

# Iterative Machine Learning for Output Tracking

Santosh DEVASIA<sup>\*)</sup>

*U. of Washington, Seattle, WA 98195-2600, USA*

This article develops iterative machine learning (IML) for output tracking. The input-output data generated during iterations to develop the model used in the iterative update. The main contribution of this article to propose the use of kernel-based machine learning to iteratively update both the model and the model-inversion-based input simultaneously. Additionally, augmented inputs with persistency of excitation are proposed to promote learning of the model during the iteration process. The proposed approach is illustrated with a simulation example.

## §1. Introduction

Iterative learning methods, initially developed in e.g.,<sup>1)–3)</sup> improve the output-tracking performance by correcting the input based on the measured tracking error. For example, iterative control has led to some of the highest precision for output tracking, e.g., in scanning probe microscopy as demonstrated in, e.g.,<sup>4)–10)</sup> Note that sets of learned trajectories can be used to enable tracking of other trajectories, e.g., by designing the feedforward control input using pre-specified basis functions e.g., using polynomial functions<sup>11)</sup> or rational functions of the reference trajectory.<sup>12)</sup> Similarly, new desired output can be generated by considering different combinations of previously-learned output segments<sup>13)</sup> and the feedforward can be represented using radial basis functions that can be optimized for a range of task parameters.<sup>14)</sup> This article proposes a kernel-based machine learning approach to use augmented inputs to iteratively learn not just the inverse input  $u_d$  needed to track a specified output  $y_d$  but to also use the data acquired during the iteration process to estimate both (i) the model  $\hat{g}$  (and its inverse) for the control update, as well as (ii) the model uncertainty needed to establish bounds on the iteration gain for ensuring tracking-error reduction..

Conditions for convergence of iterative methods have been well studied in literature, e.g.,<sup>15)–24)</sup> For example, the need to invert the system  $g$  to find the perfect input  $u_d$  for tracking a desired output  $y_d$  has motivated the use of the inverse  $\hat{g}^{-1}$  of the known model  $\hat{g}$  of the system  $g$  in early iterative control development.<sup>3)</sup> Since convergence depends on the size of modeling error, improvements of the model through parameter adaptation with data acquired during the iteration was studied in, e.g.,<sup>25)</sup> for robotics application using a discrete-time implementation. Here, for each iteration step, the sampled input vector is mapped to the sampled output vector through a lower triangular matrix map. A stochastic version using such a lower-triangular map has been studied in.<sup>26)</sup> Even in the ideal case with no mod-

---

<sup>\*)</sup> S. Devasia is with the Mechanical Engineering Department, U. of Washington, Seattle, WA 98195-2600 USA e-mail: devasia@uw.edu (see <http://faculty.washington.edu/devasia/>).

eling uncertainty, the inverse of this matrix map leads to a stable inverse only if the system is minimum-phase (i.e., no zeros on the right hand side of the complex plane), e.g.,<sup>19),27)</sup> This restriction to minimum-phase systems also applies to the use of input-output data to estimate models that enhance portability of the data-based learning to other output trajectories, e.g.,<sup>25),27)</sup> The extension of iterative learning control for nonminimum phase systems using the noncausal inverse was initially proposed in.<sup>20)</sup> The frequency domain implementation of the noncausal inverse was studied in, e.g.,<sup>16),17)</sup> and discrete-time implementation was developed in,<sup>28)</sup> where noncausality was allowed by using full matrices for the input-output map and rational basis functions were used to enable portability between different trajectories in.<sup>29)</sup> Convergence to the desired output can be guaranteed (with the frequency domain approach) if the phase uncertainty in the model is less than 90 degrees and the iteration gain is sufficiently small.<sup>16),18)</sup> Such dependence on the phase uncertainty was also developed using a discrete Fourier transform approach in.<sup>17),30)</sup>

Convergence cannot be guaranteed in regions of the frequency domain where the phase uncertainty in the model  $\hat{g}$  is greater than 90 degrees. Frequency regions where convergence cannot be guaranteed can be reduced by using the input-output data generated during the iteration procedure.<sup>31)</sup> In particular, more recent model-less approaches use the input-output data from previous iteration steps to avoid the need to model the system explicitly and improve the convergence of the iterative approach.<sup>31),32)</sup> Nevertheless, such input-output data might have substantial error at some frequencies where the signal to noise ratio is small. Kernel-based Gaussian process regression (GPR)<sup>33)</sup> is well suited to such function-estimation from noisy data. This motivates the main contribution of the article — the use of a kernel-based iterative machine learning (IML) approach to predict the magnitude and phase response of the system as functions of frequency from the measured input-output data. An added advantage is that the IML approach also yields the anticipated model uncertainty, which can be used to design the iteration law using previous results from.<sup>16)–18)</sup>

A second contribution of this article is to propose the use of additional input to the iteration law for persistency of excitation to enhance the model learning. It is shown in the article that convergence cannot be guaranteed if the size of the model is small compared to the size of the error in the model data, e.g., due to noise in the output. The effect of the noise in output measurement can be reduced by increasing the size of the output by using larger input. This increase in the output-to-noise ratio, i.e., the ratio of measured output to known input in the frequency domain, reduces the error in the model data and improves the performance of the iteration procedure. Therefore, this article proposes the injection of additional input into the iteration law whenever the the input size falls below a threshold value. This persistence of excitation leads to smaller noise in the measured system-model data with the proposed approach. (The effect of this input augmentation is removed from the measured output, using the estimated model, before the input is updated at the next iteration step.) Note that similar use of input augmentation to ensure persistency of excitation is commonly used in adaptive control, e.g.,<sup>34)</sup> The improved

model can be used to better infer the inverse input for new output trajectories. In this sense, the proposed approach enhances the portability of the frequency-domain iterative learning control.

The proposed use of additional input to improve model learning during the iteration process, is general in the sense that it can be used with other modeling methods. For example, in the current work a non-parametric Gaussian process regression (GPR) is proposed to predict the system response from the measured input-output data.<sup>33)</sup> However, the proposed model update during the iteration steps can also be applied using parametric methods for modeling, e.g., see recent review in.<sup>35)</sup> The models could then be inverted (potentially, in the time domain) during iterations for input update — in contrast, the proposed approach directly measures and stores the frequency data of the model (and hence its inverse). The GPR provides the smoothing in the presence of noisy data. Similarly, there is flexibility in the selection of the kernel used for estimating the models. While the more common smooth squared exponential (SE) kernel is used in this article, the Matern class of kernels could be used for systems with sharper features in the frequency response, e.g., for under-damped systems<sup>33)</sup> and the approach could be applied to complex-valued kernels in the frequency domain.<sup>36)</sup> The concept of model update in the frequency domain proposed here could be used with spatial-domain iterations, e.g.,<sup>37),38)</sup> with model identification methods using repetitive trajectories,<sup>39)</sup> and with the stable spline kernel for machine learning in the time domain<sup>35)</sup> that guarantees bounded-input-bounded-output stability of the resulting models. Finally, the proposed approach can be used to speed up the learning of the different segments in segmented iterative control approaches, e.g.,<sup>13)</sup>

The paper begins with problem formulation in Section 2, where the standard model-inversion-based iterative control approach is briefly reviewed followed by clarification of the research problem and the proposed solution approach using kernel-based machine learning in Section 3. Convergence conditions are then developed in Section 4, which are used to redesign the iteration law to promote persistency of excitation. Additionally, the overall IML algorithm is presented at the end of Section 4. Simulation results and discussion illustrating the proposed approach are in Section 5 followed by the conclusions in Section 6.

## **§2. Problem formulation**

This section begins by discussing the system and limits of the model-inversion approach due to modeling error, followed by background on frequency-domain iterative control to correct for modeling error. Convergence conditions and approaches to reduce the modeling error from data are presented, followed by the research-problem statement.

### 2.1. Model-based inverse feedforward

Given a desired output  $y_d$ , model inversion can be used to find the feedforward input  $u = u_d$  that achieves the desired output  $y = y_d$  for a linear system of the form

$$y(s) = g(s)u(s) \quad (2.1)$$

as

$$u_d(\omega) = g^{-1}(\omega)y_d(\omega) = [a(\omega) + jb(\omega)]^{-1}y_d(\omega) \quad (2.2)$$

with the value of the model  $\hat{g}$  evaluated on the imaginary axis of the complex plane defined as  $\hat{g}(\omega) = \hat{g}(s)|_{s=j\omega}$  and  $j = \sqrt{-1}$ .

**Assumption 1 (System properties)** *The system  $g$  is not identically zero (i.e., it is non-trivial), is stable, and has hyperbolic zero dynamics, i.e., all zeros have a nonzero real parts.*

Note that if the known system model  $\hat{g}(\omega)$  has error at frequency  $\omega$ , i.e.,  $\hat{g}(\omega) \neq g(\omega)$ , then the inverse input  $\hat{u}_d(\omega)$

$$\hat{u}_d(\omega) = \hat{g}^{-1}(\omega)y_d(\omega) \quad (2.3)$$

found using the model  $\hat{g}(\omega)$  does not lead to exact output tracking of the desired output  $y_d(\omega)$ , i.e.,

$$g(\omega)\hat{u}_d(\omega) = g(\omega)\hat{g}^{-1}(\omega)y_d(\omega) \neq y_d(\omega). \quad (2.4)$$

### 2.2. Fixed-model-based iterative control

The tracking error caused by the modeling error,  $\hat{g}(\omega) \neq g(\omega)$  at frequency  $\omega$ , can be corrected iteratively, e.g., as<sup>17),18)</sup>

$$u_k(\omega) = u_{k-1}(\omega) + \rho_k(\omega)\hat{g}_k^{-1}(\omega)[y_d(\omega) - y_{k-1}(\omega)] \quad (2.5)$$

provided  $\hat{g}_k^{-1}(\omega) \neq 0$  at frequency  $\omega$ , where at each integer iteration-step  $k > 1$ , the input  $u_k$  is computed using Eq. (2.5) with a real-valued scalar, frequency-dependent, iteration gain  $\rho_k$  and the available system model  $\hat{g}_k$  and applied to the system to find the output  $y_k$ . Note that the term in the square bracket in Eq. (2.5) represents the output error. The iterative control in Eq. (2.5) converges at frequency  $\omega$  if the modeling error is sufficiently small, as shown in,<sup>17),18)</sup> and stated formally below.

**Lemma 1 (Output-tracking convergence)** *With a finite initial input  $u_1(\omega)$ , a fixed iteration gain  $\rho_k(\omega) = \rho(\omega)$ , and non-zero system model  $\hat{g}_k(\omega) = \hat{g}(\omega) \neq 0$  at frequency  $\omega$ , the iterations in Eq. (2.5) converges to the inverse input  $u_d(\omega)$ , i.e.,*

$$\lim_{k \rightarrow \infty} u_k(\omega) = u_d(\omega) = g^{-1}(\omega)[y_d(\omega)], \quad (2.6)$$

which results in exact tracking of the desired output  $y_d(\omega)$ , i.e.,

$$\lim_{k \rightarrow \infty} y_k(\omega) = \lim_{k \rightarrow \infty} g(\omega)u_k(\omega) = y_d(\omega), \quad (2.7)$$

if and only if the magnitude of the phase uncertainty  $\Delta_p$  in the model  $\hat{g}$  and the iteration gain  $\rho$  are sufficiently small

$$\begin{aligned} |\Delta_p(\omega)| &< \pi/2 \\ 0 < \rho(\omega) &< \frac{2 \cos [\Delta_p(\omega)]}{\Delta_m(\omega)}, \end{aligned} \quad (2.8)$$

where the magnitude uncertainty  $\Delta_m$  and the phase uncertainty  $\Delta_p$  are defined by

$$\frac{\hat{g}^{-1}(\omega)}{g^{-1}(\omega)} = \frac{g(\omega)}{\hat{g}(\omega)} = \Delta_m(\omega) e^{j\Delta_p(\omega)}. \quad (2.9)$$

**Proof:** This follows from Lemma 1 in.<sup>16),18)</sup> The phase condition is the same as in .<sup>17)</sup> Briefly, multiplying Eq. (2.5) by the system  $g$  and subtracting from the desired output  $y_d$  yields

$$y_d(\omega) - y_k(\omega) = \left(1 - \rho(\omega) \frac{g(\omega)}{\hat{g}(\omega)}\right) [y_d(\omega) - y_{k-1}(\omega)], \quad (2.10)$$

which will tend to zero with increasing iteration steps  $k$  if

$$\left|1 - \rho(\omega) \frac{g(\omega)}{\hat{g}(\omega)}\right| = \left|1 - \rho(\omega) \Delta_m(\omega) e^{j\Delta_p(\omega)}\right| < 1. \quad (2.11)$$

The lemma follows by squaring the right hand side of the above equation to obtain

$$1 - 2\rho(\omega)\Delta_m(\omega)\cos(\Delta_p(\omega)) + \rho^2(\omega)\Delta_m^2(\omega) < 1 \quad (2.12)$$

and removing one from both sides to yield (since  $\Delta_m > 0$  because  $g(\omega) \neq 0$ ,  $\hat{g}(\omega) \neq 0$ ),

$$\rho(\omega) [\rho(\omega)\Delta_m(\omega) - 2\rho(\omega)\cos(\Delta_p(\omega))] < 0. \quad (2.13)$$

Note that if the phase uncertainty is small, i.e.,  $|\Delta_p(\omega)| < \pi/2$ , then  $\cos(\Delta_p(\omega)) > 0$  and  $\rho(\omega)$  needs to be positive to satisfy Eq. (2.13). However, if the phase uncertainty  $|\Delta_p(\omega)|$  is greater than  $\pi/2$ , then a fixed  $\rho(\omega)$  cannot satisfy Eq. (2.13) for all uncertainties since  $\cos(\Delta_p(\omega))$  can potentially be positive or negative. Finally, when  $|\Delta_p(\omega)| = \pi/2$ , the left hand side is positive. ■

**Remark 1 (Small phase error)** *The above condition implies that tracking errors reduce at each iteration step, and the input iterations will converge to the desired inverse input as in Eq. (2.6) at each frequency  $\omega$  if and only if the iteration gain  $\rho(\omega)$  is chosen to be sufficiently small, provided the phase error  $\Delta_p(\omega)$  is less than 90 degrees.<sup>16)–18)</sup>*

x

### 2.3. Data-based model update for iterative control

The inverse model  $\hat{g}_k^{-1}(\omega)$  in the iterative control in Eq. (2.5), used to compute the input  $u_k(\omega)$  at iteration step  $k$ , can be estimated from the previously computed input  $u_{k-1}(\omega)$  and the corresponding output  $y_{k-1}(\omega)$  as

$$\hat{g}_k^{-1}(\omega) = \frac{u_{k-1}(\omega)}{y_{k-1}(\omega)} \quad (2.14)$$

if  $y_{k-1}(\omega) \neq 0$ . Provided the noise in the estimation of the model  $\hat{g}_k(\omega)$  ( $k > 1$ ) is small, the above approach can be used with an iteration gain of  $\rho_k(\omega) = 1$  to modify the iterative input-update law in Eq. (2.5) to<sup>31)</sup>

$$u_k(\omega) = \begin{cases} \frac{u_{k-1}(\omega)}{y_{k-1}(\omega)} y_d(\omega) & \text{if } y_{k-1}(\omega) \neq 0 \\ 0 & \text{otherwise} \end{cases} \quad (2.15)$$

and the initial input at  $k = 1$  is considered to be

$$u_1(\omega) = \alpha y_d(\omega) \quad (2.16)$$

where  $\alpha$  is a constant that can be chosen, e.g., to be the inverse of the estimated DC gain of the system. The tracking error can be made small if the noise in the measurements is small, as shown in.<sup>31)</sup>

**Remark 2 (Model filtering)** *The estimated inverse model  $\hat{g}_k^{-1}(\omega)$  in the iterative control in Eq. (2.5) at iteration step  $k$  can be replaced by a weighted average from all available estimates from iteration steps  $k_i \leq k$  to reduce the effect of noise.<sup>32)</sup>*

#### 2.4. The research problem

There are two main issues with current frequency-domain iterative approaches, e.g.,<sup>17), 18), 31)</sup> The first issue is that while these frequency-domain methods iteratively find the inverse input  $u_d$  to achieve a desired output  $y_d$ , they do not directly improve the ability to track a new output trajectory  $y_{d,2}$ , with potentially different frequency content. Secondly, current approaches do not improve estimates of the uncertainties  $(\Delta_m, \Delta_p)$  in the model  $\hat{g}$ . Note that lower model uncertainties can allow the use of larger iteration gains according to Eq. (2.8) and can lead to faster convergence.

The research problem is to iteratively learn the system model  $\hat{g}$  (and therefore, its inverse  $\hat{g}^{-1}$ ) and estimate the model uncertainties  $(\Delta_m, \Delta_p)$  while iteratively learning the inverse input  $u_d$  needed to track a desired output  $y_d$ .

### §3. Proposed machine learning of model

A kernel-based machine learning approach is proposed in this section to learn the system model  $\hat{g}$  during the iteration process. The Gaussian process assumption is clarified first in Subsection 3.1, followed by background on the Gaussian process regression (GPR) approach to estimate a general function  $f$  in Subsection 3.2, and lastly, by the application of the GPR to estimate the system model in Subsection 3.3.

#### 3.1. Model is assumed to be a Gaussian process

In the following, the real  $\hat{a}$  and imaginary  $\hat{b}$  components of the model  $\hat{g}$  are considered to be independent real-valued Gaussian processes as in.<sup>41)</sup>

**Assumption 2 (Gaussian process)** *The real  $\hat{a}$  and imaginary  $\hat{b}$  components of the system model  $\hat{g}$  are considered to be zero-mean, real-valued, independent Gaussian*

processes with covariance functions  $k_a(\omega, \omega')$  and  $k_b(\omega, \omega')$  respectively, i.e.,

$$\begin{aligned}\hat{a}(\omega) &= \mathcal{R}[g(\omega)] \sim GP(0, k_a(\omega, \omega')) \\ \hat{b}(\omega) &= \mathcal{I}[g(\omega)] \sim GP(0, k_b(\omega, \omega'))\end{aligned}\quad (3.1)$$

where the measured real  $m_a$  and imaginary  $m_b$  components are given by

$$\begin{aligned}m_a &= \hat{a}(\omega) + \epsilon_a \\ m_b &= \hat{b}(\omega) + \epsilon_b\end{aligned}\quad (3.2)$$

with additive, zero-mean, independent identically distributed Gaussian noise  $\epsilon_a, \epsilon_b$  with variance  $\sigma_a^2, \sigma_b^2$ .

**Remark 3 (Lack of knowledge about the model)** *The Gaussian process has zero mean in Assumption 2 — the estimated model  $\hat{g}(\omega)$  tends to zero at frequencies  $\omega$  far from the frequency region where data is available. The size of the acceptable modeling uncertainty (for using inversion-based input update) depends on the estimated model size, as discussed in the next section.*

**Remark 4 (Related real and imaginary components)** *The real and imaginary components of a causal linear transfer function  $g$  are related to each other by the Kramers-Kronig relations, e.g.,<sup>42)</sup> However, this relation between the real and imaginary components is not pointwise in frequency. For example, the real part  $\hat{a}(\omega)$  at a frequency  $\omega$  depends on the complex part  $\hat{b}(\cdot)$  over the entire frequency domain. The approach used here to separately estimate the real  $\hat{a}$  and imaginary  $\hat{b}$  components of the model  $\hat{g}$  is conservative and over-predicts the uncertainty. Therefore, it can lead to a smaller iteration gain  $\rho$  with slower convergence.*

### 3.2. Background: Gaussian process regression

The data-based estimation of the expected value general function  $f(\cdot)$  with  $n$  measurements  $m \in \mathcal{R}_n$  at frequencies  $\Omega \in \mathcal{R}_n$  at frequency  $\omega \in \mathcal{R}$  using Gaussian process regression (GPR) is stated formally in the following lemma, e.g.,<sup>33)</sup>

**Lemma 2 (Machine learning)** *Let  $f$  be a zero-mean real-valued Gaussian process over the frequency space*

$$f(\omega) \sim GP(0, k(\omega, \omega')) \quad (3.3)$$

*with covariance function  $k(\omega, \omega')$ , and measurements  $m \in \mathcal{R}_n$  given by*

$$m = f(\omega) + \epsilon \quad (3.4)$$

*with additive, zero-mean, independent identically distributed Gaussian noise with variance  $\sigma^2$ . Then, the prediction  $f_\star$  at any prediction frequency  $\omega \in \mathcal{R}$ , given  $n$  measured data*

$$\mathbf{m} = [m_1 \ m_2 \ \dots \ m_n]^T \quad (3.5)$$

*(with the superscript  $T$  indicating matrix transpose) at frequencies*

$$\Omega = [\omega_1 \ \omega_2 \ \dots \ \omega_n]^T, \quad (3.6)$$

is Gaussian<sup>33)</sup>

$$f_\star | \Omega, \mathbf{m}, \omega \sim \mathcal{N}(\bar{f}_\star, \sigma_{f_\star}^2), \quad (3.7)$$

where the predicted mean  $\bar{f}_\star$  and variance  $\sigma_{f_\star}^2$  at any prediction frequency  $\omega$  are given by

$$\begin{aligned} \bar{f}_\star(\omega) &= K(\omega, \Omega) [K(\Omega, \Omega) + \sigma^2 I]^{-1} \mathbf{m} \\ \sigma_{f_\star}^2(\omega) &= K(\omega, \omega) \\ &\quad - K(\omega, \Omega) [K(\Omega, \Omega) + \sigma^2 I]^{-1} K(\Omega, \omega) \end{aligned} \quad (3.8)$$

with  $K(\Omega, \omega)$  denoting the covariances evaluated at all pairs of measured frequencies  $\Omega \in \mathcal{R}_n$  and the prediction frequency  $\omega \in \mathcal{R}$ . The other covariance matrices  $K(\cdot, \cdot)$  are defined similarly.

**Proof:** See, e.g., Chapter 2 in.<sup>33)</sup> ■

### 3.3. Estimation of model from data

The data-based estimation of the real  $\hat{a}$  and imaginary  $\hat{b}$  components of the system model  $\hat{g}$  using GPR from the above Lemma 2, e.g.,<sup>33)</sup> is described below.

At iteration step  $k$ , given all the measured real  $\mathbf{m}_{a,k}$  and imaginary  $\mathbf{m}_{b,k}$  components at frequencies  $\Omega_k$ , the predictive means  $\bar{a}_{\star,k}(\omega)$ ,  $\bar{b}_{\star,k}(\omega)$  and variances  $\sigma_{a_{\star,k}}^2(\omega)$ ,  $\sigma_{b_{\star,k}}^2(\omega)$  of the real  $a(\omega)$  and imaginary  $b(\omega)$  components at prediction frequency  $\omega$  can be obtained by setting

$$f_\star = z_{\star,k}, \quad \sigma_{f_\star} = \sigma_{z_{\star,k}}, \quad \mathbf{m} = \mathbf{m}_{z,k}, \quad \Omega = \Omega_k \quad (3.9)$$

in Eq. (3.7) and Eq. (3.8) of the above Lemma 2 where the subscript  $z$  is replaced by either  $a$  or  $b$  depending on whether the real or imaginary component is being predicted.

The resulting model  $\hat{g}_k$  at the  $k^{th}$  iteration step is given by

$$\hat{g}_k(\omega) = [\bar{a}_{\star,k}(\omega) + j\bar{b}_{\star,k}(\omega)] := [\hat{a}_k(\omega) + j\hat{b}_k(\omega)]. \quad (3.10)$$

## §4. Convergence with machine learned model

Convergence of the iterative approach with the estimated model from machine learning can be quantified in terms of the size of the model uncertainties in its real  $\hat{a}_k$  and imaginary  $\hat{b}_k$  components, as shown in this section. This is followed by a discussion on the augmentation of the iterative input law (at frequencies where the signal to noise ration is low) to reduce the impact of measurement noise when estimating the model. This section concludes with the proposed algorithm.

### 4.1. Convergence under bounded uncertainties

#### 4.1.1. Convergence conditions

In the following lemmas, it is assumed that the size of the model uncertainty is bounded.



**Condition 1 (Bounds on model uncertainty)** *The deviations of the real and imaginary components  $a, b$  of the system in Eq. (2.2) from the corresponding components  $\hat{a}_k, \hat{b}_k$  of the model in Eq. (3.10) at iteration step  $k$ , are bounded as*

$$\begin{aligned} |a(\omega) - \hat{a}_k(\omega)| &\leq \Delta_{a,k}(\omega) \\ |b(\omega) - \hat{b}_k(\omega)| &\leq \Delta_{b,k}(\omega). \end{aligned} \quad (4.1)$$

**Remark 5 (Confidence intervals)** *Predictive confidence intervals can be used as bounds on the model uncertainties in Eq. (4.1). In this probabilistic setting, the chance of not converging can be made small, but the method does not guarantee that the tracking error will decrease. Nevertheless, if the model uncertainty falls outside of the specified confidence intervals at frequency  $\omega$  and the input  $u_k(\omega)$  starts to grow, then the resulting output  $y(\omega)$  will also become large due to the hyperbolic-internal dynamics  $g(\omega) \neq 0$  from Assumption 1. If the output error becomes large compared to the noise in the measurement at some frequency  $\omega$ , then this yields additional data at frequency  $\omega$  for improved model estimation, and input correction.*

Conditions on the magnitude and phase uncertainties can be developed based on the uncertainties in the real and imaginary components of the model, as shown below.

**Lemma 3 (Bounds on phase and magnitude)** *Under Condition 1, the magnitude uncertainty  $\Delta_{m,k}(\omega)$  and the phase uncertainty  $\Delta_{p,k}(\omega)$ , similar to Eq. (2.9),*

$$\Delta_{m,k}(\omega) e^{j\Delta_{p,k}(\omega)} = \frac{g(\omega)}{\hat{g}_k(\omega)} = \frac{a(\omega) + jb(\omega)}{\hat{a}_k(\omega) + j\hat{b}_k(\omega)} \quad (4.2)$$

satisfy

$$|\Delta_{m,k}(\omega)| \leq \frac{|\Delta_k(\omega) + \hat{g}_{abs,k}(\omega)|}{|\hat{g}_k(\omega)|} \quad (4.3)$$

and

$$\cos[\Delta_{p,k}(\omega)] \geq \frac{|\hat{g}_k(\omega)|^2 - \hat{g}_{abs,k}(\omega) \cdot \Delta_k(\omega)}{|\hat{g}_k(\omega)| |\hat{g}_{abs,k}(\omega) + \Delta_k(\omega)|}, \quad (4.4)$$

where

$$\begin{aligned} \Delta_k(\omega) &= \Delta_{a,k}(\omega) + j\Delta_{b,k}(\omega) \\ \hat{g}_{abs,k}(\omega) &= |\hat{a}_k(\omega)| + j|\hat{b}_k(\omega)|. \end{aligned} \quad (4.5)$$

**Proof:** An expression for the magnitude  $\Delta_{m,k}$  can be found from the definition in Eq. (4.2) as

$$\Delta_{m,k}(\omega) = \frac{|g(\omega)|}{|\hat{g}_k(\omega)|} \quad (4.6)$$

and the condition in Eq. (4.3) follows since the numerator is maximized by selecting the maximal  $|a(\omega)|, |b(\omega)|$  that satisfy the lemma's condition in Eq. (4.1), as in Eq. (4.3). The cosine of the phase angle between the model  $\hat{g}_k(\omega)$  and the system  $g(\omega)$  can be found using the dot product as

$$\cos[\Delta_{p,k}(\omega)] = \frac{\hat{g}_k(\omega) \cdot g(\omega)}{|\hat{g}_k(\omega)| |g(\omega)|}. \quad (4.7)$$

The numerator in Eq. (4.7)

$$\begin{aligned}\hat{g}_k(\omega).g(\omega) &= \hat{a}_k(\omega)a(\omega) + \hat{b}_k(\omega)b(\omega) \\ &= \hat{a}_k^2(\omega) + \hat{b}_k^2(\omega) + \delta_{a,k}(\omega)\hat{a}_k(\omega) + \delta_{b,k}(\omega)\hat{b}_k(\omega),\end{aligned}\quad (4.8)$$

where

$$\begin{aligned}a(\omega) &= \hat{a}_k(\omega) + \delta_{a,k}(\omega), & b(\omega) &= \hat{b}_k(\omega) + \delta_{b,k}(\omega) \\ |\delta_{a,k}(\omega)| &\leq \Delta_{a,k}(\omega), & |\delta_{b,k}(\omega)| &\leq \Delta_{b,k}(\omega)\end{aligned}\quad (4.9)$$

is minimized (can be negative) when  $\delta_{a,k}$  and  $\delta_{b,k}$  are chosen as

$$\begin{aligned}\delta_{a,k}(\omega) &= -[\text{sgn } \hat{a}_k(\omega)]\Delta_{a,k}(\omega) \\ \delta_{b,k}(\omega) &= -[\text{sgn } \hat{b}_k(\omega)]\Delta_{b,k}(\omega),\end{aligned}\quad (4.10)$$

which results in

$$\hat{g}_k(\omega).g(\omega) = \hat{a}_k^2(\omega) + \hat{b}_k^2(\omega) - \Delta_{a,k}(\omega)|\hat{a}_k(\omega)| - \Delta_{b,k}(\omega)|\hat{b}_k(\omega)| \quad (4.11)$$

and the numerator in the right hand side of Eq. (4.4). The denominator in Eq. (4.7) is maximized when the system magnitude  $|g(\omega)|$  is the largest possible, i.e., from Eq. (4.9)

$$\begin{aligned}\delta_{a,k}(\omega) &= [\text{sgn } \hat{a}_k(\omega)]\Delta_{a,k}(\omega) \\ \delta_{b,k}(\omega) &= [\text{sgn } \hat{b}_k(\omega)]\Delta_{b,k}(\omega),\end{aligned}\quad (4.12)$$

which results in the denominator in the right hand side of Eq. (4.4). Minimizing the numerator and maximizing the denominator of Eq. (4.7) results in the lemma's claim in Eq. (4.4). ■

Next, an iterative control law is designed to reduce the tracking error when the model uncertainties satisfy the bounds in Condition 1.

**Lemma 4 (Error reduction with iteration)** *Under Condition 1, and non-zero system model  $\hat{g}_k(\omega) \neq 0$  at frequency  $\omega$ , the iteration law in Eq. (2.5) reduces the tracking error, i.e.,*

$$|y_d(\omega) - y_k(\omega)| < |y_d(\omega) - y_{k-1}(\omega)| \quad (4.13)$$

provided the iteration gain  $\rho_k$  satisfies

$$0 < \rho_k < 2 \frac{|\hat{g}_k(\omega)|^2 - \hat{g}_{abs,k}(\omega) \cdot \Delta_k(\omega)}{|\hat{g}_{abs,k}(\omega) + \Delta_k(\omega)|^2} = \rho_k^*(\omega). \quad (4.14)$$

**Proof:** From Eqs. (4.3), (4.4) and (4.14)

$$0 < \rho_k(\omega) < \rho_k^*(\omega) \leq 2 \frac{\cos[\Delta_{p,k}(\omega)]}{\Delta_{m,k}(\omega)}. \quad (4.15)$$

Since  $\Delta_{m,k}(\omega) > 0$  (from Eq. (4.6), as the model  $\hat{g}_k(\omega)$  and the system  $g_k(\omega)$  are assumed to be non-zero at frequency  $\omega$ ),

$$\rho_k(\omega)\Delta_{m,k}(\omega) - 2\cos[\Delta_{p,k}(\omega)] < 0 \quad (4.16)$$

and as  $\rho_k(\omega) > 0$  from Eq. (4.15),

$$\begin{aligned} \rho_k(\omega) \Delta_{m,k}(\omega) [\rho_k(\omega) \Delta_{m,k}(\omega) - 2\cos[\Delta_{p,k}(\omega)]] &< 0 \\ 1 + \rho_k(\omega) \Delta_{m,k}(\omega) [\rho_k(\omega) \Delta_{m,k}(\omega) - 2\cos[\Delta_{p,k}(\omega)]] &< 1. \end{aligned} \quad (4.17)$$

Then, using re-arrangement of the terms similar to those in Eqs. (2.11) and (2.12), results in

$$\left| 1 - \rho_k(\omega) \frac{g(\omega)}{\hat{g}_k(\omega)} \right| < 1 \quad (4.18)$$

and the lemma follows from Eq. (2.10) with the iteration gain  $\rho_k(\omega)$ .  $\blacksquare$

**Remark 6 (Iteration gain)** *A large iteration gain can result in a large tracking-error reduction but it can also amplify the effect of noise.<sup>18)</sup> Iteration gains larger than one can lead to oscillatory convergence. In the following, the iteration gain  $\rho_k(\omega)$  is selected to satisfy the upper bound  $\rho_k^*(\omega)$  in Eq. (4.15), as*

$$\rho_k(\omega) = \underline{\rho}(\omega) \min \{ \rho_k^*(\omega), 1 \} < \rho_k^*(\omega), \quad (4.19)$$

with  $\underline{\rho}(\omega) \in (0, 1)$ .

**Remark 7 (Convergence with varying iteration gain)** *While the tracking error is decreasing as in Eq. (4.13), the rate of decrease can also potentially decrease with a varying iteration gain  $\rho_k(\omega)$ , e.g.,  $\rho_k(\omega) \rightarrow 0$ . Nevertheless, if the modeling uncertainties decrease with additional data,  $g_k(\omega) \rightarrow g(\omega)$ , then the upper bound on the iteration gain increases  $\rho_k^* \rightarrow 2$  from Eq. (4.15) and the iteration gain in Eq. (4.19) stays bounded away from zero, with  $\rho_k(\omega) \rightarrow \underline{\rho}(\omega)$ . Alternatively, the model updates and changes in the iteration gains could be stopped after a fixed number of iterations, say  $k = k_{pe}$ , for guaranteed output-tracking convergence as in Lemma 1*

#### 4.1.2. Zero iteration gain for large model uncertainty

Guaranteed convergence of the iteratively found input  $u_k(\omega)$  to the desired inverse input  $u_d(\omega)$  at frequency  $\omega$ , i.e.,  $u_k(\omega) \rightarrow u_d(\omega)$  in Eq. (2.6), depends on the model uncertainties  $\Delta_{a,k}(\omega), \Delta_{b,k}(\omega)$  being smaller than the size of the model  $|\hat{a}_k(\omega)|, |\hat{b}_k(\omega)|$ . Therefore, the upper bound on the iteration gain  $\rho_k^*(\omega)$  becomes zero if the uncertainties are large, as stated in the lemma below.

**Lemma 5 (Large uncertainty)** *If the uncertainties  $\Delta_{m,k}(\omega), \Delta_{p,k}(\omega)$  in the real  $\hat{a}_k(\omega)$  and imaginary  $\hat{b}_k(\omega)$  components of the model  $\hat{g}_k$  are larger than the size of the model components, i.e.,*

$$\begin{aligned} \Delta_{a,k}(\omega) &\geq |\hat{a}_k(\omega)| \\ \Delta_{b,k}(\omega) &\geq |\hat{b}_k(\omega)|, \end{aligned} \quad (4.20)$$

then, the upper bound  $\rho_k^*(\omega)$  on the iteration gain in the convergence condition in Eq. (4.14) of Lemma 4 is zero

$$\rho_k^*(\omega) = 0. \quad (4.21)$$

**Proof:** The lemma follows since, the uncertainty  $\Delta_k(\omega) = \hat{g}_{abs,k}(\omega)$ , as defined in Eq. (4.7), satisfies Eq. (4.20) and results in

$$|\hat{g}_k(\omega)|^2 - \hat{g}_{abs,k}(\omega) \cdot \Delta_k(\omega) = 0. \quad (4.22)$$

■

**Remark 8 (Magnitude and phase uncertainties)** *If the uncertainties  $\Delta_{m,k}, \Delta_{p,k}$  in the real  $\hat{a}_k$  and imaginary  $\hat{b}_k$  components of the model  $\hat{g}_k$  are larger than the size of the model components, i.e., as in Eq. (4.20), the magnitude uncertainty  $\Delta_{m,k}(\omega)$  can become infinite in Eq. (4.3) and the cosine of the phase uncertainty  $\cos[\Delta_{p,k}(\omega)]$  cannot be guaranteed to be greater than zero in Eq. (4.4), i.e., the phase uncertainty  $\Delta_{p,k}(\omega)$  cannot be guaranteed to be less than  $90^\circ$ . These are required for guaranteed reduction in the output tracking error with iterations in previous results in,<sup>17),18)</sup> e.g., as in Remark 1.*

**Remark 9 (Similar conditions to robust inversion)** *The requirement, that the uncertainties  $\Delta_{m,k}(\omega), \Delta_{p,k}(\omega)$  in the model components  $\hat{a}_k(\omega), \hat{b}_k(\omega)$  should be smaller than the size of the model components to ensure a nonzero iteration gain as in Lemma 5, is similar to condition that the model uncertainty be smaller than the size of the model for robust invertibility.<sup>40)</sup>*

**Remark 10 (Limited tracking beyond system bandwidth)** *At frequencies  $\omega$  much higher than the system bandwidth  $\omega_{bw}$ , typical system model magnitudes  $|\hat{g}(\omega)|$  tend to become small. Therefore, for a given level of model uncertainties  $\Delta_{a,k}(\omega), \Delta_{b,k}(\omega)$  (due to noise in the measurements), the upper bound  $\rho^*(\omega)$  on the iteration gain in Eq. (4.14) tends to be zero at high frequencies  $\omega \gg \omega_{bw}$ .*

#### 4.2. Iteration law with persistency of excitation

Sufficient richness of the measured system-model and reduced error in the model data can be achieved by augmenting the input  $u_k(\omega)$  with an additional term  $\tilde{u}_k(\omega)$  that ensures persistency of excitation at a frequency  $\omega$  even if the desired output  $y_d(\omega)$ , and therefore the feedforward input  $u_d(\omega)$  in Eq. (2.2) are zero at that frequency  $\omega$ .

##### 4.2.1. Modified iteration

The iterative law in Eq. (2.5) is modified to

$$\begin{aligned} \hat{u}_k(\omega) &= [u_{k-1}(\omega) - \tilde{u}_{k-1}(\omega)] \\ &\quad + \rho_k(\omega) \hat{g}_k^{-1}(\omega) [y_d(\omega) - (y_{m,k-1}(\omega) - \tilde{y}_{m,k-1}(\omega))] \\ u_k(\omega) &= \hat{u}_k(\omega) + \tilde{u}_k(\omega) \end{aligned} \quad (4.23)$$

for  $\omega \leq \omega_c$  and zero elsewhere. Here the additional input  $\tilde{u}_k(\omega)$  is used whenever the un-augmented input  $\hat{u}_k(\omega)$  becomes small to provide persistence of excitation, in the first  $k_{pe} < k^*$  iterations, where  $k^*$  is the total number of iteration steps. For example, the additional input can be chosen to have magnitude  $\tilde{u}(\omega)$  and a random

phase angle  $\phi_k$ , i.e.,

$$\tilde{u}_k(\omega) = \begin{cases} \tilde{u}(\omega)e^{j[\phi_k(\omega)]} & \text{if } |\hat{u}_k(\omega)| < u_{pe}(\omega) \\ & \text{and } k \leq k_{pe} \\ 0 & \text{otherwise} \end{cases} \quad (4.24)$$

$$\phi_k(\omega) \sim \mathcal{N}(0, \pi^2),$$

where  $u_{pe}(\omega)$  is selected to designate when the input is considered to be small. Note that the additional input  $\tilde{u}_{k-1}(\omega)$  from the previous iteration is removed when updating the iterative input  $u_k(\omega)$  in Eq. (4.23). Moreover, the estimated effect  $\tilde{y}_{m,k-1}(\omega)$  of the additional input  $\tilde{u}_{k-1}(\omega)$  on the measured output  $y_{m,k-1}(\omega)$  is removed before computing the updated input  $u_{k-1}(\omega)$  in Eq. (4.23), with

$$\tilde{y}_{m,k-1}(\omega) = \hat{g}_k(\omega)\tilde{u}_{k-1}(\omega), \quad (4.25)$$

where the measured output  $y_{m,k-1}(\omega)$  includes potential measurement noise  $n_{y,k-1}(\omega)$ , i.e.,

$$\begin{aligned} y_{m,k-1}(\omega) &= y_{k-1}(\omega) + n_{y,k-1}(\omega) \\ n_{y,k-1}(\omega) &\sim \mathcal{N}(0, \sigma_{y,a}^2(\omega)) + j\mathcal{N}(0, \sigma_{y,b}^2(\omega)). \end{aligned} \quad (4.26)$$

**Remark 11 (Residual modeling error)** *If the model is not learned well, then the correction of the input augmentation in Eq. (4.25) will not be exact. The error due to in-exact compensation of the input augmentation can be corrected iteratively if the total number of iterations  $k^*$  is sufficiently larger than the initial iterations  $k \leq k_{pe}$  when the input is augmented, i.e.,  $k_{pe} \ll k^*$ .*

If an initial model  $\hat{g}_1$  is available, then the output-tracking input can be estimated as

$$\hat{u}_1(\omega) = \hat{g}_1^{-1}(\omega)[y_d(\omega)] \quad \forall \omega \leq \omega_c, \quad (4.27)$$

and zero elsewhere. If an initial model is not available, then  $\hat{u}_1(\omega) = 0$  for all frequency  $\omega$ . The input  $\hat{u}_1$  can be augmented to improve model estimation, as

$$u_1(\omega) = \hat{u}_1(\omega) + \tilde{u}_1(\omega), \quad \forall \omega \leq \omega_c, \quad (4.28)$$

and zero elsewhere, with  $\tilde{u}_1$  as in Eq. (4.24).

**Remark 12 (Initial input using learned model)** *The final model  $\hat{g}_{k^*}$  at the final iteration step  $k^*$  can be used as the initial model  $\hat{g}_1$  in Eq. (4.28) during the iterative output tracking of a new output trajectory  $y_{d,2}$ .*

#### 4.2.2. Additional data for model

New data  $m_{a,k}(\omega), m_{b,k}(\omega)$  to estimate the system model  $\hat{g}_k(\omega)$  can be computed at frequencies  $\omega$  where the input  $u_{k-1}(\omega)$  is sufficiently large, say

$$|u_{k-1}(\omega)| \geq u_{ok}(\omega) > 0, \quad (4.29)$$

as in Eq. (3.2)

$$\begin{aligned}
 m_{g,k}(\omega) &= m_{a,k}(\omega) + jm_{b,k}(\omega) = \frac{y_{m,k-1}(\omega)}{u_{k-1}(\omega)} \\
 &= \frac{y_{k-1}(\omega) + n_{y,k-1}(\omega)}{u_{k-1}(\omega)} \\
 &= g(\omega) + \frac{n_{y,k-1}(\omega)}{u_{k-1}(\omega)}
 \end{aligned} \tag{4.30}$$

with the measurement noise  $n_{y,k-1}(\omega)$  as in Eq. (4.26).

**Remark 13 (Reduction of measurement-noise effect)** *The magnitude  $\tilde{u}(\omega)$  of the additional input at frequency  $\omega$  in Eq. (4.24) should be selected to be much larger than the expected standard deviations  $\sigma_{y,a}(\omega), \sigma_{y,b}(\omega)$  of the measurement noise  $n_{y,k-1}(\omega)$  in Eq. (4.26) to increase the input  $u_{k-1}(\omega)$  and reduce the impact of measurement noise  $n_{y,k-1}(\omega)$  on the model data  $m_{a,k}(\omega), m_{b,k}(\omega)$  computed using Eq. (4.30).*

#### 4.2.3. Averaged model data

The computational cost of GPR can become prohibitive as the number of model data increases, e.g., with increasing iteration steps. Therefore, in the proposed algorithm, the model data is first averaged at each frequency  $\omega$  where  $N_{\omega,k} > 0$  data points are available similar to,<sup>32)</sup> as discussed in Remark 3,

$$\bar{m}_{g,k}(\omega) = \bar{m}_{a,k}(\omega) + j\bar{m}_{b,k}(\omega) = \frac{1}{N_{\omega,k}} \sum_{i=1}^k m_{g,i}(\omega) \tag{4.31}$$

and  $m_{g,i}$  is defined in Eq. (4.30). Then, the averaged data  $\bar{m}_{g,k}(\omega)$  is used to refine the model  $\hat{g}_k$  using the GPR in Eq. (3.8) before computing the updated input  $u_k(\omega)$  in Eq. (4.23). With sufficient number of data points  $N_{\omega,k}$  at a frequency  $\omega \leq \omega_c$  due to persistency of excitation, the estimated mean  $\bar{m}_{g,k}(\omega)$  of the model data tends to the system  $g(\omega)$ , e.g., as the number of iterations increase.

#### 4.3. Proposed algorithm

The proposed iterative machine learning (IML) algorithm using GPR is described below in Fig. 1

## §5. Simulation results and discussion

Simulation results are presented on convergence of the trajectory tracking input with the proposed IML approach. The impact of using the additional input for persistency of excitation on the learning of the model is illustrated. Additionally, the advantage of using the model learning to improve tracking of a new trajectory is illustrated.

---

**Algorithm** Iterative machine learning (IML)

---

**Require:** the desired output  $y_d(\omega)$ , iteration gain scaling  $\underline{\rho}(\omega)$  in Eq. (45), confidence interval (e.g., 95%) for model uncertainty, number of iterations  $k^*$ , number of iterations with input augmentation  $k_{pe}$ , and cut-off frequency  $\omega_c$ .

- 1: **Initialize:**
- 2: Compute input  $u_1(\cdot)$ , Eq. (54)
- 3: Apply input  $u_1(\cdot)$  to system and measure output  $y_{m,1}(\cdot)$
- 4: **Loop:**
- 5: **for**  $k = 2$  to  $k^*$  **do**
- 6:   Estimate new data  $m_{g,k}$  and its average  $\bar{m}_{g,k}(\omega)$  from Eqs. (56), (57)
- 7:   Compute new model  $\hat{g}_k(\omega)$  and uncertainty  $\Delta_{a,k}, \Delta_{b,k}$ , Eqs. (25) and (26)
- 8:   Compute iteration gain  $\rho_k(\omega)$  as in Eq. (45)
- 9:   Compute input  $\hat{u}_k(\omega)$  as in Eq. (49)
- 10:   **if**  $k \leq k_{pe}$  **then**
- 11:     Compute input augmentation  $\tilde{u}_k(\omega)$  as in Eq. (50)
- 12:   **end if**
- 13:   Compute input  $u_k(\omega)$  as in Eq. (49)
- 14:   Apply input  $u_k(\cdot)$  to system and measure output  $y_{m,k}(\cdot)$
- 15: **end for**

---

Fig. 1. Iterative machine learning (IML)

### 5.1. Example system

Consider a non-minimum phase example system  $g(s)$  in Eq. (2.1) of the form

$$g(s) = \frac{-\frac{\omega_{p1}^2 \omega_{p2}^2}{\omega_z^2} [(s - \omega_z)(s + \omega_z)]}{\left[ s^2 + 2\zeta_{p1}\omega_{p1}s + \omega_{p1}^2 \right] \left[ s^2 + 2\zeta_{p2}\omega_{p2}s + \omega_{p2}^2 \right]}. \quad (5.1)$$

Let the larger pole frequency  $\omega_{p2} = 6\pi$  rads/s be three times the smaller pole frequency  $\omega_{p1} = 2\pi$  rads/s, with the zero  $\omega_z = 4\pi$  rads/s interlaced between the poles and the damping ratios as  $\zeta_{p1} = \zeta_{p2} = 1/\sqrt{2}$ . Note that the system has relative degree two (since the number of poles is two more than the number of zeros) and hence the desired output  $y_d$  needs to be twice differentiable to enable exact tracking. The frequency response of the example system (with the above parameter values)

is shown in Fig. 2. The system bandwidth is about one Hz,  $\omega_{bw} \approx 1$  Hz. Since output tracking is not usually anticipated much beyond the system bandwidth (see Remark 10), the cutoff frequency  $\omega_c$  for computing the model  $\hat{g}$  and the iteration input  $u_k$  is selected as  $\omega_c = 5$  Hz — about five times the system bandwidth  $\omega_{bw}$ .

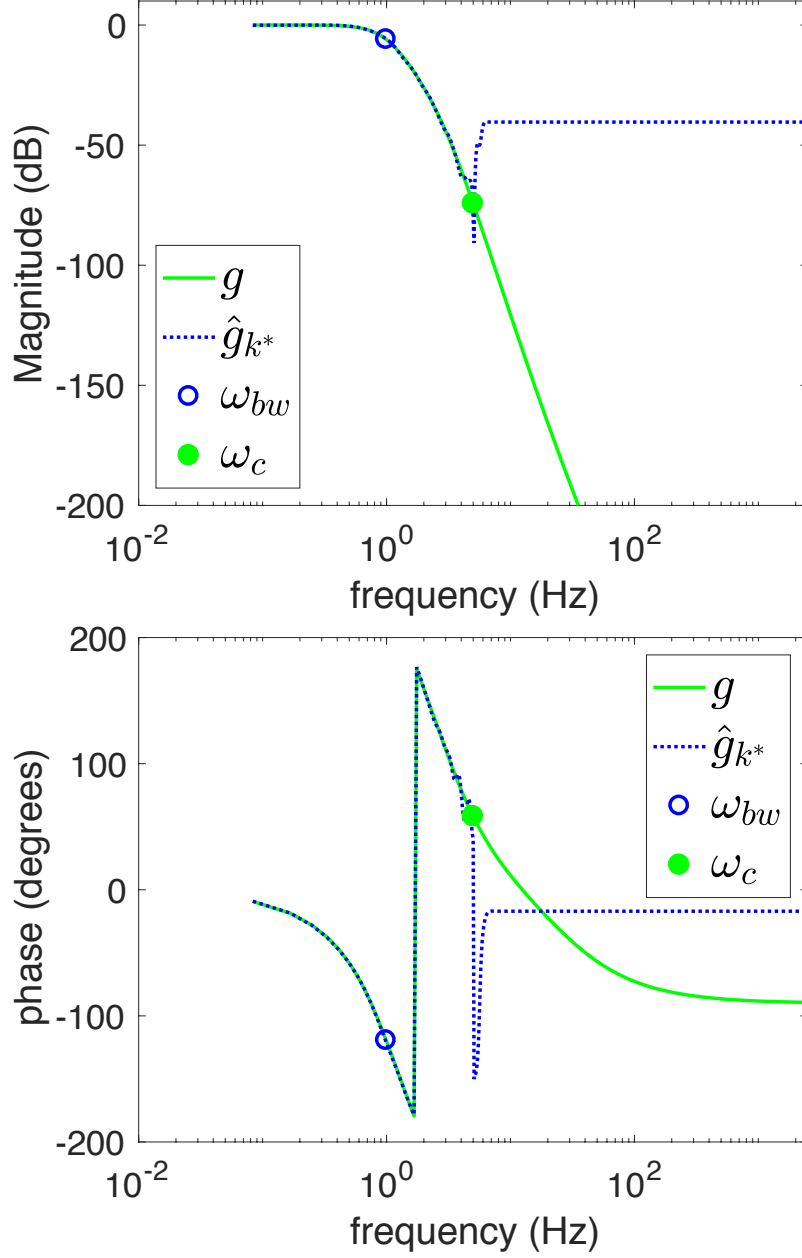


Fig. 2. Frequency response of the example system  $g$  in Eq. (5.1): (top) magnitude; and (bottom) phase in degrees. The learned model  $\hat{g}$  is also shown.



### 5.2. Desired trajectory

Consider a twice-differentiable, desired trajectory  $y_d$ , with zero initial position  $y_d(0) = 0$  and velocity  $\dot{y}_d(0) = 0$ , and specified by its second time derivative  $\ddot{y}_d$  as

$$\ddot{y}_d(t) = \begin{cases} \sum_{n=1}^{n=N} \sin[n\omega_*(t - t_0)] & \text{if } t_0 < t < t_1 \\ -\sum_{n=1}^{n=N} \sin[n\omega_*(t - t_1)] & \text{if } t_1 < t < t_2 \\ 0 & \text{otherwise,} \end{cases} \quad (5.2)$$

where main frequency component  $\omega_* = 0.5$  Hz that is about half the system bandwidth  $\omega_{bw}$ , the number of harmonics  $N$  in the desired output  $y_d$  was one, i.e.,  $N = 1$  and  $t_0 = 4$ s,  $t_1 = 6$ s,  $t_2 = 8$ s. A plot of the desired output is shown in Fig. 3. The padding around the middle section  $t \in [4, 8]$  is added to enable noncausal solutions. The computations are performed in the time domain with a sampling time of 0.2ms and in the discrete-frequency domain with the fast Fourier transform (FFT) using MATLAB.

### 5.3. Impact of additional input for persistency of excitation

The error in the model data from Eq. (4.30) can be large in the presence of noisy output measurements  $y_{m,k}(\omega)$ . To illustrate, the error in the model data are compared below, with and without the persistency of excitation, when the input  $u(\omega)$  is the exact tracking input  $u_d(\omega)$  from Eq. (2.2), for frequency  $\omega$  less than the cutoff frequency  $\omega_c$ .

The standard deviation  $\sigma_{n_y}(\omega)$  of the output noise  $n_{y,d}(\omega)$  in the simulations is, similar to (4.26),

$$\sigma_{n_y} = \sigma_{y,a}(\omega) = \sigma_{y,b}(\omega) = \frac{1}{10^4} \max_{\omega \leq \omega_c} [y_d(\omega)] \quad (5.3)$$

and the resulting input and output are shown in Fig. 3.

#### 5.3.1. Effect of measurement noise

The noisy measured output  $y_m$ , as in Eq. (4.26),

$$y_{m,d}(\omega) = y_d(\omega) + n_{y,d}(\omega) = g(\omega)u_d(\omega) + n_{y,d}(\omega) \quad (5.4)$$

shown in Fig. 3 is close to the desired output  $y_d$  with the inverse feedforward input  $u = u_d$ . With this measured output  $y_{m,d}$ , the model data can be computed from Eq. (4.30) as

$$m_g(\omega) = m_a(\omega) + jm_b(\omega) = \frac{y_{m,d}(\omega)}{u_d(\omega)}. \quad (5.5)$$

The error  $e_a, e_b$  in the model data components  $m_a, m_b$  increases with frequency since the output  $y_d$  tends to become small compared to the output noise  $\sigma_{n_y}$ , e.g., at

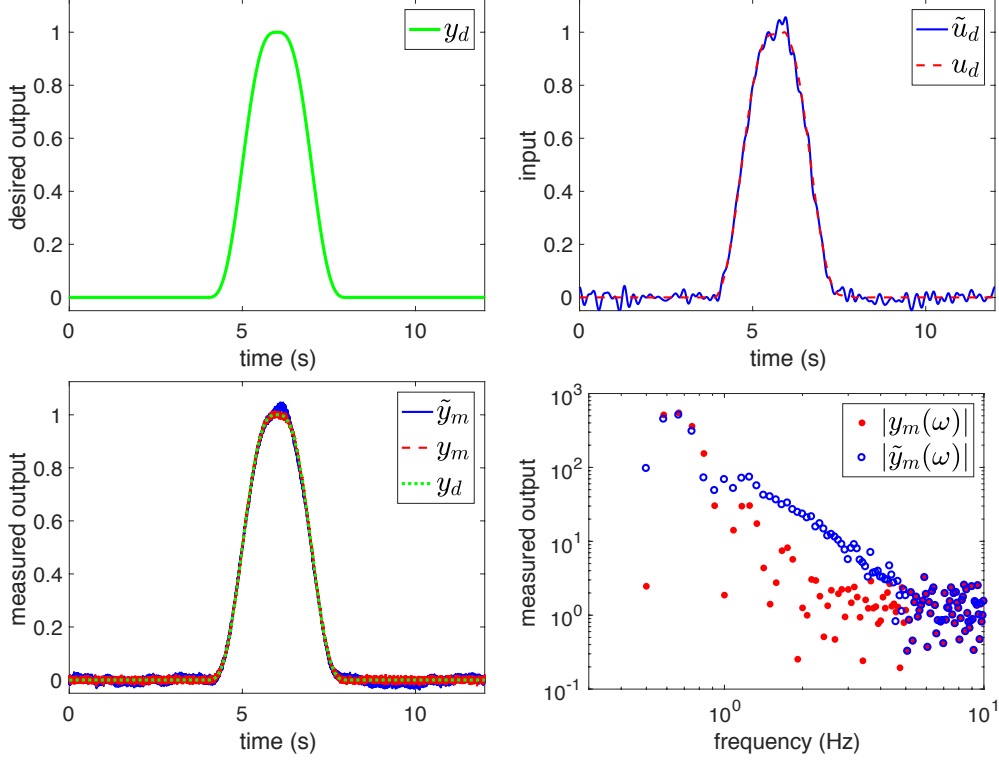


Fig. 3. Comparison of input and output with and without persistency of excitation. Desired output (top left). Input  $u$  (top right) : (case  $i$ ) the inverse input  $u_d$  from Eq. (2.2); and (case  $ii$ ) the augmented input  $\tilde{u}_d$  from Eq. (5.7) with persistency of excitation. Resulting measured output  $y_m$  from Eq. (5.4) and  $\tilde{y}_m$  from Eq. (5.10) are shown in time domain (bottom left) and frequency domain (bottom right). The plot of the measured outputs  $y_m, \tilde{y}_m$  (bottom left) overlaps the plot of the desired output  $y_d$ .

frequencies,  $\omega > 1$  Hz, as seen in Fig. 4 (bottom two plots). The maximum model-data error  $e_{a,max}, e_{b,max}$  in the real and imaginary components of the model data are

$$\begin{aligned} e_{a,max} &= \max_{\omega \leq \omega_c} [e_a(\omega) = |m_a(\omega) - a(\omega)|] = 1.58 \times 10^5 \\ e_{b,max} &= \max_{\omega \leq \omega_c} [e_b(\omega) = |m_b(\omega) - b(\omega)|] = 2.24 \times 10^5. \end{aligned} \quad (5.6)$$

### 5.3.2. Reduction of noise effect with persistency of excitation

To evaluate the advantage of persistency of excitation, consider the augmentation of the inverse input  $u_d(\omega)$  to  $\tilde{u}_d(\omega)$ , as in Eq (4.24), at frequency  $\omega$  less than the cutoff frequency  $\omega_c$ ,

$$\tilde{u}_d(\omega) = u_d(\omega) + \tilde{u}(\omega)e^{j[\phi_d(\omega)]} \quad (5.7)$$

whenever the input  $u_d(\omega)$  was small,

$$|u_d(\omega)| < u_{pe}(\omega) = 10\sigma_{n_y} \quad (5.8)$$

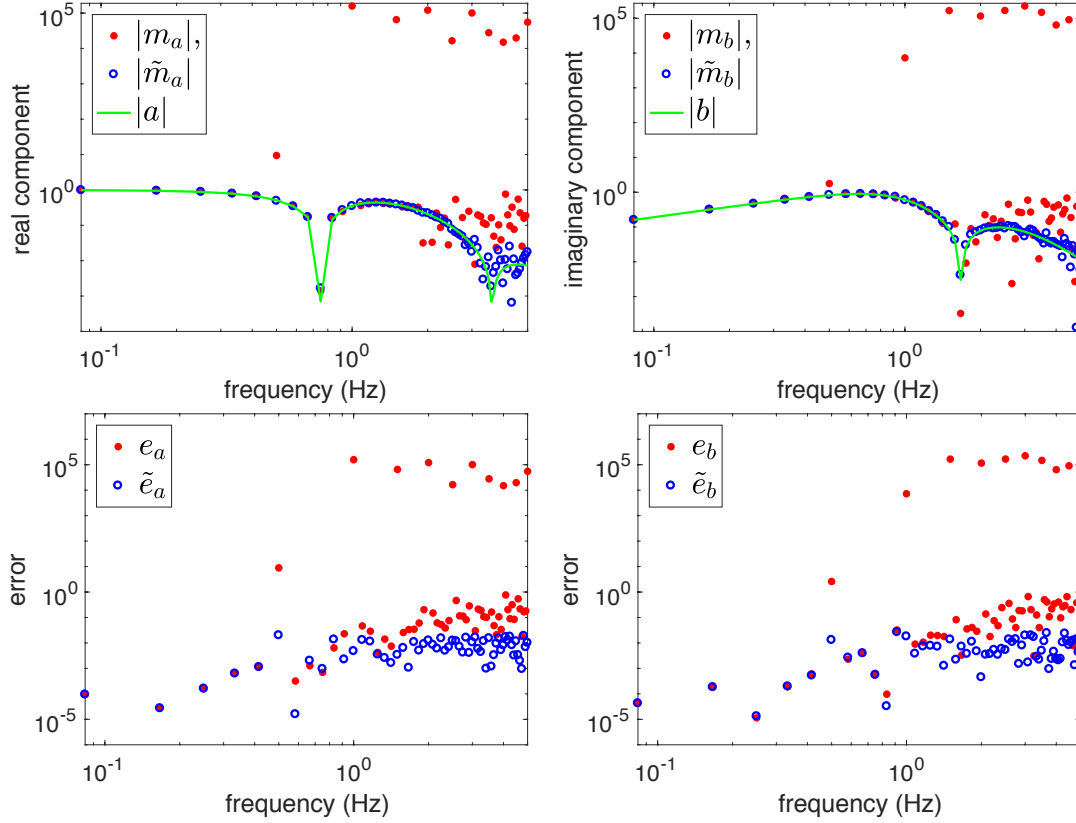


Fig. 4. Impact of persistency of excitation on acquired model data. Top two plots compare the absolute values of the real and imaginary components  $a, b$  for the example system  $g$  in Eq. (5.1) with the components of the data  $m_a, m_b$  from Eq. (5.5) and  $\tilde{m}_a, \tilde{m}_b$  from Eq. (5.11). The bottom two plots compare the model-data error for the two cases,  $e_a, e_b$  in Eq. (5.6) and  $\tilde{e}_a, \tilde{e}_b$  in Eq. (5.12).

with the additional input magnitude and phase given by

$$\tilde{u}(\omega) = 100\sigma_{n_y}, \quad \phi_d(\omega) \sim \mathcal{N}(0, \pi^2), \quad \omega \leq \omega_c. \quad (5.9)$$

The resulting augmented input  $\tilde{u}_d$  and the associated measured output  $\tilde{y}_{m,d}$  with the same noise  $n_{y,d}(\omega)$  as the unaugmented case in Eq. (5.4)

$$\tilde{y}_{m,d}(\omega) = g(\omega)\tilde{u}_d(\omega) + n_{y,d}(\omega) \quad (5.10)$$

are shown in Fig. 3. The effect of measurement noise in the model estimates from Eq. (4.30)

$$\tilde{m}_g(\omega) = \tilde{m}_a(\omega) + j\tilde{m}_b(\omega) = \frac{\tilde{y}_{m,d}(\omega)}{\tilde{u}_d(\omega)}. \quad (5.11)$$

are shown in Fig. 4 (top two plots), which compares the real and imaginary components  $a(\omega), b(\omega)$  of the example system  $g(\omega)$  in Eq. (5.1) with the real and imaginary components of the model data estimated for the two cases: (case i) the inverse input

$u_d(\omega)$  from Eq. (2.2) and (case ii) the input  $\tilde{u}_d(\omega)$  with persistency of excitation from Eq. (5.7). Note that the model data with the persistency of excitation input  $\tilde{u}_d(\omega)$  tend to be closer to the actual system  $g(\omega)$  when compared to the case without the persistency of excitation as seen in Fig. 4. The maximum model-data error  $\tilde{e}_{a,max}, \tilde{e}_{b,max}$  in the real and imaginary components, with the persistency of excitation input, over all frequencies  $\omega \leq \omega_c$ , are

$$\begin{aligned}\tilde{e}_{a,max} &= \max_{\omega \leq \omega_c} [\tilde{e}_a(\omega) = |\tilde{m}_a(\omega) - a(\omega)|] = 0.020, \\ \tilde{e}_{b,max} &= \max_{\omega \leq \omega_c} [\tilde{e}_b(\omega) = |\tilde{m}_b(\omega) - b(\omega)|] = 0.027.\end{aligned}\tag{5.12}$$

Thus, the addition the persistency of excitation input (with relatively-small change on the output, typically at high-frequency, as seen in Fig. 3) can lead to substantially smaller model error  $\tilde{e}_{a,max}, \tilde{e}_{b,max}$  in Eq. (5.12) than the model error  $e_{a,max}, e_{b,max}$  in Eq. (5.6) without the persistency of excitation — several orders of magnitude less error, as seen in Fig. 4, (bottom two plots).

#### 5.4. Convergence with iterations

Convergence with the proposed IML approach is discussed below with the initial input  $u_1$  chosen with the augmented input in Eq. (4.28) and without prior knowledge of the model, i.e.,  $\hat{u}_1(\omega) = 0$  in Eq. (4.28).

##### 5.4.1. GPR results

All available data  $m_{g,i}(\omega)$  with  $i = 1, 2, \dots, k$  is used to predict the model  $\hat{g}_k$  using GPR with *fitrgp* and *predict* functions in MATLAB, where the covariance function is the squared exponential kernel, and the hyperparameters are optimized using the data. With the total number of iterations selected as  $k^* = 5$  and model augmentation during the first three iterations  $k_{pe} = 3$  in Eq. (4.24), the estimated the model data  $\hat{g}_{k^*}(\omega)$  is close to the system  $g(\omega)$  as seen in Fig. 5. The 95% confidence intervals  $\Delta_{a,k}, \Delta_{b,k}$  (e.g., shown in Fig. 5 for the final iteration  $k = k^*$ ) are used as the expected bounds on the model uncertainty as in Eq. (4.1).

##### 5.4.2. Selection of iteration gain

The iteration gain  $\rho_k(\omega)$  was chosen as in Eq. (4.19) with  $\rho(\omega) = 0.9$ . The iteration gain  $\rho_{k^*}(\omega)$  and its upper bound  $\rho_{k^*}^*(\omega)$  from Eq. (4.15) at the final iteration step  $k^*$  are shown in Fig. 6, which tends to zero at high frequency when the model size becomes small compared to the measurement noise, as in Remark 10.

**Remark 14 (Cutoff frequency)** While the iteration gain  $\rho(\omega)$  acts as a low-pass filter and tends to zero at high frequency  $\omega$  when the system  $g(\omega)$  tends to zero, e.g., see Fig. 6, the cutoff frequency  $\omega_c$  (beyond which the iteration gain  $\rho(\omega)$  is set to zero) is used to avoid potential divergence at high-frequencies where the model size and the allowable uncertainty are expected to be small.

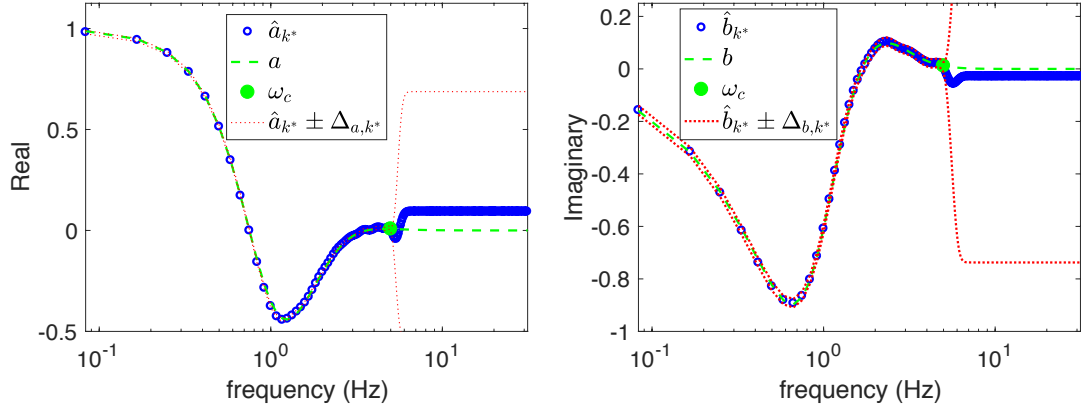


Fig. 5. Results of the Gaussian process regression. The fit from the GPR  $\hat{a}_{k^*}(\omega), \hat{b}_{k^*}(\omega)$  is close to the system  $a(\omega), b(\omega)$  — the fit overlaps the model upto the cutoff frequency  $\omega_c$  due to the augmented input and deviations become noticeable beyond  $\omega_c$ . The 95% confidence interval (dotted lines) is small till the cutoff frequency  $\omega_c$ .

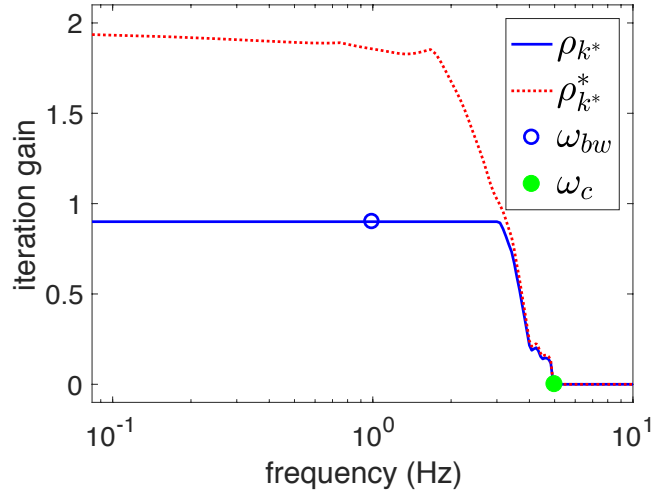


Fig. 6. The maximum possible iteration gain  $\rho_{k^*}^*(\omega)$  from Eq. (4.15) and the selected iteration gain  $\rho_{k^*}$  used to compute the final input  $u_{k^*}(\omega)$ .

#### 5.4.3. Input and model convergence

Convergence is achieved as the iteration steps  $k$  increase, i.e., the model converges to the system response  $\hat{g}_k(\omega) \rightarrow g(\omega)$  and the output converges to the desired output  $y_k(\cdot) \rightarrow y_d(\cdot)$ , as seen in Fig. 7. The modeling error  $e_{g,k}$  given by

$$e_{g,k} = \frac{\max_{\omega \leq \omega_c} |\hat{g}_k(\omega) - g(\omega)|}{\max_{\omega \leq \omega_c} |g(\omega)|} \times 100, \quad (5.13)$$

reduced to  $e_{g,k^*} = 2.0\%$  at the final iteration  $k = k^*$ . The output tracking error  $e_{y,k}$  given by

$$e_{y,k} = \frac{\max_t |y_k(t) - y_d(t)|}{\max_t |y_d(t)|} \times 100, \quad (5.14)$$

reduced to  $e_{y,k^*} = 2.48\%$  at the final iteration  $k = k^*$ , which is the same as the noise level, i.e.,

$$\frac{\max_t |n_{y,k^*}(t)|}{\max_t |y_d(t)|} \times 100 = 2.48. \quad (5.15)$$

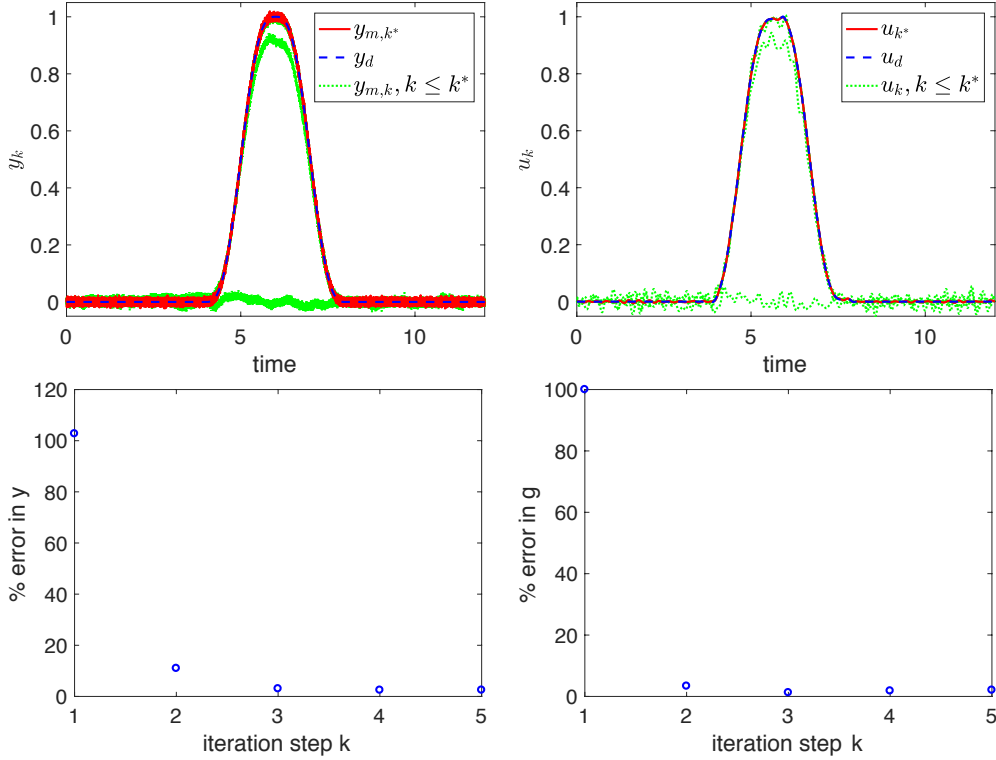


Fig. 7. (Top left) Convergence of the measured output  $y_{m,k}$  to the desired output  $y_d$ . (Top right) Convergence of the input  $u_k$  to the inverse feedforward input  $u_d$  in Eq. (2.2). (Bottom left) Output tracking error  $e_{y,k}$  from Eq. (5.14). (Bottom right) Modeling error  $e_{g,k}$  from Eq. (5.13).

### 5.5. Impact of model learning for new output

The impact of model learning for tracking was evaluated for a new desired output  $y_{d,2}$  with a higher main frequency  $\omega_* = 2/3$  Hz compared to the previous desired output, and five harmonics  $N = 5$  in Eq. (5.2). A plot of the desired output is shown in Fig. 8. Even with this increase in the amplitude of the desired output at higher frequencies, the initial input  $u_1$

$$u_1(\omega) = \hat{g}_1^{-1}(\omega) [y_d(\omega)] \quad \forall \omega \leq \omega_c, \quad (5.16)$$

found with the initial model  $\hat{g}_1$  defined as the final model  $\hat{g}_{k^*}$  from the previous iterations, led to good initial tracking as seen in Fig. 8. (The input augmentation was not added to this initial input to clarify the impact of using the model from the previous iterations.) The input augmentation was used for iteration steps  $k = 2, k = 3$  and the total number of iterations was  $k^* = 5$ . The output tracking error (with five iterations) is small — the measured outputs  $y_{m,k}$  for all five iterations tend to overlap the desired output  $y_{d,2}$  in Fig. 8. The initial tracking error  $e_{y,1}$  for the new output trajectory  $y_{d,2}$  at the first iteration step was 2.97%, which is close to the noise level of 2.12% — the tracking error  $e_{y,5}$  at the end of five iterations for the second output  $y_{d,2}$  is 2.15%. Thus, as expected, the learning of the model during the iteration process with the previous desired output trajectory  $y_d$  leads to small initial error in the iterations when learning the new desired output  $y_{d,2}$ . In this sense, the propose IML improves the performance and portability of the frequency-domain iterative control.

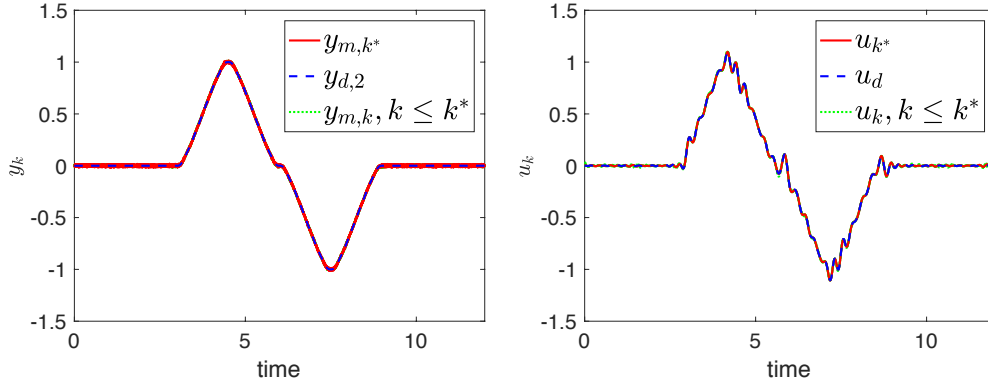


Fig. 8. Iteration results for new desired output  $y_{d,2}$  for five iterations,  $k^* = 5$ . (Left) All five measured outputs  $y_{m,k}$  ( $k = 1, \dots, 5$ ) overlap the desired output  $y_{d,2}$ . (Right) All five computed inputs  $u_k$  are close to the inverse feedforward input  $u_d$  in Eq. (2.2) and have substantial higher frequency content when compared to the input for the initial output trajectory  $y_d$  in Fig. 7.

## §6. Conclusions

This article proposed a frequency-domain iterative machine learning (IML) approach to simultaneously update the system model while learning the inverse input. Additionally, inputs with persistency of excitation were proposed to promote learning of the model by generating data at frequencies outside the main frequency content of the specified desired output. The method was applied to a simulation example, and results show the model learning can substantially reduce the initial error for other desired output trajectories.

## §7. Acknowledgment

This work was partially supported by NSF grant CMII 1536306 and the Helen R. Whiteley Center at Friday Harbor.

## References

- 1) S. Arimoto, S. Kawamura, and F. Miyazaki. Bettering operation of robots by learning. *J. of Robotic Systems*, 1(2):123–140, March 1984.
- 2) J. J. Craig. Adaptive control of manipulators through repeated trials. *Proceedings of the American Control Conference*, page 15661573, 1984.
- 3) C. G. Atkeson and J. McIntyre. Robot trajectory learning through practice. In *IEEE Int. Conf. on Robotics and Automation*, pages 1737–1742, 1986.
- 4) Yang Li and John Bechhoefer. Feedforward control of a piezoelectric flexure stage for AFM. *Proceedings of American Control Conference, Seattle, WA*, pages 2703–2709, June 11–13, 2008.
- 5) G. M. Clayton and S. Devasia. Iterative image-based modeling and control for higher scanning probe microscope performance. *Review of Scientific Instruments*, 78(8):Article No. 083704 (pp. 1–12), August 29, 2007.
- 6) G. Schitter, R. W. Stark, and A. Stemmer. Fast contact-mode atomic force microscopy on biological specimen by model-based control. *Ultramicroscopy*, 100(3-4):253–257, Aug, 2004.
- 7) K.-S. Kim and Q. Zou. Model-less inversion-based iterative control for output tracking: Piezo actuator example. *Proceedings of American Control Conference, Seattle, WA*, pages 2170–2175, June 11–13, 2008.
- 8) K. K. Leang and S. Devasia. Design of hysteresis-compensating iterative learning control: Application to atomic force microscopes. *Mechatronics*, 16(3-4):141–158, April-May, 2006.
- 9) Ying Wu and Qingze Zou. Iterative control approach to compensate for both the hysteresis and the dynamics effects of piezo actuators. *IEEE Transactions on Control Systems Technology*, 15(5):936–944, September, 2007.
- 10) M. P. Chaffe and L. Y. Pao. Iterative learning control for near-field scanning optical microscope applications. In *2011 IEEE International Conference on Control Applications (CCA)*, pages 1075–1080, Sept 2011.
- 11) Stan H. van der Meulen, Rob L. Tousain, and Okko H. Bosgra. Fixed Structure Feedforward Controller Design Exploiting Iterative Trials: Application to a Wafer Stage and a Desktop Printer. *ASME. J. Dyn. Sys., Meas., Control*, 130(7):051006–051006 1–16, Sept 2008.
- 12) Joost Bolder and Tom Oomen. Rational Basis Functions in Iterative Learning Control- With Experimental Verification on a Motion System. *IEEE Transactions on Controls Systems Technology*, 23(2):722–729, MAR 2015.
- 13) S. Mishra and M. Tomizuka. Segmented iterative learning control for precision positioning of waferstages. In *2007 IEEE/ASME international conference on advanced intelligent mechatronics AIM*, pages 1–6, Sept 2007.
- 14) D. Gorinevsky, D. E. Torfs, and A. A. Goldenberg. Learning approximation of feedforward control dependence on the task parameters with application to direct-drive manipulator tracking. *IEEE Transactions on Robotics and Automation*, 13(4):567–581, Aug 1997.
- 15) K. L. Moore. *Iterative Learning Control for Deterministic Systems*. Springer-Verlag, London, U.K., 1993.
- 16) S. Tien, Qingze Zou, and S. Devasia. Control of dynamics-coupling effects in piezo-actuator for high-speed afm operation. In *Proceedings of the 2004 American Control Conference*, volume 4, pages 3116–3121 vol.4, June 2004.
- 17) J. Hatonen, T.J. Harte, D.H. Owens, J. Ratcliffe, P. Lewin, and E. Rogers. Iterative learning control - what is it all about? *{IFAC} Proceedings Volumes*, 37(12):547 – 553, 2004. *{IFAC} Workshop on Adaptation and Learning in Control and Signal Processing (AL-COSP 04)* and *{IFAC} Workshop on Periodic Control Systems (PSYCO 04)*, Yokohama, Japan, 30 August - 1 September, 2004.
- 18) S. Tien, Q. Zou, and S. Devasia. Iterative control of dynamics-coupling-caused errors in



- piezoscanners during high-speed AFM operation. *IEEE Transactions on Control Systems Technology*, 13(6):921–931, Nov 2005.
- 19) H.-S. Ahn, Y. Q. Chen, and K. L. Moore. Iterative learning control: Brief survey and categorization. *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, 37(6):1099–1121, 2007.
- 20) J. Ghosh and B. Paden. Iterative learning control for nonlinear nonminimum phase plants. *ASME Journal of Dynamic Systems, Measurement, and Control*, 123:21–20, March, 2001.
- 21) D.A. Bristow and A.G. Alleyne. Monotonic convergence of iterative learning control for uncertain systems using a time-varying filter. *IEEE Transactions on Automatic Control*, 53(2):582 – 585, March, 2008.
- 22) J. Ghosh and B. Paden. A pseudo-inverse based iterative learning control. *IEEE Trans. on Automatic Control*, 47(5):831–837, May, 2002.
- 23) C. Peng, L. Sun, W. Zhang, and M. Tomizuka. Optimization-based constrained iterative learning control with application to building temperature control systems. In *2016 IEEE International Conference on Advanced Intelligent Mechatronics (AIM)*, pages 709–715, July 2016.
- 24) Bing Chu, David H. Owens, and Christopher T. Freeman. Iterative Learning Control With Predictive Trial Information: Convergence, Robustness, and Experimental Verification. *IEEE Transactions on Controls Systems Technology*, 24(3):1101–1108, MAY 2016.
- 25) M. Norrlof. An adaptive iterative learning control algorithm with experiments on an industrial robot. *IEEE Transactions on Robotics and Automation*, 18(2):245–251, Apr 2002.
- 26) Mark Butcher, Alireza Karimi, and Roland Longchamp. Iterative learning control based on stochastic approximation. In *17th IFAC World Congress, Seoul, Korea*, July 6-11 2008.
- 27) X. Gao and S. Mishra. An iterative learning control algorithm for portability between trajectories. In *2014 American Control Conference, Portland, OR*, pages 3808–3813, June 4-6 June, 2014.
- 28) Joost Bolder, Tom Oomen, Sjirk Koekebakker, and Maarten Steinbuch. Using iterative learning control with basis functions to compensate medium deformation in a wide-format inkjet printer. *MECHATRONICS*, 24(8):944–953, DEC 2014.
- 29) J. van Zundert, J. Bolder, and T. Oomen. Iterative learning control for varying tasks: Achieving optimality for rational basis functions. In *2015 American Control Conference (ACC)*, pages 3570–3575, July 2015.
- 30) C. T. Freeman, P. L. Lewin, E. Rogers, D. H. Owens, and J. J. Hatonen. Discrete fourier transform based iterative learning control design for linear plants with experimental verification. *ASME Journal of Dynamic Systems, Measurement, and Control*, 131(3):031006–1 – 031006–10, May, 2009.
- 31) K.-S. Kim and Q. Zou. A modeling-free inversion-based iterative feedforward control for precision output tracking of linear time-invariant systems. *IEEE/ASME Transactions on Mechatronics*, 18(6):1767–1777, Dec. 2013.
- 32) W. Wang and Q. Zou. A modeling-free differential-inversion-based iterative control approach to simultaneous hysteresis-dynamics compensation: high-speed large-range motion tracking example. *American Control Conference, Palmer House Hilton, Chicago, IL, USA*, July 1-3 2015.
- 33) C. E. Rasmussen and C. K. I. Williams. *Gaussian Processes for Machine Learning*. The MIT Press, Cambridge, MA, 2006.
- 34) K. J. Astrom and B. Wittenman. *Adaptive Control*. Addison Wesley, New York, 1989.
- 35) Gianluigi Pillonetto, Francesco Dinuzzo, Tianshi Chen, Giuseppe De Nicolao, and Lennart Ljung. Kernel methods in system identification, machine learning and function estimation: A survey. *Automatica*, 50(3):657 – 682, 2014.
- 36) R. Boloix-Tortosa, F. J. Payn-Somet, E. Arias de Reyna, and J. J. Murillo-Fuentes. Complex kernels for proper complex-valued signals: A review. In *2015 23rd European Signal Processing Conference (EUSIPCO)*, pages 2371–2375, Aug 2015.
- 37) P. M. Sammons, D. A. Bristow, and R. G. Landers. Iterative learning control of bead morphology in laser metal deposition processes. In *2013 American Control Conference*, pages 5942–5947, June 2013.
- 38) David J. Hoelzle and Kira L. Barton. On Spatial Iterative Learning Control via 2-D Convolution: Stability Analysis and Computational Efficiency. *IEEE Transactions on*

- Controls Systems Technology*, 24(4):1504–1512, JUL 2016.
- 39) Nanjun Liu and Andrew Alleyne. Iterative Learning Identification for Linear Time-Varying Systems. *IEEE Transactions on Controls Systems Technology*, 24(1):310–317, JAN 2016.
  - 40) S. Devasia. Should model-based inverse inputs be used as feedforward under plant uncertainty? *IEEE Transactions on Automatic Control*, 47(11):1865–1871, Nov, 2002.
  - 41) F. Perez-Cruz, J. J. Murillo-Fuentes, and S. Caro. Nonlinear channel equalization with gaussian processes for regression. *IEEE Transactions on Signal Processing*, 56(10):5283–5286, Oct 2008.
  - 42) John Bechhoefer. Kramers-Kronig, Bode, and the meaning of zero. *American Journal of Physics*, 79(10):1053–1059, 2011.