ASME Letters in Dynamic Systems and Control
Online journal at:
https://asmedigitalcollection.asme.org/lettersdynsys

A Recursive System Identification With Non-Uniform Temporal Feedback Under Coprime Collaborative Sensing[1]

Jinhua Ouyang
Mechatronics, Automation, and Control Systems Laboratory,
Department of Mechanical Engineering,
University of Washington,
Seattle, WA 98195
e-mail: ouyang36@uw.edu

Xu Chen[2]
Mem. ASME
Mechatronics, Automation, and Control Systems Laboratory,
Department of Mechanical Engineering,
University of Washington,
Seattle, WA 98195
e-mail: chx@uw.edu

We present a system identification method based on recursive least-squares (RLS) and coprime collaborative sensing, which can recover system dynamics from non-uniform temporal data. Focusing on systems with fast input sampling and slow output sampling, we use a polynomial transformation to reparameterize the system model and create an auxiliary model that can be identified from the non-uniform data. We show the identifiability of the auxiliary model using a Diophantine equation approach. Numerical examples demonstrate successful system reconstruction and the ability to capture fast system response with limited temporal feedback. [DOI: 10.1115/1.4063481]

Keywords: recursive system identification, collaborative sensing, identifiability, least-squares method, model reparameterization, fast input slow output

1 Introduction

A common assumption for real-time control systems design is that the sampling of input and output signals is uniform, periodic, and synchronous [1]. In the information-rich world, however, data streams are often non-uniform and asynchronous. (In fact, real-time control system implementations often have to adjust the sampling rate to deal with irregular data [2].) While non-uniformly sampled data intuitively contain more temporal information for system analysis and controls [3,4], they violate the classical real-time control framework, and most existing methods for non-uniformly sampled systems are heuristic and specific [5]. It remains not well understood how to systematically leverage non-uniform data streams for real-time dynamic systems. In particular, as the first critical step in real-time controls, classic system identification requires synchronous input and output data when building the model of a dynamical system [6].

From a signal processing point of view, non-uniform data are naturally dense in certain temporal regions where more information about the system dynamics can be revealed [4]. The non-uniformly sampled data can be collected by triggering the sensor with events, by randomized sampling, or by fusing measurements from multiple sensors. On the one hand, the temporal resolution is increased due to the data irregularity [3]. On the other hand, it challenges conventional system identification algorithms.

One approach to identifying a system under non-uniform data is based on the approximation theory [7]. Briefly, the non-uniformly collected data are approximated or reconstructed by a sequence of uniform samples, and then, the conventional system identification algorithms can be applied to the resulting uniform data [8]. Several techniques have been proposed for the data reconstruction, including linear [9], polynomial [10], and spline interpolations [11]. Other works on system identification subject to non-uniformly sampled data have also been conducted using the expectation maximization approach [12–14], the maximum likelihood estimation [14,15], the lifting operator [16,17], and the output error method [18].

Stepping further beyond the existing approaches, this paper contributes to a novel system identification that leverages the temporal advantage of non-uniform sampling but overcomes the obstacle imposed by non-uniform data collection for general input–output models. We first propose a coprime collaborative sensing scheme, which generates one set of data that appears non-uniform with respect to time while, in the meantime, having systematic underlying sampling patterns. Next, we implement a model reparameterization tailored for the selected sensing scheme based on polynomial transformation to construct an auxiliary model that can be directly identified with the available observations. Then, a recursive least-squares (RLS)-based algorithm is designed to identify the auxiliary model and to illustrate the feasibility of working with the mechanism of collaborative sampling and model reparameterization. Lastly, the parameters of the original fast system model are recovered by removing the highest common factors between the denominator and numerator polynomials.

The remainder of this paper is organized as follows. In Sec. 2, technical preliminaries regarding the model reparameterization are reviewed and introduced. The proposed coprime collaborative sensing and system modeling are formally defined in Sec. 3. In Sec. 4, we derive recursive system identification algorithms based on the proposed sensing scheme and model reparameterization strategies. Section 5 contains multiple classes of numerical examples. Section 6 concludes the paper.

[1]Paper presented at the 2023 Modeling, Estimation, and Control Conference (MECC 2023), Lake Tahoe, NV, Oct. 2–5, Paper No. MECC2023-88.
[2]Corresponding author.

Manuscript received July 24, 2023; final manuscript received August 27, 2023; published online October 25, 2023. Assoc. Editor: Murat Inalpolat.

2 Preliminaries

In this section, we review the transfer operator and model parameterization in standard single-rate and multi-rate system identification. Consider the deterministic autoregressive-moving-average model for a linear time-invariant system:

$$y(k) = \frac{q^{-d}B(q^{-1})}{A(q^{-1})}u(k) \tag{1}$$

where d is an integer number of sampling periods contained in the time delay of the systems, and q is the time-domain shift operator defined as $qy(k) = y(k+1)$ and $q^{-1}y(k) = y(k-1)$, and $A(q^{-1}) = 1 + a_1q^{-1} + \cdots + a_{n_a}q^{-n_a}$ and $B(q^{-1}) = b_1q^{-1} + b_2q^{-2} + \cdots + b_{n_b}q^{-n_b}$ are polynomials of q^{-1}. Equation (1) can be rewritten as

$$A(q^{-1})y(k) = q^{-d}B(q^{-1})u(k) \tag{2}$$

or alternatively $(1 + \cdots + a_{n_a}q^{-n_a})y(k) = (b_1q^{-1} + \cdots + b_{n_b}q^{-n_b})u(k-d)$. Expanding and rearranging yield

$$\begin{aligned} y(k) = &-a_1y(k-1) - \cdots - a_{n_a}y(k-n_a) \\ &+ b_1u(k-d-1) + \cdots + b_{n_b}u(k-d-n_b) \end{aligned} \tag{3}$$

or $y(k) = \theta^T\phi(k)$, where

$$\begin{aligned} \phi(k) = [&-y(k-1), -y(k-2), \ldots, -y(k-n_a), u(k-d-1), \\ &u(k-d-2), \ldots, u(k-d-n_b)]^T \\ \theta = [&a_1, a_2, \ldots, a_{n_a}, b_1, b_2, \ldots, b_{n_b}]^T \end{aligned}$$

When the system input and output are sampled at different rates (slower output sampling in this paper), the available data become $y(Jk) = \{y(k-J),\ y(k-2J),\ \ldots\}$ and $u(k) = \{u(k-1),\ u(k-2),\ \ldots\}$, where J is a positive integer representing the ratio between the input and output sampling rates. The original single-rate model described in Eq. (1) can be transformed into a dual-rate version that can be identified directly from the available measurements [19]. The solution approach is first to recognize the factorization:

$$1 - x^J = (1-x)(1 + x + x^2 + \cdots + x^{J-1}) \tag{4}$$

Next, consider the characteristic equation $A(q^{-1})$ in the multiplication form:

$$A(q^{-1}) \triangleq \prod_{i=1}^{n_a}[1 - (\lambda_i q)^{-1}] \tag{5}$$

where λ_i's are the reciprocals of the poles of the system, and n_a is the order of the characteristic equation (i.e., the number of poles). Observing the structure of Eq. (4), we notice that by designing a polynomial:

$$\begin{aligned} F_J(q^{-1}) &= \prod_{i=1}^{n_a}\left[1 + (\lambda_i q)^{-1} + (\lambda_i q)^{-2} + \cdots + (\lambda_i q)^{-J+1}\right] \\ &= 1 + f_1q^{-1} + \cdots + f_{n_aJ-n_a}q^{-n_aJ+n_a} \end{aligned} \tag{6}$$

the original characteristic equation described in Eq. (5) can be transferred into

$$\begin{aligned} A_J(q^{-J}) &= A(q^{-1})F_J(q^{-1}) \\ &= [1 - (\lambda_1 q)^{-J}][1 - (\lambda_2 q)^{-J}]\cdots[1 - (\lambda_{n_a} q)^{-J}] \\ &= 1 + a_{J,1}q^{-J} + a_{J,2}q^{-2J} + \cdots + a_{J,n_a}q^{-n_aJ} \end{aligned}$$

with a down-sampled observation space. Applying the same transformation polynomial shown in Eq. (6) to the numerator of Eq. (1) yields a multi-rate system model:

$$\begin{aligned} y(k) &= \frac{q^{-d}B(q^{-1})F_J(q^{-1})}{A(q^{-1})F_J(q^{-1})}u(k) \\ &= \frac{q^{-d}B_J(q^{-1})}{A_J(q^{-J})}u(k) \end{aligned} \tag{7}$$

or in a form similar to Eq. (2):

$$A_J(q^{-J})y(k) = q^{-d}B_J(q^{-1})u(k) \tag{8}$$

where $B_J(q^{-1}) = b_{J,1}q^{-1} + \cdots + b_{J,n_a(J-1)+n_b}q^{-n_a(J-1)-n_b}$. Rewriting Eq. (8) in a predictor form similar to Eq. (3), we have

$$\begin{aligned} y(k) = &-a_{J,1}y(k-J) - \cdots - a_{J,n_a}y(k-n_aJ) \\ &+ b_{J,1}u(k-d-1) + \cdots + b_{J,n_b}u(k-d-n_a(J-1)-n_b) \end{aligned}$$

where the output prediction is precisely a linear combination of $u(k)$ and $y(Jk)$. Therefore, directly identifying the multi-rate model parameters becomes possible after the aforementioned model reparameterization.

The key insight of the introduced model reparameterization is to recognize that the historical measurements required for system identification depend solely on the order of system polynomials (i.e., $A(q^{-1})$ and $B(q^{-1})$). By designing a transformation polynomial, we can freely adjust the order of system polynomials. Consequently, the challenge posed by input and output asynchronism in identifying system dynamics is effectively circumvented.

3 Proposed Coprime Collaborative Sensing and Model Reparameterization

Figure 1 illustrates the proposed coprime collaborative sensing scheme, where multiple sensors with coprime sampling rates collaboratively sense the system output. Assuming the fundamental sampling period is T, and S represents the set of sensors sampling rate, we define the coprime sampling rate as $S = \{aT, bT, cT, \ldots\}$, where a, b, c, … are coprime integers. The data collected from these sensors are then combined chronologically, assuming that all sensors begin sampling simultaneously. The coprime sampling rates result in fewer measurements overlapping when multiple sensor measurements are fused, providing the highest temporal resolution as more details of the system response become available. This enables the parameter estimation to be updated with the maximum information entropy precisely when all sensor measurements overlap.

When n sensors of different rates are used, the available output measurements become

$$\begin{aligned} y(J_1k, \ldots, J_nk) = \{&y(k-J_1), y(k-2J_1), \ldots \\ &\vdots \\ &y(k-J_n), y(k-2J_n), \ldots\} \end{aligned}$$

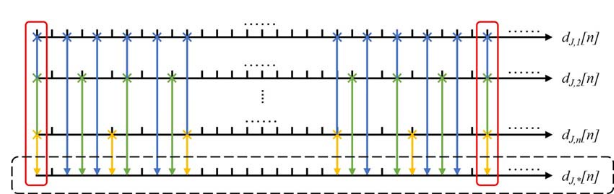

Fig. 1 The proposed collaborative sensing scheme of multiple sensors with coprime sampling rates. The illustration depicts the case when three coprime sensors' data are merged for use (boxed in dashed lines). The instants enclosed by solid lines represent valid measurements for updating parameter estimation (i.e., when all sensors' measurements overlap).

Based on the aforementioned multi-rate model reparameterization, we know that there will be n unique transformation polynomials

$$F_{J_1}(q^{-1}) = \prod_{i=1}^{n_a}\left[1 + (\lambda_i q)^{-1} + (\lambda_i q)^{-2} + \cdots + (\lambda_i q)^{-J_1+1}\right]$$

$$\vdots$$

$$F_{J_n}(q^{-1}) = \prod_{i=1}^{n_a}\left[1 + (\lambda_i q)^{-1} + (\lambda_i q)^{-2} + \cdots + (\lambda_i q)^{-J_n+1}\right]$$

if we want to identify the system model with the given output observation space. Let

$$F_{J_*}(q^{-1}) = \frac{1}{n}\sum_{i=1}^{n} F_{j_i}(q^{-1}) \tag{9}$$

Multiplying Eq. (9) to both sides of Eq. (2), we obtain the auxiliary system model:

$$A_{J_*}(q^{-J_1}, \ldots, q^{-J_n})y(k) = q^{-d}B_{J_*}(q^{-1})u(k) \tag{10}$$

where

$$A_{J_*}(q^{-J_1}, \ldots, q^{-J_n}) = 1 + \frac{a_{J_1,1}}{n}q^{-J_1} + \cdots + \frac{a_{J_1,n_a}}{n}q^{-n_aJ_1}$$
$$\vdots$$
$$+ \frac{a_{J_n,1}}{n}q^{-J_n} + \cdots + \frac{a_{J_n,n_a}}{n}q^{-n_aJ_n}$$

Therefore, the auxiliary model can be directly identified since the required measurements now match the available measurements after rewriting Eq. (10) in a predictor form. We will focus on the case where two output sensors of different rates are used in the remaining content for notational simplicity.

4 Recursive System Identification Under Collaborative Sensing

4.1 Recursive Least-Squares Formulation. We present the recursive least-squares formulation for the case where two coprime sensors are deployed collaboratively in the scheme above for the output measurement. First, we design transformation polynomials for the characteristic equation of the original system model as follows:

$$F_{J_1}(q^{-1}) = \prod_{i=1}^{n_a}(1 + (\lambda_i q)^{-1} + \cdots + (\lambda_i q)^{1-J_1})$$

$$F_{J_2}(q^{-1}) = \prod_{i=1}^{n_a}(1 + (\lambda_i q)^{-1} + \cdots + (\lambda_i q)^{1-J_2})$$

where J_1 and J_2 are coprime integers. Without loss of generality, we assume that a smaller index denotes the sensor with a faster sampling rate (i.e., $J_1 < J_2$). Summing up the two polynomials above and implementing the normalization in Eq. (9) yield

$$F_{J_*}(q^{-1}) = \frac{1}{2}[F_{J_1}(q^{-1}) + F_{J_2}(q^{-1})] \tag{11}$$

Next, multiplying the polynomial shown in Eq. (11) to both sides of the original system model in Eq. (2) yields $F_{J_*}(q^{-1})A(q^{-1})y(k) = q^{-d}F_{J_*}(q^{-1})B(q^{-1})u(k)$ or

$$A_{J_*}(q^{-J_1}, q^{-J_2})y(k) = q^{-d}B_{J_*}(q^{-1})u(k) \tag{12}$$

where

$$A_{J_*}(q^{-J_1}, q^{-J_2}) = 1 + \frac{a_{J_1,1}}{2}q^{-J_1} + \cdots + \frac{a_{J_1,n_a}}{2}q^{-n_aJ_1} + \frac{a_{J_2,1}}{2}q^{-J_2} + \cdots + \frac{a_{J_2,n_a}}{2}q^{-n_aJ_2}$$
$$B_{J_*}(q^{-1}) = b_{J_*,1}q^{-1} + b_{J_*,2}q^{-2} + \cdots + b_{J_*,n_a(J_2-1)+n_b}q^{-n_a(J_2-1)-n_b}$$

The order of $B_{J_*}(q^{-1})$ here comes from the sum of the order of $B(q^{-1})$, i.e., n_b, and that of $F_{J_*}(q^{-1})$, i.e., $n_a(J_2-1)$. For simplicity, let $\chi = n_a(J_2-1) + n_b$. Equation (12) can be rearranged as

$$y(k) = -\frac{a_{J_1,1}}{2}y(k-J_1) - \cdots - \frac{a_{J_1,n_a}}{2}y(k-n_aJ_1) - \frac{a_{J_2,1}}{2}y(k-J_2) - \cdots - \frac{a_{J_2,n_a}}{2}y(k-n_aJ_2) + b_{J_*,1}u(k-d-1) + \cdots + b_{J_*,\chi}u(k-d-\chi)$$

or in a vector form:

$$y(k) = \theta^T\phi(k) \tag{13}$$

where

$$\theta = \begin{bmatrix} a_{J_1,1} \\ a_{J_1,2} \\ \vdots \\ a_{J_1,n_a} \\ \hline a_{J_2,1} \\ a_{J_2,2} \\ \vdots \\ a_{J_2,n_a} \\ \hline b_{J_*,1} \\ b_{J_*,2} \\ \vdots \\ b_{J_*,\chi} \end{bmatrix}, \phi(k) = \begin{array}{l} \left.\begin{bmatrix} -\frac{1}{2}y(k-J_1) \\ -\frac{1}{2}y(k-2J_1) \\ \vdots \\ -\frac{1}{2}y(k-n_aJ_1) \end{bmatrix}\right\} \phi_{J_1} \\ \left.\begin{bmatrix} -\frac{1}{2}y(k-J_2) \\ -\frac{1}{2}y(k-2J_2) \\ \vdots \\ -\frac{1}{2}y(k-n_aJ_2) \end{bmatrix}\right\} \phi_{J_2} \\ \left.\begin{bmatrix} u(k-d-1) \\ u(k-d-2) \\ \vdots \\ u(k-d-\chi) \end{bmatrix}\right\} \phi_u \end{array} \tag{14}$$

From the vector form of the predictor function in Eq. (13), we see that the required historical output measurements are integer multiples of J_1 or J_2 steps behind the current instant k. At time instant iJ_1J_2, we have $\hat{y}(iJ_1J_2) = \phi^T(iJ_1J_2)\hat{\theta}(iJ_1J_2)$, $i = 0, 1, 2, \ldots$ Consider the performance index:

$$J_k = \sum_{i=1}^{k} e(i\varkappa)^2 = \sum_{i=1}^{k}\left[y(i\varkappa) - \phi^T(i\varkappa)\hat{\theta}(k\varkappa)\right]^2$$

where $\varkappa = J_1J_2$ for brevity of notation. The solution $\hat{\theta}(k\varkappa)$ can then be obtained using techniques from single-rate recursive least-squares, and the parameter adaptation algorithm (PAA) is as follows:

$$\hat{\theta}((k+1)\varkappa) = \hat{\theta}(k\varkappa) + F(k+1)\phi((k+1)\varkappa)\epsilon^o((k+1)\varkappa) \tag{15}$$

$$F(k+1) = F(k) - \frac{F(k)\phi((k+1)\varkappa)\phi^T((k+1)\varkappa)F(k)}{1 + \phi^T((k+1)\varkappa)F(k)\phi((k+1)\varkappa)} \tag{16}$$

where the a priori output estimation $\hat{y}^o$ and a priori output estimation error ϵ^o are defined as

$$\hat{y}^o((k+1)\varkappa) = \phi^T((k+1)\varkappa)\hat{\theta}(k\varkappa)$$
$$\epsilon^o((k+1)\varkappa) = y((k+1)\varkappa) - \hat{y}^o((k+1)\varkappa)$$

The stability of the PAA follows from standard hyperstability analysis for system identification and adaptive control [6]. Between $k\varkappa$ and $(k+1)\varkappa$, we keep the data asynchronous and hold the parameter estimate: $\hat{\theta}(k\varkappa + j) = \hat{\theta}(k\varkappa), j = 1, 2, \ldots, \varkappa - 1$.

4.2 PAA Convergence and Identifiability Analysis.

Parameter convergence in standard system identification requires the model to be irreducible, meaning that the polynomial orders cannot be further reduced and there are no common factors between $B(q^{-1})$ and $A(q^{-1})$.

When the input signal persistently excites the system dynamics, the convergence condition reduces to the existence of the solution to the following Diophantine equation associated with the polynomial parameters [6]

$$\tilde{A}(q^{-1})B(q^{-1}) - A(q^{-1})\tilde{B}(q^{-1}) = 0$$

where $\tilde{A}(q^{-1})$ and $\tilde{B}(q^{-1})$ represent the difference between the ground truth and the estimated system polynomials. In deterministic cases, parameters converge to the true values when the only solution to the Diophantine equation is $\tilde{A}(q^{-1}) = 0$ and $\tilde{B}(q^{-1}) = 0$. For the proposed collaborative sensing, it can be shown that parameter convergence still holds due to the following lemma.

Lemma 1 (Diophantine multiplicative equations). *Given a polynomial of the following form:*

$$F(z^{-1}) = f_0 + f_1 z^{-1} + f_2 z^{-2} + \cdots + f_m z^{-m}$$

where not all f_i's are zero, and

$$\alpha(z^{-1}) = 1 + \alpha_1 z^{-1} + \cdots + \alpha_n z^{-n}$$
$$\beta(z^{-1}) = \beta_1 z^{-1} + \beta_2 z^{-2} + \cdots + \beta_n z^{-n}$$

Then, the Diophantine equation

$$F(z^{-1})[\alpha(z^{-1}) \underbrace{\sigma(z^{-1})}_{\text{unknown}} + \beta(z^{-1}) \underbrace{\gamma(z^{-1})}_{\text{unknown}}] = 0 \tag{17}$$

has the unique zero solution for $\sigma(z^{-1})$ and $\gamma(z^{-1})$ (i.e., $\sigma(z^{-1}) = 0$ and $\gamma(z^{-1}) = 0$), if the numerators of $\alpha(z^{-1})$ and $\beta(z^{-1})$ are coprime, and the orders of σ and γ are restricted to be less than n as follows:

$$\sigma(z^{-1}) = \sigma_0 + \sigma_1 z^{-1} + \cdots + \sigma_{n-1} z^{-(n-1)}$$
$$\gamma(z^{-1}) = \gamma_0 + \gamma_1 z^{-1} + \cdots + \gamma_{n-1} z^{-(n-1)}$$

The proof is provided in the Appendix.

4.3 Parameter Recovery.

Recall the reparameterized system model:

$$y(k) = \frac{q^{-d}B(q^{-1})F_{J_*}(q^{-1})}{A(q^{-1})F_{J_*}(q^{-1})} u(k) = \frac{q^{-d}B_{J_*}(q^{-1})}{A_{J_*}(q^{-J_1}, q^{-J_2})} u(k)$$

By applying the aforementioned RLS-system identification algorithm, the intermediate parameter vector, i.e., the coefficients of $B_{J_*}(q^{-1})$ and $A_{J_*}(q^{-J_1}, q^{-J_2})$, can be identified directly. By removing the highest-order common factor from $B_{J_*}(q^{-1})$ and $A_{J_*}(q^{-J_1}, q^{-J_2})$, the original fast model polynomials $B(q^{-1})$ and $A(q^{-1})$ can then be obtained.

5 Case Study

We present three cases with different system setups, including a practical example in motion controls. We assume that two sensors are deployed for the output data collection. For the first two simulation cases, J_1 and J_2 are 2 and 3 times slower than the input sampling rate, respectively, and an input pseudo-random binary sequence (PRBS) signal is generated at 1024 Hz. A sufficiently long time horizon is selected to ensure parameter convergence (within ten iterations). For the third motion control example implemented on a hard drive drive (HDD) benchmark, we assume that J_1 and J_2 are 9 and 13 times slower. The PRBS signal is generated at 50,400 Hz. Algorithm 1 outlines the implementation steps for the proposed algorithm.

Algorithm 1 Collaborative sensing RLS system identification

```
Input: u(k), y(k), F, J1, J2, na, nb, d
θ, φ ← na, nb, d while t ≤ t_operation do
  if t mod J1 · J2 = 0 then
    Update θ, F;                    // refer to Eqs. (15) and (16)
    Update φ;                       // refer to Eq. (14)
  else
    if t mod J1 = 0 then
      Update φ_J1, φ_u              // refer to Eq. (14)
    end
    if t mod J2 = 0 then
      Update φ_J2, φ_u              // refer to Eq. (14)
    end
  end
end
B_J*(q^-1), A_J*(q^-J1, q^-J2) ← θ
B(q^-1), A(q^-1) ← B_J*(q^-1), A_J*(q^-J1, q^-J2)
Return: B(q^-1), A(q^-1)
```

5.1 Third-Order System.

Consider

$$y(k) = \frac{q^{-2} + 0.5q^{-3}}{1 + 0.9q^{-1} + 0.26q^{-2} + 0.024q^{-3}} u(k)$$

where the poles are at −0.2, −0.3, and −0.4, and the zero is at −0.5. Rewrite the transfer function into the general form for system identification as follows:

$$G(q^{-1}) \triangleq \frac{q^{-d}B(q^{-1})}{A(q^{-1})} = \frac{q^{-1}(q^{-1} + 0.5q^{-2})}{1 + 0.9q^{-1} + 0.26q^{-2} + 0.024q^{-3}}$$

From the general form, we record the hyperparameters for the algorithm, which are $d = 1$, $n_a = 3$, and $n_b = 2$. The model parameters needed to be identified are $\tilde{B}(q^{-1})$: [1.0, 0.5], $\tilde{A}(q^{-1})$: [1.0, 0.9, 0.26, 0.024]. The identified system response is shown in Fig. 2 and the parameters convergence of the auxiliary model is shown in Fig. 3. We also plotted the Nyquist frequencies of the individual sensors and observed the accurate model identification beyond the limitations of the individual sensors.

5.2 Higher-Order System.

Consider a fourth-order system

$$G(q^{-1}) = \frac{q^{-1}(q^{-1} + 1.5q^{-2} + 0.56q^{-3})}{1 + 1.4q^{-1} + 0.71q^{-2} + 0.154q^{-3} + 0.012q^{-4}}$$

where the poles are at −0.2, −0.3, −0.4, and −0.5, and the zeros are at −0.6 and −0.7. The hyperparameters are $d = 1$, $n_a = 4$, and $n_b = 3$. The identified parameters are $\tilde{B}(q^{-1})$: [0.9999, 1.4999, 0.5602], $\tilde{A}(q^{-1})$: [1.0000, 1.3999, 0.7102, 0.1541, 0.0120].

Figure 4 compares the original and identified system responses.

5.3 Hard Drive Drive Benchmark System.

Consider the major first two modes in the voice coil motor of an HDD benchmark

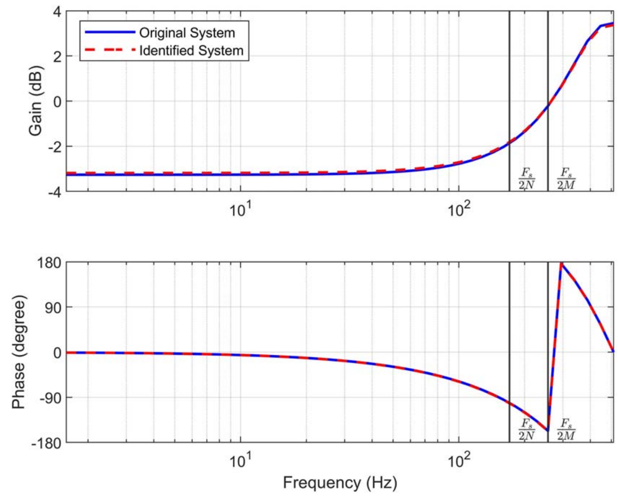

Fig. 2 The frequency response comparison of the third-order system, and the identification beyond the Nyquist frequency

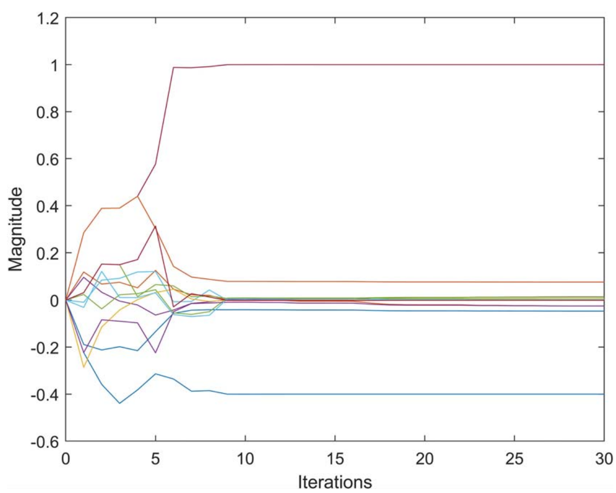

Fig. 3 Illustration of parameter convergence of the auxiliary model for 30 iterations

system [20]:

$$G(q^{-1}) = \frac{0.00033q^{-1} + 0.003q^{-2} + 0.00264q^{-3} + 0.00019q^{-4}}{1 - 3.559q^{-1} + 5.091q^{-2} - 3.506q^{-3} + 0.9739q^{-4}}$$

where the poles are at $0.7792 + 0.6055i$, $0.7792 - 0.6055i$, 1, and 0.9999, and the zeros are at -8.2069, -0.0831, and -0.8835. The plant has common characteristics that relate torque/force to position in motion control. The hyperparameters are $d = 0$, $n_a = 4$, $n_b = 4$, $M = 9$, and $N = 13$. The identified parameters are $\tilde{B}(q^{-1})$: [0.00033, 0.00303, 0.00265, 0.00020], $\tilde{A}(q^{-1})$: [1, -3.55851, 5.09094, -3.50634, 0.97391]. Figure 5 compares the original and identified system responses.

In all cases, the proposed algorithm was observed to have accurately identified the underlying system dynamics beyond the individual sensor's Nyquist sampling limit.

6 Conclusion and Future Work

This paper presented a novel framework for non-uniformly sampled system identification based on the proposed coprime collaborative sensing and the RLS-based algorithm. Leveraging a polynomial transformation and characteristics of coprime numbers, we

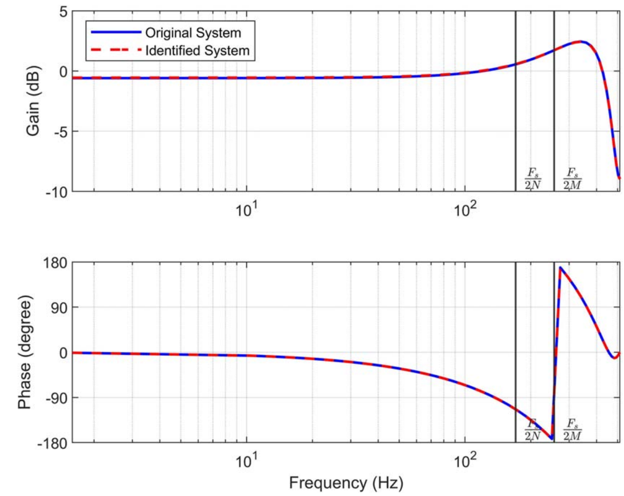

Fig. 4 Higher-order system frequency response comparison, and the identification beyond the Nyquist frequency

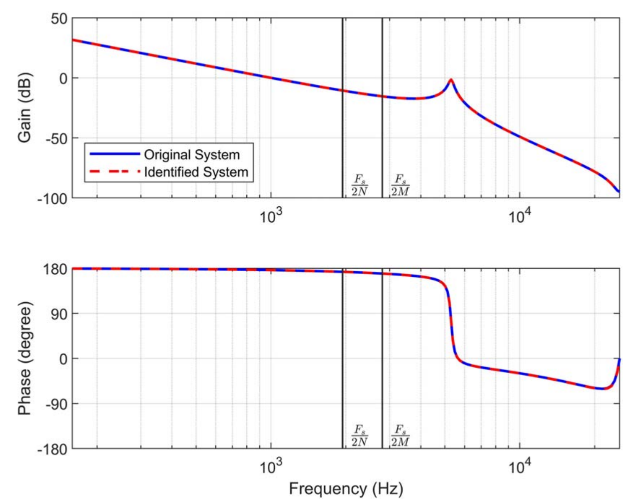

Fig. 5 HDD benchmark system frequency response comparison, and the identification beyond the Nyquist criterion

showed how the algorithm can recover fast system models beyond the Nyquist frequencies of multiple slow sensors. Example applications in motion control illustrate the effectiveness of the process. Future work includes optimal sensor rate selection, minimum data requirements, and addressing noise in stochastic environments.

Acknowledgment

This material is based upon work supported by the National Science Foundation under Grants Nos. CMMI-1953155 and CMMI-2141293. The opinions, findings, and conclusions, or recommendations expressed are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

Conflict of Interest

There are no conflicts of interest.

Data Availability Statement

The datasets generated and supporting the findings of this article are obtainable from the corresponding author upon reasonable request.

Appendix: Diophantine Multiplicative Equations Proof

Proof. Given

$$F(z^{-1}) = f_0 + f_1 z^{-1} + f_2 z^{-2} + \cdots + f_m z^{-m}$$

let

$$\tilde{\eta}(z^{-1}) = \alpha(z^{-1})\sigma(z^{-1}) + \beta(z^{-1})\gamma(z^{-1}) \tag{A1}$$

where

$$\alpha(z^{-1}) = 1 + \alpha_1 z^{-1} + \cdots + \alpha_n z^{-n}$$
$$\beta(z^{-1}) = \beta_1 z^{-1} + \beta_2 z^{-2} + \cdots + \beta_n z^{-n}$$

and

$$\sigma(z^{-1}) = \sigma_0 + \sigma_1 z^{-1} + \cdots + \sigma_{n-1} z^{-(n-1)}$$
$$\gamma(z^{-1}) = \gamma_0 + \gamma_1 z^{-1} + \cdots + \gamma_{n-1} z^{-(n-1)}$$

are unknown a priori.

We show that $F(z^{-1})\tilde{\eta}(z^{-1}) = 0$ holds only when $\tilde{\eta}(z^{-1}) = 0$ and subsequently $\sigma(z^{-1})$ and $\gamma(z^{-1})$ must all be zero. After forming the Sylvester matrix, Eq. (A1) is equivalent to the linear equality:

$$S \begin{bmatrix} 1 \\ \alpha_1 \\ \vdots \\ \alpha_{n-1} \\ \gamma_0 \\ \gamma_1 \\ \vdots \\ \gamma_{n-1} \end{bmatrix} = \begin{bmatrix} \tilde{\eta}_0 \\ \tilde{\eta}_1 \\ \vdots \\ \tilde{\eta}_{n-1} \\ \tilde{\eta}_n \\ \tilde{\eta}_{n+1} \\ \vdots \\ \tilde{\eta}_{2n-1} \end{bmatrix}$$

where

$$S = \begin{bmatrix} 1 & 0 & \dots & 0 & 0 & \dots & \dots & 0 \\ \alpha_1 & \ddots & \ddots & \vdots & \beta_1 & \ddots & & \vdots \\ \vdots & \ddots & \ddots & 0 & \vdots & \ddots & \ddots & \vdots \\ \vdots & & \ddots & 1 & \vdots & & \ddots & 0 \\ \alpha_{n-1} & & & \alpha_1 & \beta_{n-1} & & & \beta_1 \\ \alpha_n & \ddots & & \vdots & \beta_n & \ddots & & \vdots \\ 0 & \ddots & \ddots & \vdots & 0 & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \alpha_{n-1} & \vdots & \ddots & \ddots & \beta_{n-1} \\ 0 & \dots & 0 & \alpha_n & 0 & \dots & 0 & \beta_n \end{bmatrix}_{2n,2n}$$

Then, the coefficients of the filter product

$$F(z^{-1})\tilde{\eta}(z^{-1}) = \eta_0 + \eta_1 z^{-1} + \cdots + \eta_{m+2n-1} z^{-m-2n+1}$$

satisfies

$$\underbrace{\begin{bmatrix} \eta_0 \\ \eta_1 \\ \vdots \\ \eta_{m+2n-1} \end{bmatrix}}_{\eta_*} = F_* \underbrace{\begin{bmatrix} \tilde{\eta}_0 \\ \tilde{\eta}_1 \\ \vdots \\ \tilde{\eta}_{2n-1} \end{bmatrix}}_{\tilde{\eta}_*} = F_* S \underbrace{\begin{bmatrix} \sigma_0 \\ \sigma_1 \\ \vdots \\ \sigma_{n-1} \\ \gamma_0 \\ \vdots \\ \gamma_{n-1} \end{bmatrix}}_{\xi_*}$$

where

$$F_* = \begin{bmatrix} f_0 & 0 & \dots & 0 \\ f_1 & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ \vdots & & \ddots & f_0 \\ f_{m-1} & & & f_1 \\ f_m & \ddots & & \vdots \\ 0 & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & f_{m-1} \\ 0 & \dots & 0 & f_m \end{bmatrix}_{m+2n-1,2n}$$

and all columns of F_* are linearly independent. Thus, if $\eta_* = 0$, $\tilde{\eta}_*$ must be a zero vector. If the numerators of $\alpha(z^{-1})$ and $\beta(z^{-1})$ are copirme, S will be nonsingular and thus $\tilde{\eta}_*$ and ξ_* form a one-to-one mapping. The unique solution to Eq. (10) is thus $\xi_* = 0$. ■

References

[1] Isermann, R., 1989, *Digital Control Systems*, Springer Science & Business Media, Heidelberg.

[2] Gambier, A., 2004, "Real-Time Control Systems: A Tutorial," 5th Asian Control Conference (IEEE Cat. No. 04EX904), Melbourne, Australia, July 20–23, Vol. 2, IEEE, pp. 1024–1031.

[3] Khan, S., Goodall, R. M., and Dixon, R., 2013, "Non-Uniform Sampling Strategies for Digital Control," Int. J. Syst. Sci., **44**(12), pp. 2234–2254.

[4] Ding, F., Qiu, L., and Chen, T., 2009, "Reconstruction of Continuous-Time Systems From Their Non-uniformly Sampled Discrete-Time Systems," Automatica, **45**(2), pp. 324–332.

[5] Albertos, P., and Crespo, A., 1999, "Real-Time Control of Non-uniformly Sampled Systems," Control Eng. Pract., **7**(4), pp. 445–458.

[6] Landau, I. D., Lozano, R., M'Saad, M., and Karimi, A., 2011, *Adaptive Control: Algorithms, Analysis and Applications*, Springer Science & Business Media, London.

[7] Powell, M. J. D., 1981, *Approximation Theory and Methods*, Cambridge University Press, Cambridge.

[8] Raghavan, H., Gopaluni, R. B., Shah, S., Pakpahan, J., Patwardhan, R., and Robson, C., 2005, "Gray-Box Identification of Dynamic Models for the Bleaching Operation in a Pulp Mill," J. Process Control, **15**(4), pp. 451–468.

[9] Amirthalingam, R., Sung, S. W., and Lee, J. H., 2000, "Two-Step Procedure for Data-Based Modeling for Inferential Control Applications," AIChE J., **46**(10), pp. 1974–1988.

[10] Sircar, P., and Sarkar, T. K., 1988, "System Identification From Nonuniformly Spaced Signal Measurements," Signal Process., **14**(3), pp. 253–268.

[11] Gillberg, J., and Ljung, L., 2010, "Frequency Domain Identification of Continuous-Time Output Error Models, Part II: Non-uniformly Sampled Data and B-Spline Output Approximation," Automatica, **46**(1), pp. 11–18.

[12] Dempster, A. P., Laird, N. M., and Rubin, D. B., 1977, "Maximum Likelihood From Incomplete Data Via the EM Algorithm," J. R. Stat. Soc.: Ser. B (Methodol.), **39**(1), pp. 1–22.

[13] Shumway, R. H., and Stoffer, D. S., 1982, "An Approach to Time Series Smoothing and Forecasting Using the EM Algorithm," J. Time Series Anal., **3**(4), pp. 253–264.

[14] Isaksson, A. J., 1993, "Identification of ARX-Models Subject to Missing Data," IEEE Trans. Automat. Contr., **38**(5), pp. 813–819.

[15] Jones, R. H., 1980, "Maximum Likelihood Fitting of ARMA Models to Time Series With Missing Observations," Technometrics, **22**(3), pp. 389–395.

[16] Li, D., Shah, S. L., and Chen, T., 2001, "Identification of Fast-Rate Models From Multirate Data," Int. J. Control, **74**(7), pp. 680–689.

[17] Li, D., Shah, S. L., and Chen, T., 2002, "Analysis of Dual-Rate Inferential Control Systems," Automatica, **38**(6), pp. 1053–1059.

[18] Zhu, Y., Telkamp, H., Wang, J., and Fu, Q., 2009, "System Identification Using Slow and Irregular Output Samples," J. Process. Control, **19**(1), pp. 58–67.

[19] Lu, W., and Grant Fisher, D., 1988, "Output Estimation With Multi-Rate Sampling," Int. J. Control, **48**(1), pp. 149–160.

[20] Atsumi, T., 2023, "Magnetic-Head Positioning Control System in HDDs," https://www.mathworks.com/matlabcentral/fileexchange/111515-magnetic-head-positioning-control-system-in-hdds, MATLAB Central File Exchange, Retrieved July 20, 2023.