^2 + \frac{1}{\|P\|}}}{\langle \Gamma \rangle}.
\]

**Corollary 1** If \( \Gamma = H^{-1} \), the maximum AIU of the interval ILC is \( \frac{1}{\|\Gamma\|} = \frac{1}{\|H^{-1}\|} \).
Monotonic Convergence Condition of Interval ILC

**Theorem 2** Given $\Gamma$ designed for the nominal plant $H$, if there exists a symmetric, positive definite matrix $P_s$ that satisfies the constraint

$$T_s^T P_s T_s - P_s = -I_{2n \times 2n}$$

then the maximum AIU, $\Delta H$, for which $(I - (H \pm \Delta H)\Gamma)$ is guaranteed to have monotonic stability is bigger than $\Delta H$ that satisfies

$$\langle \Delta H \rangle_k \equiv \frac{-\|T_s\|_2 + \sqrt{\|T_s\|_2^2 + \frac{1}{\|P_s\|_2}}}{\langle \Gamma \rangle_k},$$

where $k$ is 1 or $\infty$.

**Corollary 2** The 2-norm based AIU is calculated as:

$$\langle \Delta H \rangle_2 \equiv \frac{-\|T_s\|_2 + \sqrt{\|T_s\|_2^2 + \frac{1}{\|P_s\|_2}}}{\langle \Gamma \rangle_k}$$

where $k = 1$ or $\infty$. 
Optimization (Problem 2)

• The purpose of optimization is to maximize $\|\Delta H_{\text{asym}}\|$ and $\|\Delta H_{\text{mono}}\|$ by designing $\Gamma$ for either asymptotic stability or monotonic convergence.

• To find the optimal $\Gamma$ that allows more interval uncertainties in terms of asymptotic stability, the following optimization scheme is suggested:

$$\max_{\Gamma} \Delta H_{\text{asym}}$$

s.t. $$(I - H\Gamma)^T P(I - H\Gamma) - P = -I.$$  

• The same optimization idea can be used for increasing the uncertainty interval of the system in terms of monotonic convergence. It is designed as:

$$\max_{\Gamma} \Delta H_{\text{mono}}$$

s.t. $T_s^T P_s T_s - P_s = -I_{2n \times 2n}.$
Part 4: Robust ILC

Simulation Illustration

• The following discrete system is considered in this presentation:

\[
x_{k+1} = \begin{bmatrix} -0.50 & 0.00 & 0.00 \\ 1.00 & 1.24 & -0.87 \\ 0.00 & 0.87 & 0.00 \end{bmatrix} x_k + \begin{bmatrix} 1.0 \\ 0.0 \\ 0.0 \end{bmatrix}
\]

\[
y_k = \begin{bmatrix} 2.0 & 2.6 & -2.8 \end{bmatrix} x_k,
\]

which has three poles (0.62 + j0.62, 0.62 − j0.62, and −0.50) and two finite zeros (0.65 and −0.71). We assume a zero initial condition.

• The simulation test is performed with the following reference sinusoidal signal:

\[Y_d = \sin(8.0 j/n),\]

where \(n = 10\) and \(j = 1, \cdots, n\). The band size is fixed at 3, and learning gains are determined by optimization. Since the gains of each band are not fixed at the same value, the ILC algorithm is considered to be linear, time-varying, and non-causal. The uniformly distributed random number generator of MATLAB was used to make interval uncertainties in Markov parameters according to:

\[h_i = h_i + \delta |h_i| w,\]

where \(w \in [-1, 1]\) is a uniformly distributed random number; and \(\delta\) is tuned to limit the interval amount (in matrix 2 norm).
Part 4: Robust ILC

Test Results

1. **Asymptotic stability test**: tests were performed using the ILC learning gain matrix designed from optimization with $\Delta_{\text{asym}} = 0.737$:

2. **Monotone convergence test**: tests were performed using the ILC learning gain matrix designed from optimization $\Delta_{\text{mono}} = 0.6954$:

- In following figures:
  - Left figures show the interval amount of random plants in matrix 2-norms
    - First row corresponds to perturbations less that $\Delta_{\text{asym}}$
    - Perturbation size is bigger on second row and biggest on third row.
  - Right figures show ILC performance corresponding to left figures
    - Circle-marked lines are the $l_2$-norm errors vs. ILC iteration number corresponding to the plant with the maximum matrix 2-norm
    - Diamond-marked lines are $l_2$-norm errors vs. ILC iteration number corresponding to the plant with the minimum matrix 2-norm, among random plants

- The figures confirm the sufficiency of the proposed optimization approach.
Test Results-1: AS

Fig. 1.1.a

Fig. 1.1.b

Fig. 1.2.a

Fig. 1.2.b

Fig. 1.3.a

Fig. 1.3.b

Fig. 1. Asymptotical stability test: Tests were performed using the ILC learning gain matrix designed from optimization.

1. Left figures show the interval amount of random plants in matrix 2-norms. Right figures show ILC performance corresponding to left figures. Circle marked lines are maximum $l_2$ norm errors and diamond marked lines are minimum $l_2$ norm errors among random plants.
Part 4: Robust ILC

Test Results-2: MC

Fig. 2.1. a

Fig. 2.1. b

Fig. 2.2. a

Fig. 2.2. b

Fig. 2.3. a

Fig. 2.3. b

Fig. 2. Monotone convergence test: Tests were performed using the ILC learning gain matrix designed from optimization-2. Left figures show the interval amount of random plants in matrix 2-norms. Right figures show ILC performance corresponding to left figures. Circle marked lines are maximum $l_2$ norm errors and diamond marked lines are minimum $l_2$ norm errors among random plants.
Conclusion

• In this section we have considered the interval ILC problem.

• We calculated bounds on the the maximum allowable uncertainty in the plant Markov parameters for both asymptotic stability and monotonic convergence.

• These bounds were then used to design the ILC learning gain matrix to maximize the asymptotic and monotonic stability radii of the nominal plant.

• Simulation results illustrated the ideas. This approach, though conservative, provides an effective scheme for designing a robust ILC system.
Outline

- Review and Summary

- Interval ILC
  - Problem 1: Analysis of Maximum Allowable Perturbation
  - Problem 2: Design for Maximum Allowable Perturbation
  - Problem 3: Convergence Analysis
    - Problem 4: Design for Convergence (with Interval Model Conversion)

- Iteration-Domain $H_\infty$ ILC Design
Part 4: Robust ILC

Research Motivation

- In the ILC literature, robust design of the learning gain matrix has been considered using standard techniques such as $H_\infty$-ILC, LQ-ILC, optimal-ILC, etc.

- However, much of the literature approaches this problem from the perspective of the time-domain axis.

- In this presentation we will study the stability of the ILC problem when the plant Markov parameters are subject to interval uncertainty, formulating the problem in the iteration-domain using the supervector notation.

- In the robust control literature there are numerous results related to Hurwitz stability for interval matrixes and Schur stability. Kharitonov’s theorem has also been very popular for interval matrix stability analysis.

- Here, we exploit these works to develop an analysis method for checking the convergence properties of the interval ILC problem.

- Similar to the Kharitonov vertex polynomial method, it will be shown that the extreme values of the interval Markov parameters provide a sufficient condition for checking the monotonic convergence of interval ILC systems.
Asymptotic Stability for Interval ILC

- For interval ILC asymptotic stability conditions, existing results can be adopted:

**Lemma 1** With a given interval matrix $A^I$, the spectral radius of $A^I$ is bounded by the maximum value of the spectral radii of vertex matrices $A^v$.


**Lemma 2** Let the interval matrix be given as $A \leq A^I \leq A$. If $\beta = \max\{\rho(MS_1), \rho(MS_2)\} < 1$, where $MS_1 = a_{ij}$ if $i = j$ and $MS_1 = \max\{|a_{ij}|, |a_{ij}|\}$ if $i \neq j$; $MS_2 = a_{ij}$ if $i = j$ and $MS_2 = \min\{-|a_{ij}|, -|a_{ij}|\}$ if $i \neq j$, then the interval matrix $A^I$ is Schur stable.


- These lead to the following result:

**Theorem 3** Let the first Markov parameter $h_1$ be an interval parameter given by $h_1^I \in [\underline{h_1}, \overline{h_1}]$ and let Arimoto-like/causal ILC gains be used in $\Gamma$. Then the interval ILC system, $E_{k+1} = (I - H^I \Gamma) E_k$, is asymptotically stable if

$$\max\{|1 - \gamma_{ii}\underline{h_1}|, |1 - \gamma_{ii}\overline{h_1}|\} < 1, \ i = 1, \cdots, n.$$
Asymptotic Stability for Interval ILC (cont.)

• Now consider the case of a general $\Gamma$.

• In $I - H^I \Gamma$, the interval matrix is $H^I$. So, the lower bound and the upper bound of $I - H^I \Gamma$ should be re-calculated:
  
  – For convenience, let $T = H \Gamma$, calculated as:

    \[
    t_{ij} = \sum_{k=1}^{i} h_k \gamma(i+1-k)_j, \quad i, j = 1, \ldots, n
    \]

    where $t_{ij}$ are elements of $T$ and $\gamma(i+1-k)_j$ are ILC learning gains.

  – Similarly define $T^I = H^I \Gamma$ and also define $P = I - T$ and $P^I = I - T^I$.

• The lower and upper bounds of $P^I$, i.e., $\underline{P}$ and $\overline{P}$, can be calculated easily from the lower triangular Toeplitz matrix structure of $T^I$.

• Then, using the lower and upper bounds of $P^I$, it can be shown from Lemma 1 that the maximum spectral radius of $I - H^I \Gamma$ occurs at one of vertex matrices, $P^v$, of $P^I$. 
Monotone Convergence Condition for Interval ILC

- For monotonic convergence of interval ILC, the following lemmas are developed first.

**Lemma 3** Let $x^I \in [\underline{x}, \overline{x}]$ be an interval parameter. Then for

$$y = |\gamma_{11}x^I + \gamma_{12}| + |\gamma_{21}x^I + \gamma_{22}|, \forall \gamma_{11}, \gamma_{12}, \gamma_{21}, \gamma_{22} \in \mathbb{R},$$

the $\max\{y\}$ occurs at a vertex point of $x$ (i.e., $x^v \in \{\underline{x}, \overline{x}\}$).

**Lemma 4** Let $x^I \in [\underline{x}, \overline{x}]$ be an interval parameter. Then for

$$y = |\gamma_{11}x^I + \gamma_{12}| + |\gamma_{21}x^I + \gamma_{22}| + \cdots + |\gamma_{n1}x^I + \gamma_{n2}|, \forall \gamma_{i1}, \gamma_{i2} \in \mathbb{R}, i = 1, \cdots, n,$$

the $\max\{y\}$ occurs at one of vertex points of $x^v$ (i.e., $x^v \in \{\underline{x}, \overline{x}\}$).

**Lemma 5** Let $x^j \in [\underline{x^j}, \overline{x^j}], j = 1, \cdots, m$ be interval parameters (for convenience we omit the superscript $I$ and $v$). Then for

$$y = |(\gamma^1_{11}x^1 + \gamma^1_{12}) + \cdots + (\gamma^m_{11}x^m + \gamma^m_{12})| + \cdots + |(\gamma^1_{n1}x^1 + \gamma^1_{n2}) + \cdots + (\gamma^m_{n1}x^m + \gamma^m_{n2})|,$$

$$\forall \gamma^j_{i1}, \gamma^j_{i2} \in \mathbb{R}, i = 1, \cdots, n, j = 1, \cdots, m,$$

the $\max\{y\}$ occurs at the vertices of $x^j$. 
Monotone Convergence Condition for Interval ILC (cont.)

- Based on the lemmas of the preceding slides, the main results for monotone ILC conditions with interval uncertainty can be obtained, as summarized in the following theorems:

**Theorem 4** Given interval Markov parameters $h_i^I \in [h_i, \overline{h}_i]$, the interval ILC system is monotonically convergent in the $l_\infty$-norm topology if

$$\max\{\| I - H^v \Gamma \|_\infty \} < 1,$$

where $H^v$ are vertex Markov matrices of the interval plant.

**Theorem 5** Given interval Markov parameters $h_i^I \in [h_i, \overline{h}_i]$, the following equality is true:

$$\max\{\| I - H^I \Gamma \|_1 \} = \max\{\| I - H^v \Gamma \|_1 \},$$

where $\| \cdot \|_1$ is a matrix 1-norm, which is defined as: $\| A \|_1 = \max_{j=1,...,n} \sum_{i=1}^{n} |A_{ij}|.$
Example

- Let us consider a single-input, single-output system given as:

\[
A = \begin{bmatrix}
0.72 & 0.0 & 0.0 \\
1.0 & -1.04 & -0.81 \\
0.0 & 0.81 & 0.0
\end{bmatrix};
B = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix};
C = [1.0 - 0.98 - 1.09]
\]

which has first and second Markov parameters given as \(h_1 = CB = 1\) and \(h_2 = CAB = -0.266\).

- It is assumed that there are interval uncertainties in \(h_1\) and \(h_2\) given as

\[h_1^I \in [0.9, 1.1]; \quad \text{and} \quad h_2^I \in [-0.366, -0.166].\]

- For “Case-1” we suppose Arimoto-like ILC (only diagonal terms).

- For “Case-2” we use the inverse of the nominal (without interval) Markov matrix.

- To increase the robustness, in the Case-1 test, we used two different \(\gamma_1\): the dot-dashed lines are results with \(\gamma_1 = \frac{1}{h_1}\), and the solid lines are results with \(\gamma_1 = \frac{1}{h_1^I}\).
Part 4: Robust ILC

Norms of Case-1

![Diagram showing norms of Case-1 with vertex points marked on the 3D plot. The intervals for $h_1$ and $h_2$ are shown on the x and y axes, respectively. The z-axis represents the norms with values ranging from 0.9 to 2.4. The vertex points are indicated by specific markers.](image-url)
Part 4: Robust ILC

Norms of Case-2
Part 4: Robust ILC

Performance Test of Case-1

![Graph showing performance test results for Case-1 with three different values of $h_1$. Each value has a specific trend in absolute errors over trial number.](image-url)
Performance Test of Case-2

![Graph showing performance test results with trial number on the x-axis and absolute errors on the y-axis. The graph includes lines for maximum error, minimum error, and average error with markers for different values of h1.]
Conclusions

• We have presented a stability analysis of the ILC problem when the plant Markov parameters are subject to interval uncertainty.

• It was shown that checking just the vertex Markov matrices of an interval plant is enough to determine the asymptotic stability and the monotonic convergence properties of the interval ILC system.

• This is a powerful result from a computational perspective.

• Next we consider the design problem: given an interval Markov matrix $H^I$, find $\Gamma$ so as to guarantee that the monotonic and asymptotic convergence conditions are satisfied.
Part 4: Robust ILC

Outline

• Review and Summary

• Interval ILC

  – Problem 1: Analysis of Maximum Allowable Perturbation

  – Problem 2: Design for Maximum Allowable Perturbation

  – Problem 3: Convergence Analysis

  – Problem 4: Design for Convergence (with Interval Model Conversion)

• Iteration-Domain $H_\infty$ ILC Design
Overview

- In this section we present an approach to:
  
  - Design the learning gain matrix using the super-vector framework to guarantee the monotone convergence of the output response on the iteration axis
  
  - For linear plants in which the $A$-matrix is an interval matrix.

- To do this, we must convert the interval uncertainty in the $A$-matrix into an uncertainty in the system Markov parameters. This is called interval model conversion.

- Then, we introduce the idea of interval matrix eigenpair bounds.

- From the interval matrix eigenpair bounds we can then deduce monotonic convergence conditions, which can then be used for robust design.
Interval Model Conversion

- Consider the nominal single-input, single-output discrete-time system model given in state space form:

\[
\begin{align*}
    x(t + 1) &= Ax(t) + Bu(t) \\
    y(t) &= Cx(t),
\end{align*}
\]

where \( A, B, \) and \( C \) are the matrices describing the system in the state space; and \( x(t), u(t), \) and \( y(t) \) are the state, input, and output variables, respectively.

- The system Markov parameters are calculated by \( h_k = CA^{k-1}B \). These Markov parameters form the Markov matrix, denoted \( H \), in the usual way.

- Suppose that \( A \) is subject to interval model uncertainty according to \( A \in [\underline{A}, \overline{A}] \) and denote this interval matrix as \( \underline{A}^{I} \).

- Given \( A, B, \) and \( C, \) in particular \( A \in [\underline{A}, \overline{A}] \), the interval model conversion is defined as finding \( \underline{h}_k \) and \( \overline{h}_k \) such that \( h_k \in [\underline{h}_k, \overline{h}_k] \), where the underline (\( \cdot \)) represents the minimum value of (\( \cdot \)) and the overline (\( \cdot \)) represents the maximum value of (\( \cdot \)).
Basic Definitions

• The $n \times n$ interval matrix set $A^I$ is defined as

$$A^I = \left\{ A = [a_{ij} : a_{ij} \in [\underline{a}_{ij}, \overline{a}_{ij}]] \right\}$$

• The nominal matrix $A_0$ is defined to be

$$A_0 = \left[ a^0_{ij} : a^0_{ij} = \frac{a_{ij} + \overline{a}_{ij}}{2} \right]$$

where $a_{ij}$ and $\overline{a}_{ij}$ are the lower and upper bounds associated with the elements $a_{ij}$ of $A^I$.

• The interval perturbation matrix set $\Delta A^I$ is defined as

$$\Delta A^I = \left\{ \Delta A = [\Delta a_{ij} : \Delta a_{ij} = \overline{a}_{ij} - a^0_{ij} = a^0_{ij} - \underline{a}_{ij}] \right\}$$

where $a_{ij}$, $\overline{a}_{ij}$, and $a^0_{ij}$ are the lower bounds, upper bounds, and nominal values associated with the elements $a_{ij}$ of $A^I$, respectively.

• Note that we can also write

$$\Delta A^I = \left\{ \Delta A : \Delta A = A - A_0, \forall A \in A^I \right\}$$
Definitions (cont.) and Assumptions

- The vertex matrix set $A^v$ is a subset of the interval matrix set $A^I$ that is defined as:
  \[
  A^v = \left\{ A \in A^I : a_{ij} \in \{a_{ij}, \bar{a}_{ij}\} \right\}
  \]

- Next, to proceed we make two assumptions. The first is a technical assumption. Although we conjecture that this assumption can be relaxed, it is easiest to proceed with the following requirement. The second is more practical, as will be described in a later remark.

**Assumption 1** Any matrix $A \in A^I$ is diagonalizable and can be decomposed as: $A = X \Lambda X^{-1}$, with $\Lambda = \text{diag}(\lambda_i)$, where $\lambda_i$ are the eigenvalues of $A$.

**Assumption 2** Any matrix $A \in A^I$ is Schur stable.
Power of Interval Matrix

- Throughout this presentation, we call $X$ the left eigenvector matrix, $\Lambda$ the eigenvalue matrix, and $Y := X^{-1}$ the right eigenvector matrix. We also define the matrix sets (might not be interval matrices)

  $$
  \Lambda_A = \{ \Lambda : A = X\Lambda X^{-1}, A \in A^I \}
  $$

  $$
  X_A = \{ X : A = X\Lambda X^{-1}, A \in A^I \}
  $$

  $$
  Y_A = \{ Y = X^{-1} : X \in X_A \}
  $$

- The set of the power $k$ of the interval matrix $A$ is defined as

  $$
  (A^I)^k = \{ A^k : A \in A^I \}.
  $$

- The set of the generalized power $k$ of the interval matrix set is defined as

  $$
  A_k^I = \{ A = X\Lambda^k Y : \Lambda \in \Lambda_A, X \in X_A, Y \in Y_A \}.
  $$

- Note that clearly

  $$
  (A^I)^k \subset A_k^I.
  $$
Remarks

- **Remark** From a computational perspective, it should be noticed that the boundary of the set \((A^I)^k\) can be estimated by multiplying the interval matrices using interval calculation software such as Intlab [1], but the result will be quite conservative as \(k\) increases and it requires huge amount of computation. Further, notice that the exact boundary of \((A^I)^k\) cannot be calculated mathematically or analytically.

- **Remark** On the other hand, since \((A^I)^k \subset A^I_k\), we can estimate the boundary of the set \((A^I)^k\) by estimating the boundary of the set \(A^I_k\), which is also easily done in Intlab, but which can also be done analytically, as we show below. Specifically, since the boundary of \(A^I_k\) can be estimated by using three different sets: \(\Lambda_A, X_A,\) and \(Y_A\), the lower and upper boundary of \((A^I)^k\) can be subsequently estimated.

- **Remark** From the fact that the boundary of \(A^I_k\) is estimated from \(\Lambda_A, X_A,\) and \(Y_A\), we observe that if the maximum eigenvalue of \(A \in A^I\) is less than 1, for all \(A^k \in A^I_k\), \(A^k\) will converge to zero as \(k \to \infty\) because \(\Lambda^k, \Lambda \in \Lambda_A,\) converges to zero as \(k \to \infty\). However, if the maximum eigenvalue is bigger than 1, any \(A^k \in A^I_k\) will diverge, hence the Markov parameters become bigger and our bound on the the uncertain interval boundary of \(h_k\) become bigger as \(k\) increases. For this reason we assumed \(A\) is stable.

- **MAIN POINT:** We can contain our original interval system inside a “bigger” interval system according to: \((A^I)^k \subset A^I_k\). Therefore, the remaining work is to estimate the boundaries of \(\Lambda_A, X_A,\) and \(Y_A\) and from these estimates to compute bounds on \(A^k\).
Interval Matrix Eigenpair Bounds and Bounds of Markov Parameters

- First-order perturbation theory can be used to find an analytical bounds of \( \Lambda_A \), \( X_A \), and \( Y_A \) from \( A^I \) using two lemmas:

**Lemma 6** The lower and upper boundary matrices of \( \Lambda_A \) are estimated from vertex matrices of \( A^I \) in real part and imaginary part separately.

**Lemma 7** The lower and upper boundary matrices of \( X_A \) and \( Y_A \) are estimated from vertex matrices of \( A^I \) in real part and imaginary part separately.

- Then, the interval boundaries of \( A^k \), where \( A \in A^I \), can be bounded using the inequality: \( (A^I)^k \subset A^I_k \), which provides the following relationship:

\[
\underline{A}^k \leq A^k \leq \overline{A}^k \leq \overline{A}^k
\]

where \( A^k \in (A^I)^k \) and \( A^k \in A^I_k \).

- Then, the interval boundaries of Markov parameters (ex: \( h_{k+1} = CA^k B \)) can be estimated such as:

\[
\underline{h}_{k+1} = C\underline{A}^k B; \quad \overline{h}_{k+1} = C\overline{A}^k B
\]

where \( C, B \) are constant vectors describing the system (1) and \( A^k \) are the interval matrices which are lower-bounded and upper-bounded by \( \underline{A}^k \leq A^k \leq \overline{A}^k \).
Part 4: Robust ILC

Monotone Convergence Condition

- Assuming that we calculated $h^I_k \in [h_k, \bar{h}_k]$ from $A^I$, let us discuss the stability of interval ILC system.

- The interval Markov matrix is expressed as:

$$H^I = \begin{bmatrix} h^I_1 & 0 & \cdots & 0 \\ h^I_2 & h^I_1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ h^I_n & h^I_{n-1} & \cdots & h^I_1 \end{bmatrix}$$

where $n$ is the total number of the discrete time points in time domain.

- Let $H_0$ be the nominal Markov matrix without interval uncertainty, calculated as:

$$H_0 = \frac{h^I_i + \bar{h}^I_i}{2}, i = 1, \cdots, n$$

- The following symbols $T_0$ and $T^I$ are used to represent super-vector ILC: $T_0 := I_{n \times n} - H_0 \Gamma$, and $T^I := I_{n \times n} - H^I \Gamma$, where $I_{n \times n}$ is an $n \times n$ identity matrix, and $\Gamma$ is a learning gain matrix.
Part 4: Robust ILC

Cont.

• Make the following substitutions:

\begin{align*}
    s_{ij}^1 &:= \bar{a}_{ij} \text{ if } i = j \\
    s_{ij}^1 &:= \max\{|a_{ij}|, |\bar{a}_{ij}|\} \text{ if } i \neq j \\
    s_{ij}^2 &:= a_{ij} \text{ if } i = j \\
    s_{ij}^2 &:= \min\{-|a_{ij}|, -|\bar{a}_{ij}|\} \text{ if } i \neq j,
\end{align*}

where \( s_{ij}^1 \) is \( i^{th} \) row and \( j^{th} \) column element of matrix \( S^1 \); \( s_{ij}^2 \) is an element of matrix \( S^2 \), and \( a_{ij}^I \) is an element of general interval matrix \( A^I \).

• **Lemma 8** Let an interval matrix be given element-wise as \( A \leq A \leq \bar{A}, \ A \in A^I \). If \( \beta = \max\{\rho(S^1), \rho(S^2)\} < 1 \), where \( \rho \) is the spectral radius, then the interval matrix set \( A^I \) is Schur stable [2].

• The monotonic convergence in 2-norm topology can be checked by Lemma 8 after small modifications. Consider the following error vector update law: \( E_{k+1} = (I - H^I \Gamma)E_k \), where the singular value of \((I - H^I \Gamma) = T^I \) is calculated as:

\[
\bar{\sigma}(T^I) = \rho \left[ (T^I)^T T^I \right] = \sqrt{\rho \left[ \begin{array}{c} 0 \\ T^I \end{array} \right] \left[ \begin{array}{c} (T^I)^T \ \\
0 \end{array} \right]} \]
Part 4: Robust ILC

Cont.

• So, if the following interval matrix is defined:

\[ H^I = \begin{bmatrix} 0 & (T^I)^T \\ T^I & 0 \end{bmatrix}, \]

the monotonic convergence property of the interval ILC system is then checked by analyzing the Schur stability of \( H^I \) in the 2-norm topology using the Lemmas above.

• Monotonic convergence conditions in 1 and \( \infty \) norm topologies can also be developed:

**Lemma 9** Given \( h_i \in [\underline{h}_i, \overline{h}_i] \), the interval ILC system is monotonically convergent, if

\[ \| I - H^v \Gamma \|_k < 1, \]

where \( k \) is 1 or \( \infty \); and \( H^v \) are vertex Markov matrices.
Part 4: Robust ILC

Design of Interval ILC

- **Suggestion 1** Let $h_{ij}^I$ be the $i^{th}$ row and $j^{th}$ column elements of $H^I$. If we define a matrix $M$ whose elements are given as:

$$m_{ij} = \max\{h_{ij}, \overline{h_{ij}}\}$$

then, we solve the following optimization problem to design $\Gamma$:

$$\min_{\Gamma} \rho(M)$$

s.t. $h_k \in [\|h_k\|, \|\overline{h_k}\|]$

- **Remark** We explain why the matrix $M$ is introduced. Let us define $s_{ij}^1$ ($S^1 = \{s_{ij}^1\}$) and $s_{ij}^2$ ($S^2 = \{s_{ij}^2\}$) such as:

$$s_{ij}^1 := \overline{h_{ij}} \text{ if } i = j;$$

$$s_{ij}^1 := \max\{|h_{ij}|, |\overline{h_{ij}}|\} \text{ if } i \neq j;$$

$$s_{ij}^2 := h_{ij} \text{ if } i = j;$$

$$s_{ij}^2 := \min\{-|h_{ij}|, -|\overline{h_{ij}}|\} \text{ if } i \neq j$$

Then, since matrix $H^I$ is symmetric, $S^1 = -S^2$; hence using the fact that $\rho(S^1) = \rho(S^2)$ and diagonal terms of $H^I$ are all zeros, we only check the spectral radius of the matrix composed of the off-diagonal terms of $S^1$, which is denoted as $M$ above. Therefore, Suggestion 1 is reasonable.
• If Lemma 9 is used, another optimization scheme is suggested without constraint:

**Suggestion 2** If \( k = 1 \) or \( \infty \), the following optimization is straightforward.

\[
\min_{\Gamma} \| I - Hv\|_k.
\]

• **Remark** The optimization is to minimize \( \| I - Hv\|_k \) using \( \Gamma \), where \( \Gamma \) is a band-fixed learning gain matrix. In a small band size, it is possible that there might not exist an optimization solution such that \( \| I - Hv\|_k < 1 \). In this case, the band size should be increased until the optimization algorithm finds \( \Gamma \) such that \( \| I - Hv\|_k < 1 \).

• **Remark** Depending on the interval ILC system, the optimization scheme suggested above may not find the optimization solution even with the fully populated learning gain matrix. In this case, the following control update law could be used:

\[
U_{k+1} = Q(U_k + \Gamma E_k),
\]

where \( Q \) is a time-invariant diagonal matrix. Then, since the error vector is updated by the following formula:

\[
E_{k+1} = Q(I - H\Gamma)E_k + (I - Q)Y_d
\]

it is easy to make \( \| Q(I - H\Gamma) \| < 1 \) by \( Q \) and \( \Gamma \). However, the remaining term \( (I - Q)Y_d \) makes a non-zero steady-state error. This is a trade-off.
Example

- Let us assume that the following simple discrete servo system was given from a continuous system:

\[
\begin{align*}
    x_1(k+1) &= a_{11}x_1(k) + a_{12}x_2(k) + b_1u(k) \\
    x_2(k+1) &= a_{21}x_1(k) + a_{22}x_2(k) + b_2u(k) \\
    y(k) &= c_1x_1(t) + c_2x_2(k),
\end{align*}
\]

where \(b_1 = 2\), \(b_2 = 0.5\), \(c_1 = 1\), \(c_2 = 0\), interval parameters are bounded as: \(-0.74 \leq a_{11} \leq -0.66\), \(-0.53 \leq a_{12} \leq -0.47\), \(0.95 \leq a_{21} \leq 1.05\), and \(0.19 \leq a_{22} \leq 0.21\), and \(u\) is the control force.

- For this system we have \(A_0\) and \(\Delta A^I\) given by:

\[
A_0 = \begin{bmatrix} -0.7 & -0.5 \\ 1.0 & 0.2 \end{bmatrix}; \quad \Delta A = |\Delta A| = \begin{bmatrix} 0.04 & 0.03 \\ 0.05 & 0.01 \end{bmatrix}
\]

where \(|\Delta A|\) is the matrix composed of the absolute values of \(\Delta A\) element-wise.
Part 4: Robust ILC

Results

Left: Calculated interval uncertain boundaries of Markov parameters. Right: ILC convergence test.
Conclusion

• We have shown how to design a robust iterative learning controller for plants subject to interval uncertainty in their $A$-matrix.

• The interval uncertainty of the system plant was converted into the super-vector iterative learning control system (i.e., into an interval Markov matrix).

• Optimization schemes were suggested based on an interval matrix stability analysis method and a norm-based method.

• In the case of the norm-based method, the ILC learning gain matrix guaranteed monotonic convergence of the uncertain ILC system with zero steady-state error.

• However, the interval matrix method only guaranteed monotonic convergence with non-zero steady error.

• From these results, we conclude that the norm-based method is less conservative than the interval matrix method. However, the norm based method requires much more computational time than the interval matrix method.
Part 4: Robust ILC

Experimental Test: Setup
Part 4: Robust ILC

Impulse Responses and Desired Trajectory

- Interval ranges vs. Markov parameters
- Speed (radians/s) vs. Time (seconds)

IEEE ICMA 2006 Tutorial Workshop – Iterative Learning Control: Algebraic Analysis and Optimal Design
Part 4: Robust ILC

Achieved results

![Graphs showing achieved results](image-url)
Part 4: Robust ILC

Iteration-Domain $H_\infty$ ILC

Prehistory of automatic control

Primitive period

Classical control

Modern control

Classic control
Nonlinear control
Estimation
Robust control
Optimal control
Adaptive control
Intelligent control

H_inf
Interval
Fuzzy
Neural Net
ILC
...

H_inf ILC

IEEE ICMA 2006 Tutorial Workshop – Iterative Learning Control: Algebraic Analysis and Optimal Design
For the most part, existing work has focused on ILC design for performance improvement, assuming the plant is iteration-invariant and with external disturbances treated along the time axis.

To date, iteration axis robustness has not been treated in a systematic way.

A new framework is suggested for robust ILC design assuming both time-varying model uncertainty and iteration-varying external disturbances.

Using the super-vector approach, we can easily incorporate iteration-varying disturbances and the system can be analyzed using discrete (iteration axis) frequency domain techniques.
Part 4: Robust ILC

Review: Higher-Order ILC

- Higher-order update:

\[
U_{k+1} = (I - D_{n-1})U_k + (D_{n-1} - D_{n-2})U_{k-1} + \cdots \\
+ (D_2 - D_1)U_{k-n+2} + (D_1 - D_0)U_{k-n+1} + D_0U_{k-n} \\
+ N_n E_k + N_{n-1} E_{k-1} + \cdots + N_1 E_{k-n+1} + N_0 E_{k-n}
\]

- Taking the “\(w\)-transform” of the ILC update equation, combining terms, and simplifying gives

\[
(w - 1)D_c(w)U(w) = N_c(w)E(w)
\]

where \(D_c(w) = w^n + D_{n-1}w^{n-1} + \cdots + D_1w + D_0 \), and \(N_c(w) = N_n w^n + N_{n-1} w^{n-1} + \cdots + N_1 w + N_0 \)

- This can also be written in a matrix fraction as \(U(w) = C(w)E(w)\), where \(C(w) = (w - 1)^{-1}D_c^{-1}(w)N_c(w)\).

- For this update law the repetition-domain closed-loop dynamics become:

\[
G_{cl}(w) = H \left( I + \frac{I}{(w - 1)C(w)H} \right)^{-1} \frac{I}{(w - 1)}C(w), \\
= H[(w - 1)D_c(w) + N_c(w)H]^{-1}N_c(w).
\]
Higher-order ILC as a MIMO control problem

- Figure 1-(a) below depicts the general higher-order ILC problem as a MIMO control problem with the plant $H$. This figure highlights the fact that the ILC process (1) inherently is a relative degree one process; and (2) should have an integrator in order to converge to zero steady-state error.

- However, it can also be noted that the controller $C(w)$ has relative degree zero. This makes it convenient to consider the reformulation shown in Figure 1-(b).

![Diagram](image-url)
Problem Formulation

- The integrator has been grouped with the plant, so that we now define a new plant

\[ H_p(w) = (w - 1)^{-1} H. \]

- We suppose that \( H \) is subject to a perturbation such as \( H = H_0 + \Delta H(w) \) where \( \Delta H(w) \) represents iteration varying uncertainty in the plant model.

- The plant is disturbed by a plant input disturbance \( d_I \) and a plant output disturbance \( d_o \).

- These disturbances and plant perturbation models lead to a standard \( H_\infty \) problem. Specifically, the design problem for the uncertain ILC system can be formulated as:

**Problem**: Given \( H_p(w) = (w - 1)^{-1} H \) and \( Y_d(w) \), find \( C(w) \) in Figure 1-(b) such that \( \|E_k\|_2 \) is minimum from the \( l_2 \)-bounded disturbances \( d_I \) and \( d_o \) and the closed-loop system is stable over all \( H = H_0 + \Delta H(w) \), with \( \|\Delta H(w)\|_\infty < \epsilon_H \).
Part 4: Robust ILC

Problem Formulation (cont.)

- **Definition** In super-vector ILC, the $l_2$ norm is defined along the iteration axis (i.e., in the $w$ domain). To distinguish this concept from the discrete-time domain, the iteration domain $l_2$ norm is written as: $\| \cdot \|_{2w}$ and denoted $l_2w$. For example, we replace $\| E_k \|_2$ in the problem statement by $\| E_k \|_{2w}$. In the same way, the iteration domain $H_\infty$ norm is written as: $\| \cdot \|_{\infty w}$

- Notice that the problem set-up above indicates that the higher order ILC system can be synthesized from the $H_\infty$ framework. That is, the minimization problem of $\| E(w) \|_{2w}$ is translated as the minimization problem of

$$\| T_{EW} \|_{\infty w} = \sup_{W_k \in l_2w} \frac{\| E_k \|_{2w}}{\| W_k \|_{2w}}$$

such that

$$\| T_{EW} \|_{\infty w} < \gamma$$  \hspace{1cm} (1)

where $W = [d_I, d_o]^T$, $\| E_k \|_{2w} = \sum_{k=0}^{\infty} E(k)^T E(k)$ and $\| W_k \|_{2w} = \sum_{k=0}^{\infty} W(k)^T W(k)$. When $\gamma$ is not fixed, this is an optimal $H_\infty$ ILC problem and when $\gamma$ is fixed this a the sub-optimal $H_\infty$ ILC.

- Note: we assume $E_k$ and $E(k)$ have the same meaning. Both represent the iteration trial number. For convenience, we use both symbols.
Remark In ILC, the minimization of $T_{EW}$ means the reduction of the $H_\infty$ gain of the transfer matrices from $W_k$ to $E_k$. In other words, the reference input $Y_d$ is not counted in this performance problem. So, minimization of $T_{EW}$ does not guarantee the minimization of $E_k = Y_d - Y_k$. Instead, the sensitivities of $d_I$ and $d_o$ to $E_k$ are reduced by minimization of $\|T_{EW}\|_\infty^w$. The minimization of $E_k = Y_d - Y_k$ is ensured by the presence of the integrator in the loop gain and by adequate solution of the robust stability problem.

In our work, we try to design the ILC controller $C(w)$ with fixed $\gamma = 1$. Further notice that:

1. $H_p(w)$ has what is called a structured perturbation, because we have

$$H_p(w) = (w - 1)^{-1}H = \frac{I}{(w - 1)}(H_0 + \Delta H(w)),$$

not

$$H_p = H_{p_0} + \Delta H(w).$$

That is, there is no modelling uncertainty associated with the integrator, as it is actually due to the controller, not the plant.

2. Figure 1-(b) above, with (1), is the classic $H_\infty$ robust control problem. Basically, the point is to formulate and solve any of the robust stability and robust performance problems with $H_\infty$ an existence solution.
Part 4: Robust ILC

Standard $H_\infty$ framework

- Standard $H_\infty$ approach.

\begin{align*}
    x_{k+1} &= Ax_k + B_1 w_k + B_2 u_k \\
    z_k &= C_1 x_k + D_{11} w_k + D_{12} u_k \\
    y_k &= C_2 x_k + D_{21} w_k + D_{22} u_k
\end{align*}

where $z_k = [z_k^1, z_k^2]^T$ is the performance outputs, $y_k$ is the observation output to be used in output feedback control, and $w_k = [d_I, d_o]^T$ are the exogenous inputs (disturbances in plant input and plant output).

Figure 2: Typical discrete $H_\infty$ diagram.
Part 4: Robust ILC

Algebraic $H_\infty$ Approach of ILC
(To reduce the sensitivities from iteration-varying disturbances)

- The higher-order ILC system can be written as:

$$
U_{k+1} = U_k + V_k + d_I \\
Y = HU_k + d_o
$$

where we have defined $V_k(w) = C(w)E(w)$. Matching (5) with (2), (3), and (4), we draw

![ILC $H_\infty$ diagram with plant input and output disturbances.](image)

- The performance penalty is selected as $Z_k = [z_{k1}, z_{k2}]^T = [E_k, D_{12}V_k]^T$, where $V_k = C(w)E_k$, such that the input plant sensitivity matrix (from $d_I$ to $E_k$) becomes $S_I = H_p(w)(I + C(w)H_p(w))^{-1}$ and the output sensitivity matrix becomes $S_o = (I + C(w)H_p(w))^{-1}$.
Part 4: Robust ILC

Algebraic $H_\infty$ ILC formulation

- From Figure 3, the following state-space form of the ILC equations can be derived:

\begin{align*}
U_{k+1} &= IU_k + B_1W_k + I V_k \\
Z_k &= [H, 0]^T U_k + D_{11}W_k + D_{12}V_k \\
Y_k &= HU_k + D_{21}W_k,
\end{align*}

where $H$ is the Markov matrix, $W_k = [d_I, d_o]^T$, $B_1 = [\alpha I \ 0]$, $V_k = C(w)E_k$, and $D_{11} = \begin{bmatrix} 0 & \alpha I \\ 0 & 0 \end{bmatrix}$, $D_{21} = [0, \alpha I]^T$ ($\alpha$ is used to limit the disturbance intensity).

- For analytical solution, let us define:

\begin{align*}
\Theta_1 &:= \begin{bmatrix} I - \alpha^2\bar{X} & 0 \\ 0 & I - \alpha^2 I \end{bmatrix}; \quad \tilde{B}_1 := [\alpha\bar{X}, 0]^T; \quad \tilde{B}_2 := \bar{X} + H; \quad \tilde{B}_3 := [\alpha\bar{X}, \alpha I]^T \\
\Theta_2 &:= I + \bar{X} + \alpha^2\bar{X}(I - \alpha^2\bar{X})^{-1}\bar{X} + \frac{\alpha^2}{1 - \alpha^2}I; \quad \Theta_3 := H + \bar{X} + \alpha^2\bar{X}(I - \alpha^2\bar{X})^{-1}\bar{X} \\
\overline{A} &:= I + \alpha^2(I - \alpha^2\bar{X})^{-1}\bar{X}; \quad \overline{B}_1 := [\alpha(I - \alpha^2\bar{X})^{-\frac{1}{2}}, 0] \\
\overline{B}_2 &:= I + \alpha^2(I - \alpha^2\bar{X})^{-1}\bar{X}; \quad \overline{C}_1 := \Theta_2^{-\frac{1}{2}}\Theta_3; \quad \overline{C}_2 := H; \quad \overline{D}_{21} := \frac{\alpha}{\sqrt{1 - \alpha^2}}I; \quad \overline{D}_{22} := \frac{\alpha^2}{1 - \alpha^2}I \\
\Theta_4 &:= I - \overline{C}_1\overline{Z}\overline{C}_1^T; \quad E_1 := \overline{C}_1\overline{Z}\overline{C}_1^T; \quad E_2 := \overline{C}_2Z\overline{A}^T + \overline{D}_{21}\overline{B}_1; \quad E_3 := \overline{C}_1\overline{Z}\overline{C}_2^T \\
\Theta_5 &:= \overline{D}_{21}\overline{D}_{21}^T + \overline{C}_2\overline{Z}\overline{C}_2^T + E_3^T\Theta_4^{-1}E_3; \quad \Theta_6 := E_2 + E_3^T\Theta_4^{-1}E_1
\end{align*}
Part 4: Robust ILC

Algebraic $H_\infty$: Main Theorem

**Theorem 6** For the higher-order ILC system given by (6), (7), (8), if there exist $\bar{X} > 0$ and $F = -\Theta_2^{-1}\Theta_3$ such that $\Theta_1 > 0$ and $I + \alpha^2\bar{X}(I - \alpha^2\bar{X})^{-1}\bar{X} - \Theta_3^T\Theta_2^{-1}\Theta_3 < 0$, and if there also exist $\bar{Z} > 0$ such that $\Theta_4 > 0$ and $\bar{Z}A\bar{Z}^T - \bar{Z} + B_1^T\bar{B}_1 + E_1^T\Theta_4^{-1}E_1 - \Theta_6^T\Theta_5^{-1}\Theta_6 < 0$, then the observer gain matrix $L := -\Theta_6^T\Theta_5^{-1}$ and the controller given as

$$\xi_{k+1} = A_c\xi_k + B_cY_k$$

$$V_k = C_c\xi_k$$

where $A_c := \bar{A} + (\bar{B}_2 + LD_{22})F + L\bar{C}_2$, $B_c = L$, and $C_c = F$, stabilize the system (6), (7), (8) and guarantee $\|T_{ZW}\|_{\infty\omega} < 1$.

**Remark** In the controller defined in Theorem 6, the dimensions of $A_c$, $B_c$, and $C_c$ are the same as that of the Markov matrix $H$. This implies that the controller $C(w)$ will be of order $N$ in iteration, a typical $H_\infty$ result.
Algebraic $H_\infty$ ILC with model uncertainty

- We now extend the results of the previous slides to include the model uncertainty $\Delta H$ in system (6), (7), (8). Figure 4 shows the ILC block diagram with model uncertainty and with the performance penalty given as $Z_k = D_{12} V_k + C_1 U_k$. In this case, the input plant sensitivity matrix (from $d_I$ to $Z_k$) is defined as:

$$S_I = C(w) H_p(w)(I + C(w) H_p(w))^{-1} + (wI - I)^{-1} (I + C(w) H_p(w))^{-1}$$

and the output plant sensitivity matrix (from $d_o$ to $Z_k$) is defined as:

$$S_0 = C(w)(I + C(w) H_p(w))^{-1} + C(w)((wI - I) (I + H_p(w) C(w))^{-1}.$$
**Part 4: Robust ILC**

### $H_\infty$ ILC with model uncertainty (cont.)

- The purpose of $H_\infty$ synthesis (with both external disturbances and model uncertainty) is to minimize those sensitivity matrices and to robust stabilize the system with uncertainty $\Delta H$.

- From Figure 4, the ILC system can be expressed in the state-space form as:

  \[
  U_{k+1} = IU_k + B_1 W_k + IV_k \tag{9}
  \]

  \[
  Z_k = C_1 U_k + D_{11} W_k + D_{12} V_k \tag{10}
  \]

  \[
  Y_k = (H + \Delta H) U_k + D_{21} W_k + D_{22} V_k, \tag{11}
  \]

  where $Z_k = [z_k^1, z_k^2]^T$, $W_k = [d_I, d_o]^T$, $A = I$, $B_1 = [\alpha I, 0]$, $B_2 = I$, $C_1 = [0, I]^T$, $D_{11} = 0_{2n \times 2n}$, $D_{12} = [I, 0]^T$, $C_2 = H + \Delta H$, $D_{21} = [0, \alpha I]$, and $D_{22} = 0$.

- In this case $(I, I)$ is stabilizable, $(H, I)$ is detectable, $D_{12}$ is full column rank, and $D_{21}$ is full row rank. Thus all the basic assumptions for the existence of an $H_\infty$ solution are satisfied.
Part 4: Robust ILC

$H_\infty$ ILC with model uncertainty (cont.)

To solve the $H_\infty$ robust performance problem with model uncertainty algebraically, from Figure 4, we define

$$
\begin{bmatrix}
\Delta H & 0 \\
\end{bmatrix} =
\begin{bmatrix}
\Delta H \\
0 \\
\end{bmatrix} I,
$$

where $M_1 = 0, M_2 = \Delta H, F = I, N_1 = I$, and $N_2 = 0$. Then typical $H_\infty$ synthesis can be performed with the augmented ILC system given as:

$$
x_{a,k+1} = A x_{a,k} + B_1 w_{a,k} + B_2 u_{a,k} 
$$

(12)

$$
z_{a,k} = C_1 x_{a,k} + D_{11} w_{a,k} + D_{12} u_{a,k} 
$$

(13)

$$
y_{a,k} = C_2 x_{a,k} + D_{21} w_{a,k} + D_{22} u_{a,k} 
$$

(14)

where $A = I, B_1 = [0, \alpha I, 0], B_2 = I, C_1 = \left[ \frac{I}{\sqrt{\epsilon}}, 0, I \right]^T, D_{11} = 0_{3n \times 3n}, D_{12} = [0, I, 0]^T, C_2 = H, D_{21} = \left[ \sqrt{\epsilon} \Delta H, 0, \alpha I \right], \text{ and } D_{22} = 0.$

Now, a similar result to that given in in Theorem 6 above can be derived. This process is quite messy, but straightforward, so we do not include this result here.
Simulation Illustrations

• Consider the following discrete system:

\[
x_{k+1} = \begin{bmatrix} 0.25 & 0.6 \\ 0.6 & 0 \end{bmatrix} x_k + \begin{bmatrix} 1.0 \\ 0.0 \end{bmatrix} u_k
\]

\[
y_k = [1.0 \ -1.3] x_k
\]

which has nominal eigenvalues at \(-0.5\) and \(0.75\). In the time axis, 50 discrete samples are used and in the iteration axis, 100 times iteration tests are performed.

• For this system, the maximum model uncertainty is assumed to be 10 percent of the nominal \(H\) (note, for convenience we do not pick up plants with uncertainty \(\Delta H(w)\), but rather use plants with interval uncertainty; however, this is acceptable as our plants still satisfy a norm-bound of the form \(\|\Delta H\|_{\infty w} < \epsilon_H\)).

• The external disturbances satisfy \(\| \cdot \|_{l_2w} < \alpha = 0.1\). Simulation tests were performed for the case of external disturbances alone and for the case of both external disturbances together with model uncertainty.
Part 4: Robust ILC

With external disturbances, without model uncertainty

- The left plots of the figure are the ILC performance designed from the MATLAB `dhinf` and from the inverse of $H$ along the iteration axis. The right plots of the figure are the results from the suggested algebraic method and from Arimoto-like gains. However, from these figures, it is not clear how to compare the performance differences between the different controller options.

- Note that the error does not go to zero as the number of iterations increase. That is because the goal is to have minimized the gain between the disturbances and the error. By assumption it cannot be driven to zero (though, the error between $Y_d$ and the output is in fact going to zero). But, the error will depend on the signals $d_I$ and $d_o$.

- Thus, to compare the controllers it is useful to consider the actual gain between the disturbances and the error. These are shown in next slide for a number of different plants defined by our uncertainty model.
With both external disturbances and model uncertainty

The summation of the plant input and output disturbances \((\alpha I)d_I\) and \((\alpha I)d_o\) and \(\| (T_E W) W_k \|_2^w \).
Part 4: Robust ILC

With both external disturbances and model uncertainty (cont.)

- The previous figure plots $\|E_k\|_2$ and $\|W_k\|_2$ as well as their ratios for 50 plants. Recall that the algebraic $H_\infty$ controller and MATLAB based $H_\infty$ controller were designed to achieve $\|T_{EW}\|_\infty < 1$ (but that the Arimoto-like gains and the inverse of $H$ do not guarantee $\|T_{EW}\|_\infty < 1$). However, this is only guaranteed for the nominal plant. We have not solved the complete, simultaneous robust stability and robust performance problem.

- From sub-figures (a) and (b) (ILC systems designed from Arimoto-like gains and $C(w) = H^{-1}$, respectively), we observe that the robust performance requirement $\|T_{EW}\|_\infty < 1$ is not achieved for many cases.

- However from (c) and (d) (ILC systems designed from MATLAB and the algebraic method, respectively), we observe that $\|T_{EW}\|_\infty < 1$ is achieved for most plants (exception due to $\Delta H$, as noted).

- Clearly, from these figures, we conclude that the ILC system designed based on $H_\infty$ methods are more robust than the first order ILC systems.
Conclusion

• We have presented a new framework to design ILC controllers using $H_\infty$, which can be effectively used for iteration-varying ILC systems.

• In our approach the $H_\infty$ design was done in the discrete frequency domain along the iteration axis (not the time axis), in the super-vector framework.

• We did not use any filtering to characterize the input and output disturbances. So, there was no way to further reduce the baseline errors. We are currently studying the use of weighting filters that can be used to reduce the baseline error in the iteration domain.

• With structured model uncertainty, $\mu$ synthesis can be used to reduce the conservativeness.

• The algebraic $H_\infty$ design is much faster than MATLAB based $H_\infty$ and enables us to understand the $H_\infty$ mechanism in ILC.
Part 4: Robust ILC

Outline

- Review and Summary

- Interval ILC
  - Problem 1: Analysis of Maximum Allowable Perturbation
  - Problem 2: Design for Maximum Allowable Perturbation
  - Problem 3: Convergence Analysis
  - Problem 4: Design for Convergence (with Interval Model Conversion)

- Iteration-Domain $H_{\infty}$ ILC Design