Data-Driven Decomposition Control to Output Tracking With Nonperiodic Tracking–Transition Switching Under Input Constraint

Jiangbo Liu[1]
Mathworks Inc.,
3 Apple Hill Drive,
Natick, MA 01760
e-mail: robin2010.liu@gmail.com

Qingze Zou[2]
Fellow ASME
Mechanical and Aerospace Engineering Department,
Rutgers University,
98 Brett Road,
Piscataway, NJ 08854
e-mail: qzzou@soe.rutgers.edu

This paper is concerned with solving, from the learning-based decomposition control viewpoint, the problem of output tracking with nonperiodic tracking–transition switching. Such a nontraditional tracking problem occurs in applications where sessions for tracking a given desired trajectory are alternated with those for transiting the output with given boundary conditions. It is challenging to achieve precision tracking while maintaining smooth tracking–transition switching, as postswitching oscillations can be induced due to the mismatch of the boundary states at the switching instants, and the tracking performance can be limited by the nonminimum-phase (NMP) zeros of the system and effected by factors such as input constraints and external disturbances. Although recently an approach by combining the system-inversion with optimization techniques has been proposed to tackle these challenges, modeling of the system dynamics and complicated online computation are needed, and the controller obtained can be sensitive to model uncertainties. In this work, a learning-based decomposition control technique is developed to overcome these limitations. A dictionary of input–output bases is constructed offline a priori via data-driven iterative learning first. The input–output bases are used online to decompose the desired output in the tracking sessions and design an optimal desired transition trajectory with minimal transition time under input-amplitude constraint. Finally, the control input is synthesized based on the superpositioning principle and further optimized online to account for system variations and external disturbance. The proposed approach is illustrated through a nanopositioning control experiment on a piezoelectric actuator. [DOI: 10.1115/1.4053763]

1 Introduction

Output tracking with nonperiodic tracking–transition switching appears in various applications, ranging from manufacturing (e.g., wielding operation) [1], robotic operation (e.g., noncontact-to-contact transition) [2], nanoscale measurement and manipulation to telesurgery [3]. In these applications, precision tracking of an operation-specified desired trajectory (e.g., follow the given part contour as in robotic wielding operation) is immediately followed and alternated by transition of the output between different positions in a nonperiodic manner (e.g., transit the welding head from one location to another as in robotic welding operation). Challenges arise as postswitching oscillations of the output can be induced by the tracking–transition switching, the tracking precision can be limited by the nonminimum-phase (NMP) dynamics of the system, and the operation efficiency in minimizing the total operation time can be confined by constraints such as the input amplitude. These challenges, however, have not yet been addressed efficiently and robustly and thereby motivate this work.

Limitations exist in current techniques developed for transition-involved output tracking. For example, the problem might be approached by treating and solving the output transition involved as an optimal state transition problem (e.g., Ref. [4]). The solution obtained, however, might be nonoptimal for output tracking and does not lend itself to track the desired trajectory after the transition [5]. Although output tracking after the transition might be addressed by using the input shaping technique [6] or the optimal output transition (OOT) technique [5], only can some simple and special cases be accommodated—either the after transition output needs to be maintained at a constant value [6] or the tracking–transition switching needs to be periodic [7]. Particularly, although the input energy is minimized in the OOT approach, the optimal transition trajectory obtained can be highly oscillatory when the system dynamics is lightly damped [7]. These issues have been tackled through the developments a multi-objective two-norm optimization scheme [8] or an inversion-based OOT technique [9]. In these techniques, however, constraints such as the input amplitude and optimization such as the minimization of the transition time have not been considered. Such a constrained optimization problem has recently been addressed in the stable-inversion framework through a multi-objective optimization approach and solved using an improved conjugate gradient technique [10]. It is shown that an optimal smooth output trajectory along with the corresponding control input can be obtained. Not only is smooth tracking–transition switching without postswitching oscillation ensured, but also the total transition time is minimized under the input (amplitude and energy) constraints [10]. Practical implementation of this technique, however, faces issues arising from the online computation being demanding and the control input obtained being sensitive to system dynamics variations. These critical issues related to efficiency and robustness require further investigation of this problem.

In this work, we propose a data-driven decomposition-based approach to achieve both time-minimal, input-constrained optimal design of the output transition trajectory and precision output tracking across tracking–transition switching. The basic idea is to decompose the desired output trajectory in both the tracking and the transition sessions into linear combination of output bases while taking into account the boundary conditions at the switching instants, then minimize the transition time under input constraint through a one-dimensional search algorithm. Then, the control

[1]J. Liu was with Zou's research group during this work.
[2]Corresponding author.

Contributed by the Dynamic Systems Division of ASME for publication in the Journal of Dynamic Systems, Measurement, and Control. Manuscript received November 13, 2020; final manuscript received January 25, 2022; published online March 7, 2022. Assoc. Editor: Sandipan Mishra.

input needed is synthesized using the corresponding input bases based on the *superpositioning principle*. The input–output bases are selected from a dictionary constructed a priori via a data-driven iterative learning control (ILC) technique [11]. Specifically, by utilizing the modeling-free ILC technique [11,12] along with B-spline as the output bases [13] in the dictionary construction, the proposed data-driven approach not only avoids the dynamics modeling process, but also is efficient in both the offline learning and the online output decomposition and input synthesis. Moreover, by using the input–output data to update both the dictionary offline (right before the operation) [12] and the control input online [14], both the robustness and the performance of the control are enhanced. This work extends the notion of the decomposition control [15–18] to input-constrained optimal tracking with nonperiodic tracking–transition switching. The proposed approach is illustrated through implementation in output tracking of a piezoelectric actuator in experiment. In this paper, the preliminary results reported in a recent conference [19] have been substantially enriched and strengthened with online adaptive optimization, extension to multiple-input multiple-output (MIMO) systems, and more complete experimental results.

2 Input-Constrained Optimal Nonperiodic Tracking–Transition Switching: Problem Formulation

We consider the input-constrained optimal (transition time minimal) nonperiodic tracking–transition switching (IC-ONTTS) problem for a square linear time invariant (LTI) system given by its state-space model as

$$\dot{x} = Ax + Bu, \quad y = Cx \tag{1}$$

where $x(\cdot) \in \Re^n$ is the state, and $u(\cdot) \in \Re^q$, $y(\cdot) \in \Re^q$ are the input and the output, respectively. We assume that

ASSUMPTION 1. *System (1) is stable, controllable, observable, and hyperbolic with a well-defined relative degree* $r = [r_1, r_2, \ldots, r_q]$ *([20]).*

The above hyperbolic assumption (i.e., system (1) has no zeros on the imaginary axis) is to guarantee the existence and uniqueness of the control input, i.e., for any given sufficiently smooth desired trajectory, there exists a *unique* control input to track the desired trajectory [21] This requirement can be alleviated by the right invertibility condition [22].

The NTTS occurs in preview-based output tracking involving nonperiodic tracking–transition switching, where the future desired output in a finite amount of preview time $T_p < \infty$ is partially specified in the tracking sessions, such that (see Fig. 1):

The number of tracking–transition switching, $\mathbf{N}_{tr}$, within the preview time T_p is known, i.e., there are $\mathbf{N}_{tr} - 1$ transition sessions.

The desired output trajectory in each tracking session $\mathbf{I}_{k,tr}$ is specified, i.e., $y_d(t)$ for $t \in \mathbf{I}_{k,tr} = (t_{k,i}, t_{k,f})$ is known for $\forall k = 1, 2, \ldots, \mathbf{N}_{tr}$, and sufficiently smooth, i.e., it is differentiable up to the relative degree of system (1).

The output boundary condition at each tracking–transition switching is specified: For any given kth transition session $\mathbf{I}_{k,tn} = [t_{k,i}, t_{k,f}]$ $(k = 1, 2, \ldots \mathbf{N}_{tk} - 1)$,

$$\begin{aligned} y^{(j)}_{k,dtn}(t_{k,i}) &= y^{(j)}_{k,dtr}(t_{k-1,f}) \\ y^{(j)}_{k,dtn}(t_{k,f}) &= y^{(j)}_{k,dtr}(t_{k,i}) \end{aligned} \tag{2}$$

for $j = 1, 2, \ldots, r - 1$, where $y_{k,dtr}(\cdot)$ and $y_{k,dtn}(\cdot)$ are the desired output in the kth tracking session and the to-be-designed output in the kth transition session, respectively.

The desired output being smooth in the tracking sessions is needed to ensure the existence of the control input for exact tracking of the desired output—by the stable-inversion theory [21]. Moreover, as the exact-tracking control input is completely determined by the desired output along with the system dynamics [21],

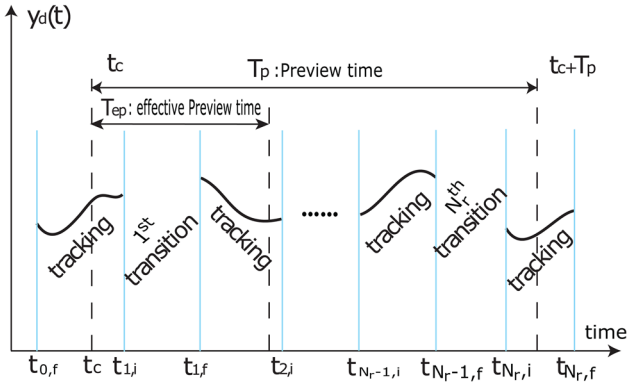

Fig. 1 Schematic representation of output tracking with nonperiodic tracking–transition switching

it is natural to assume that the desired output in each tracking session is trackable under the given input-amplitude constraint:

ASSUMPTION 2. *For a given* $\mathbf{M}_u < \infty$ *and any given kth tracking session* $(k = 1, 2, \ldots, \mathbf{N}_{tr})$*, the desired output* $y_{k,dtr}(t)$ *for* $\forall t \in \mathbf{I}_{k,tr}$ *satisfies the condition that the control input* $u_{k,dtr}(\cdot)$ *to exactly tracking* $y_{k,dtr}(\cdot)$ *has* $||u_{k,dtr}(\cdot)||_\infty \le \mathbf{M}_u$.

In practice, the desired trajectories in tracking sessions can be designed to satisfy the above Assumption 2 without knowing the system dynamics model (1) explicitly—based on the a priori knowledge of the system dynamics from, for example, past operation experience of the system.

For NMP systems, a large-enough preview time is necessary to ensure the tracking precision—by the stable-inversion theory [21]. We define the lower bound of such a preview time as the effective preview time T_{ep} (see Fig. 1):

Definition: Effective preview time T_{ep}: For any given $\varepsilon > 0$ and any given desired output trajectory $y_d(\cdot)$, the effective preview time T_{ep} is the amount of preview time such that with $u(\cdot) = u_p(\cdot)$, we have $||y(\cdot) - y_d(\cdot)||_2 \le \varepsilon$, where $u_p(\cdot)$ is the preview-based inverse control input obtained with T_{ep} [9].

Remark 3. For a given tracking precision (i.e., ε in the above definition), the amount of preview time T_p needed is determined by the zero dynamics of the system [21] and can be quantified by the smallest real part of the NMP zeros of the system. In practices, such a preview time can be estimated experimentally without knowing the dynamics model (1) explicitly. For example, the modeling-free inversion-based iterative control (MIIC) technique [11], a data-driven ILC technique, can be employed to track a given desired trajectory of a finite support (i.e., finite duration), and the preview time T_p can be chosen for different tracking precision by truncating the pre-actuation time of the converged control input [9].

Thus, we assume that enough preview time exists throughout the tracking course:

ASSUMPTION 4. *At any given time instant* t_c *(see Fig. 1), the available preview time* T_p *is at least twice of the effective preview time* T_{ep}*, i.e.,* $T_p \ge 2T_{ep}$.

The above assumption holds in practice as the amount of preview time can be adjusted, for example, by extending the transition time. Moreover, a long enough preview time also ensures that the input-constrained minimization of the transition time is well-defined as the inputs for the tracking sessions and the transition sessions, due to the pre-actuation and postactuation time, are coupled together.

In the presence of tracking–transition switching, the entire previewed desired trajectory can be partitioned into

$$\begin{aligned} y_d(\cdot) &= \left(\bigcup_{n=1}^{\mathbf{N}_{tr}} y_{n,dtr}(\cdot) \right) \cup \left(\bigcup_{m=1}^{\mathbf{N}_{tr}-1} y_{m,dtn}(\cdot) \right), m, n \in \mathbb{N} \\ &\triangleq y_{dtr}(\cdot) \cup y_{dtn}(\cdot) \end{aligned} \tag{3}$$

where $\cup$ denotes union of two sets. Correspondingly, the control input can be given as

$$u_{\text{Tot}}(\cdot) = \left(\bigcup_{n=1}^{\mathbf{N}_{tr}} u_{n,tr}(\cdot)\right) \cup \left(\bigcup_{m=1}^{\mathbf{N}_{tr}-1} u_{m,tn}(\cdot)\right) \triangleq u_{tr}(\cdot) \cup u_{tn}(\cdot) \tag{4}$$

where $u_{n,tr}(\cdot)$ and $u_{m,tn}(\cdot)$ are the control input for the nth tracking and the mth transition sessions, respectively.

Data-driven decomposition control (DD-DC) of IC-ONTTS: Let Assumptions 1, 2, and 4 hold for the LTI system (1), then the DD-DC of the IC-ONTTS problem is to use only the input–output data acquired a priori or in real-time without identifying and knowing the parameters of the dynamics model (1) to

- $\mathcal{O}_1$: Design the desired output trajectory in each of the transition sessions, $y_{m,dtn}(\cdot)$ for $m = 1, 2, \ldots, \mathbf{N}_{tr-1}$, such that the designed transition trajectory $y_{m,dtn}(\cdot)$ satisfies the required boundary condition in Eq. (2) for $m = 1, \ldots, \mathbf{N}_{tr-1}$.
- $\mathcal{O}_2$: Synthesize the control input $u_{\text{Tot}}(\cdot)$ as a linear combination of the input bases $\mathbf{u}_k(\cdot)$

$$u_{\text{Tot}}(t) = \sum_{k=1}^{N_p} g_k \mathbf{u}_k(t) \tag{5}$$

such that the required tracking precision is reached, i.e., with $u(t) = u_{\text{Tot}}(t)$, we have

$$||E_y(\cdot)||_2 \triangleq ||y(\cdot) - y_d(\cdot)||_2 \le \varepsilon, \quad \text{for} \quad \forall \varepsilon > 0 \tag{6}$$

for any given $\varepsilon > 0$ under Assumption 4.

- $\mathcal{O}_3$: With objectives $\mathcal{O}_1$ and $\mathcal{O}_2$ satisfied, minimize the total transition time

$$\min \sum_{m=1}^{\mathbf{N}_{tr}-1} \mathbf{I}_{m,tn}, \quad \text{subject to } ||u_{\text{Tot}}(\cdot)||_\infty \le \mathbf{M}_u \tag{7}$$

where $||u_{\text{Tot}}(\cdot)||_\infty \triangleq \sup_{t\in\Re} |u_d(t)|$, with $\mathbf{M}_u$ the given threshold of the input amplitude, respectively.

As the value of $\mathbf{M}_u$ is user-specified in practice, all the desired output transitions are reachable. Thus, the above minimization problem (7) is well-defined—the challenge is how to solve it. Unlike previous work [10] where the IC-ONTTS problem is approached in the stable-inversion framework, we seek to obtain, in this work, a decomposition control approach based on the superpositioning principle to achieve objectives $\mathcal{O}_1$ to $\mathcal{O}_3$ above. Such a superpositioning-based approach aims to not only avoid heavy online computation but also enhance the robustness against system dynamics variations.

3 Data-Driven Decomposition Control

We propose to extend the learning-based decomposition approach [16] to the IC-ONTTS problem. To ease the notation and simplify the presentation, we present the proposed approach below for single input single output (SISO) systems—extension to MIMO systems will be discussed later in Sec. 3.6.

3.1 Data-Driven Construction of Dictionary of Input–Output Bases. The dictionary is utilized online to both design the transition trajectory and synthesize the control input. Specifically, a dictionary consisting of pairs of input–output bases, $\mathbb{D}_B$, is constructed a priori

$$\mathbb{D}_B = \{(\mathbf{u}(\cdot, s_k), \mathbf{b}(\cdot, s_k)) : s_k \in \Re^+, k = 1, 2, \ldots, N_D\} \tag{8}$$

where $\mathbf{u}(\cdot, s_k)$ and $\mathbf{b}(\cdot, s_k)$ denote a pair of input–output basis at speed s_k, respectively, i.e., the output basis $\mathbf{b}(t, s_k)$ is obtained from an output basis at a prechosen speed $\hat{\mathbf{b}}(t)$s (called the *base-speed output basis*) via time scale

$$\mathbf{b}(t, s_k) = \hat{\mathbf{b}}(t/s_k), \ (s_k \in (0, 1] : \text{speed factor}) \tag{9}$$

and $\mathbf{u}(\cdot, s_k)$ is the control input to track $\mathbf{b}(\cdot, s_k)$, i.e., application of $\mathbf{u}(\cdot, s_k)$ to system (1) results in the output tracking error, $y_{\mathbf{b}}(\cdot, s_k)$, smaller than the given threshold value $\varepsilon_B > 0$

$$||y_{\mathbf{b}}(\cdot, s_k) - \mathbf{b}(\cdot, s_k)||_2 \le \varepsilon_B \tag{10}$$

We assume that

ASSUMPTION 5. *The base-speed output basis $\hat{\mathbf{b}}(\cdot)$ is positive, continuously differential up to the relative degree of system (1), compacted supported in $[-\tau_1, \tau_2]$ with $\tau_1, \tau_2 > 0$ much smaller than the domain of the trajectory to be tracked, and is maximum at $t = 0$, i.e., $\max_t \hat{\mathbf{b}}(t) = \hat{\mathbf{b}}(0)$.*

This assumption is general and can be satisfied by many types of bases employed in function approximation (e.g., uniform B-splines) [23]. Inclusion of output bases at different speeds s_ks in the dictionary $\mathbb{D}_B$ is to optimize the decomposition—select basis of speed that matches that of the trajectory to be decomposed will not only reduce the number of bases used but also improve the tracking precision.

Once the output bases satisfying Assumption 5 are chosen, construction of the dictionary $\mathbb{D}_B$ is equivalent to obtaining the corresponding input bases $\mathbf{u}(\cdot, s_k)$ to attain Eq. (10). We propose to utilize data-driven iterative learning control techniques [12,24] such as the MIIC technique [11] to obtain, offline a priori, the input basis $\mathbf{u}(\cdot, s_k)$ for tracking $\mathbf{b}(\cdot, s_k)$ accurately (see Eq. (29) in Sec. 4.2 later). Moreover, it can be shown [25] that only one type of output (input) bases—one output basis $\hat{\mathbf{b}}(\cdot)$ and its time-shifted copies—is needed in the decomposition (synthesis) process (see Ref. [25] for the details). Therefore, the offline learning process can be highly efficient to encompass bases at different speeds, i.e., at each speed we only need to learn the tracking of one output basis, and do so at different speeds s_ks to construct the dictionary $\mathbb{D}_B$.

The efficacy of the above offline data-driven dictionary construction stems from the tracking precision of the ILCs used. It can be shown [11] that in the presence of random output disturbance and noise, precision tracking of the output bases can be achieved by using the data-driven ILCs like the MIIC technique without having a parameterized system dynamics model (1). The efficacy of the data-driven ILCs has been experimentally demonstrated [11]—tracking error around noise level has been achieved in experiments for complicated trajectories such as a band-limited white noise [11].

3.2 $\mathcal{O}_1$: Transition Trajectory Design Via Decomposition. To design the transition trajectory in any given kth transition session ($k = 1, 2, \ldots, \mathbf{N}_{tr}$) in the previewed window, we first decompose the desired trajectory in the preceding kth and the following $(k+1)$th tracking session by using the output bases $\mathbf{b}(t, s_k)$ selected from the dictionary $\mathbb{D}_B$ (see Eq. (8)). This decomposition is needed such that the boundary conditions (Eq. (2)) are satisfied in the transition trajectory design, and the input synthesis (see Eq. (5)) can be obtained via the superpositioning principle.

The desired trajectory in the adjacent tracking sessions can be decomposed as

$$y_{i,dtr}(t) \approx \sum_{j=N_i-p}^{M_i+q} g_j \mathbf{b}(t - j\Psi_{s_k}, s_k) \triangleq \bar{y}_{i,dtr}(t) \tag{11}$$

for $i = k, k+1$, where, respectively, $\mathbf{b}(t, s_k) \in \mathbb{D}_{\mathbf{b}}$, Ψ_{s_k} is the corresponding speed s-dependent knot period

$$\Psi_{s_k} = s_k \Psi_0 \tag{12}$$

Ψ_0 is the base knot period of $\hat{\mathbf{b}}(t)$, and $p, q \in \mathbb{N}$ are integer constants determined by the support of the bases $b(\cdot, s_k)$, $[-\tau_{1,k}, \tau_{2,k}]$

$$\begin{aligned} p &= \tau_{2,k}/\Psi_{s_k} = \tau_2/\Psi_0 \\ q &= \tau_{1,k}/\Psi_{s_k} = \tau_1/\Psi_0 \end{aligned} \tag{13}$$

In Eq. (13), $\tau_{i,k} = s_k \tau_i$ $(i = 1, 2)$, and $\lfloor x \rfloor$ is the floor function (i.e., $\lfloor x \rfloor$ is the largest integer not larger than $x \in \Re$), respectively. As the desired trajectory in each kth tracking session, in general, has nonzero boundary values, the decomposed trajectory $\bar{y}_{k,dtr}(t)$ in Eq. (11) extrudes to the preceding $(k-1)$th and the next kth transition session, called the *head-* and the *tail-extrusion* (see Fig. 2), respectively. Thus, to match the boundary condition (2) in the transition trajectory design, these head- and tail-extrusions must be accounted for, and to further avoid the interference to the preceding and the next tracking session, we assume that

ASSUMPTION 6. *The transition session period* $\mathbf{I}_{k,tn}$ *is no less than* $(p+q-1)\Psi_{s_k}$, *i.e.*,

$$\min_k \mathbf{I}_{k,tn} \geq (p+q-1)\Psi_{s_k}, \quad k = 1, 2, \ldots, \mathbf{N}_{tr} - 1 \tag{14}$$

where p, q, *and* Ψ_{s_k} *are as given in Eqs. (13) and (12), respectively.*

In practice, this minimal transition time requirement is reasonable as both the knot period, Ψ_{s_k}, and the normalized support of the output basis, $[-p, q]$, are much smaller than the length of the tracking sessions $\mathbf{I}_{k,tr}$s.

Then, the transition trajectory in any given kth transition session $\mathbf{I}_{k,tn}$ $(k = 1, 2, \ldots, \mathbf{N}_{tr} - 1)$ will be designed by incorporating the head- and tail-extrusions as

$$y_{k,dtn}(t) = \bar{y}_{k,dtr}(t) + \bar{y}^{fp}_{k,dtn}(t) + \bar{y}_{k+1,dtr}(t), \; t \in \mathbf{I}_{k,tn} \tag{15}$$

where $\bar{y}_{k,dtr}(t)$ for $t \in [M_k + (q_k - p_k - 1), M_k + q_k]$ and $\bar{y}_{k+1,dtr}(t)$ for $t \in [N_{k+1} - p_k, N_{k+1} - (p_k - q_k - 1)]$ are the tail- and head-extrusion from the preceding kth and the following $(k+1)$th tracking session, respectively, as specified in Eq. (11), and $\bar{y}^{fp}_{k,dtn}(\cdot)$ is the "free-portion" of the transition trajectory obtained via decomposition as

$$\bar{y}^{fp}_{k,dtn}(t) = \sum_{j=M_k+p_k+1}^{N_{k+1}-q_k-1} g_j \mathbf{b}(t - j\Psi_s, s_k), \quad t \in \mathbf{I}_{k,tn} \tag{16}$$

where the coefficients g_js can be "freely" designed to meet other control purposes, for example, to meet objective $\mathcal{O}_3$ in this work. Smoothness of the trajectory designed in Eq. (15) is guaranteed as all the output bases are smooth, so is their summation [26]. Moreover, by including the bases across the tracking–transition switching in the transition trajectory design, truncation of the output bases to match the switching boundary condition [9] is avoided, and the boundary conditions are satisfied automatically. Moreover, the decomposition coefficients g_ks in Eq. (16) can be adjusted and even optimized to reduce and minimize the oscillation of the designed trajectory in the transition sessions.

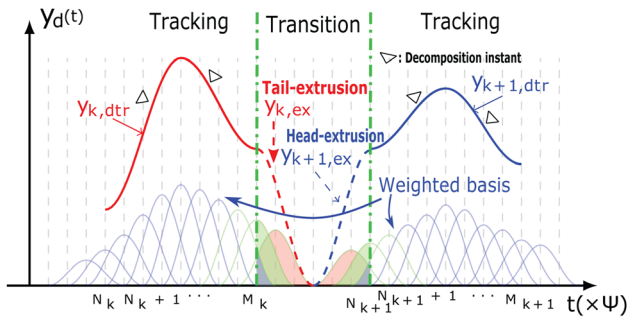

Fig. 2 Decomposition of the desired trajectory in the tracking sessions, resulting in tail- and head-extrusion into the transition session, where "Δ" denotes the decomposition instant (at which the trajectory decomposition occurs

3.3 $\mathcal{O}_2$: Input Synthesis Via Superpositioning. The decomposition representation of the desired trajectory in both the tracking and the transition sessions implies that the corresponding input can be obtained via the *superpositioning* principle of LTI systems [9]. Thus, to track the desired trajectory in any kth transition trajectory, $y_{k,dtn}(\cdot)$ in Eq. (16), the corresponding control input $u_{k,dtn}(\cdot)$ can be synthesized as

$$u_{k,dtn}(t) \;\; = u_{k,dtr}(t) + u^{fp}_{k,dtn}(t) + u_{k+1,dtr}(t) \tag{17}$$

where $u_{k,dtr}(\cdot)$, $u^{fp}_{k,dtn}(\cdot)$, and $u_{k+1,dtr}(\cdot)$ are the control input to track $\bar{y}_{k,dtr}(t)$, $\bar{y}^{fp}_{k,dtn}(t)$, and $\bar{y}_{k+1,dtr}(t)$ in Eq. (15), respectively, with the corresponding coefficients g_js given by those used in decomposing $\bar{y}_{k,dtr}(t)$, $\bar{y}^{m}_{k,dtn}(t)$, and $\bar{y}_{k+1,dtr}(t)$, respectively, and $\mathbf{u}(t, s_k)$s are the input bases corresponding to the output bases $\mathbf{b}(t, s_k)$s.

Similarly, the control input for each kth tracking session can be synthesized as

$$u_{k,dtr}(t) = \sum_{j=N_k-p_k}^{M_k+q_k} g_j \mathbf{u}(t - k\Psi_{s_k}, s_k) \tag{18}$$

where the coefficients $g_i \in \Re$s are as given in Eq. (11), respectively. Thus, the corresponding control input can be directly obtained via the superpositioning principle as

$$\begin{aligned} u_{\mathrm{Tot}}(t) &= \left(\bigcup_{k=1}^{\mathbf{N}_{tr}} u_{k,dtr}(t) \right) \cup \left(\bigcup_{k=1}^{\mathbf{N}_{tr}-1} u_{k,dtn}(t) \right) \\ &= \sum_{k=-p_k}^{N_D+q_k} g_k \mathbf{u}_k(t - k\Psi_{s_k}, s_k) \end{aligned} \tag{19}$$

where, respectively, g_ks are as those given in Eqs. (17) and (18), and N_D is the total number of knot periods over the entire previewed trajectory, respectively.

The performance of the above synthesized control $u_{\mathrm{Tot}}(\cdot)$ is guaranteed by the tracking precision of the output bases and that at any given time instant t, only few number of output bases are employed in the decomposition, i.e., at any time instant t, there are only a handful bases added together—a new output basis is added only at time instant $t = n\Psi_s$ for $n \in \mathbb{Z}$. For example, only three output bases are added at any time instant t when the third-order uniform B-splines are chosen as the output bases [16]. Thus, precision tracking can be largely inherited from that of the bases tracking. Readers are referred to Ref. [16] for the quantification of the tracking performance of the decomposition control approach.

In the proposed DD-DC approach above, the decomposition-synthesis operation only occurs at the so-called *decomposition instants* [16]—at each decomposition instant t_{dec}, the desired output and the corresponding input are decomposed and synthesized for time $t \in [t_{\mathrm{dec}}, t_{\mathrm{dec}} + T_p - 2T_{pa}]$, where T_{pa} is the pre-actuation time (see Fig. 2). Readers are referred to Ref. [16] for the detail.

Moreover, the tracking performance of the proposed DD-DC of OIC-NTTC is not limited by the NMP zeros of the system. For NMP systems, pre-actuation is needed for precision output tracking [21]—the tracking error can be rendered arbitrarily small by having the pre-actuation time (i.e., by applying the control input ahead of the start of the output early enough) no less than the effective preview time T_{ep} (see Remark 3). Thus, the transition trajectory design and the control input synthesis should be conducted ahead with a pre-actuation time of T_{ep}, and the control input applied becomes

$$\bar{u}_{\text{Tot}}(t) = u_{\text{Tot}}(t) + u_{\text{Tot,nt}}(t), \quad t \in [t_{k,\text{dec}},\, t_{k+1,\text{dec}} - T_{ep}) \tag{20}$$

where $t_{k,\text{dec}}$ denotes any kth decomposition instant, $u_{\text{Tot}}(t)$ is as in Eq. (19) (called below *the current decomposition input*), and $u_{\text{Tot,nt}}(t)$ is the pre-actuation control input (i.e., the synthesized input obtained from the design and decomposition at the next decomposition instant).

3.4 $\mathcal{O}_3$: Input-Constrained Transition Time Minimization. The linear combination representation of the desired transition trajectory (11) implies that the transition time is determined by the support of each output basis $\mathbf{b}(\cdot, s_k)$s and the total number of bases used in the transition trajectory design. Thus, we convert the input-constrained time-minimization in Eq. (7) to the minimization of the knot period Ψ_{s_k} and the total number of knot periods involved—under the input-amplitude constraint. More concretely, the optimal knot period $\Psi^*_{s_k}$ and the optimal number of knot periods $\mathbf{F}^*$ are sought to minimize the total transition time

$$\begin{aligned} &\min_{\Psi_{s_k}, F^*_k} \sum_{k=1}^{\mathbf{N}_{tr}} F_k \Psi_{s_k}, \quad \text{such that} \\ &||u_{\text{Tot}}(\cdot)||_\infty < M_u, \quad \text{with} \quad u_{\text{Tot}}(t) = \sum_{k=-p}^{N_D+q} g_k \mathbf{u}(t - k\Psi_{s_k}) \end{aligned} \tag{21}$$

where $F_k = N_{k+1} - M_k + 1$ is the number of knot periods in each kth transition session, with $\mathbf{F} = \sum_{k=1}^{\mathbf{N}_{tr}-1} F_k$. We propose to solve the above optimization problem sequentially: First, an optimal knot periods Ψ^* is sought offline a priori. Then second, during online tracking, the optimal knot period Ψ^* is selected and used to minimize the total number of knot periods $\mathbf{F}$ via searching.

As the length of the support for the combined output bases, in general, is proportional to that of each basis, and the tracking precision achieved in practice, is inverse proportional to the number of input bases combined at each time instant, the optimal knot period Ψ^* is sought to minimize, under the input saturation constraint, the length of the support of the output basis and thereby the length of both the head- and tail-extrusions and the transition session $\mathbf{I}_{k,tn}$. As the length of the output basis support is governed by the speed factor s_k, and the number of input bases added at each time instant is governed by its settling time (when the type of basis function has been chosen), we propose to select the optimal knot period Ψ^* by minimizing the following "speed indicator" $\mathbf{S}_\Psi$:

$$\begin{aligned} \min_{\Psi_{s_k}} \mathbf{S}_\Psi &= \min_{\Psi_{s_k}} \left(\lambda_1 s_k + \lambda_2 \frac{T_r}{q s_k \Phi_0} \right) \\ &= \min_{\Psi_{s_k}} \left(\lambda_1 \frac{\Psi_{s_k}}{\Psi_0} + \lambda_2 \frac{T_r}{q \Psi_{s_k}} \right) \end{aligned} \tag{22}$$

where $\lambda_1, \lambda_2 \in \Re^+$, with $\lambda_1 \in (0,\, 1)$, $\lambda_1 + \lambda_2 = 1$ are the weights, respectively, and T_r is the rising time of the input basis measured as the first time instant when the input basis reaches 2% of its maximal value, respectively. As $q\Psi_s$ quantifies the decaying time of the input to asymptotically approach to zero, $q\Psi_{s_k}/T_r$ provides a measure of the oscillation and thereby the settling time of the input basis Ψ_{s_k}. The weights λ_1 and λ_2 can be adjusted to account for the a priori knowledge of the tracking task (e.g., the estimated frequency spectrum of the desired trajectory in the tracking sessions, and the tracking precision required) in the above search of optimal knot period Ψ^*. For example, for tracking a relatively high speed trajectory with high-speed tracking–transition switching, a larger λ_1 over λ_2 can be chosen. Whereas a larger λ_2 over λ_1 shall be selected when a smooth input with shorting settling time is preferred.

After the optimal knot period Ψ^*, and thereby, the optimal input–output basis is determined, we propose to minimize the transition time session by session through a one-dimensional search. First, as the desired trajectory in each kth transition session consists of three portions, where the tail extrusion and the head extrusion are completely determined by the decomposition in the preceding and the following tracking session, respectively (see Eq. (11)), each of the kth transition time can be adjusted/reduced through the design of the free-portion, $\bar{y}^{fp}_{k,dtn}(\cdot)$. Particularly, under Assumption 2, the input portion for the tail extrusion and head extrusion, $u_{k,dtr}(t)$ and $u_{k+1,dtr}(t)$ in Eq. (17), respectively, are both below the saturation threshold. Thus, as the free-portion $\bar{y}^{fp}_{k,dtn}(\cdot)$ does not depend on the boundary condition, $\bar{y}^{fp}_{k,dtn}(\cdot)$ can be designed without altering the boundary conditions—to minimize the transition time, set $\bar{y}^{fp}_{k,dtn}(\cdot) = 0$ and the corresponding time interval $\mathbf{T}_{k,m} = 0$. Thus next, the minimization of each kth transition time amounts to minimizing the total number of knot periods of the preceding tail extrusion and the following head extrusion, by overlapping these two portions in time (see Fig. 2) under Assumption 6.

We propose an online one-dimension search algorithm such as the bisection search (see Algorithm 1 below) to reduce the total number of knot periods until the constraints are reached. Other search algorithm such as the Fibonacci search algorithm [23] can also be employed. As a modeling-free approach, this search algorithm also does not require identifying the model (1). The proposed approach can be easily extended to consider other constraint such as limited input energy (i.e., $||u_{\text{Tot}}(\cdot)||_2 < M_u$) or limited input-energy and amplitude combination (i.e., $\lambda_u ||u_{\text{Tot}}(\cdot)||_2 + (1 - \lambda_u)||u_{\text{Tot}}(\cdot)||_\infty < M_u$ with $\lambda_u \in (0,\, 1)$ a constant), by using the corresponding bound in the above one-dimension search. Moreover, the proposed approach can also be extended to optimize the input (e.g., minimal input energy) for a given transition time, through the design of the free-portion of the transition input $u^{fp}_{k,dtn}(\cdot)$ in Eq. (17).

Algorithm 1 Input-constrained time-minimal transition trajectory design via bisection search

1: Initialization: $k \leftarrow 1$,
2: **for** $k = 1 : N_{tr} - 1$ **do**
3: Decompose the kth and $k+1$th tracking sessions by using the selected output element $\mathbf{b}(t, s^*_k)$ corresponding to the optimal knot period Ψ^*; Obtain the decomposition coefficients g_ks for $k = N_k, N_k + 1, \ldots, M_{k+1}$;
Set the initial search index as the prechosen transition session period: $a \leftarrow 0$, $b \leftarrow \lfloor T_{k,tr}/\Psi^* \rfloor$;
4: **while** $a < b$ **do**
5: Obtain the total number of independent decomposition knot periods: $S_k \leftarrow \lfloor (a+b)/2 \rfloor$;
6: Obtain the starting index of the $k+1$th transition: $N_{k+1} \leftarrow M_k + S_k + p + q + 1$; ▷ Probably no need to mention
7: Obtain the corresponding input $u_{k,dtn}$ via Eq. (17) with coefficients g_ks for $k = N_k, N_k + 1, \ldots, M_{k+1}$;
8: **if** $\sup(u_{k,dtn}) > \mathbf{M}_u$ **then**
9: $a \leftarrow \lfloor (a+b)/2 \rfloor$
10: **else if** N is odd **then**
11: $b \leftarrow \lfloor (a+b)/2 \rfloor$
12: **end if**
13: **end while**
14: **end for**

3.5 Online Adaptation and Optimization. Finally, to further improve the robustness of the synthesized input against random disturbances (e.g., measurement noise, external disturbances, and rapid and/or nonlinear variations of the system dynamics), we propose an online moving-horizon input optimization. Similar to the idea explored in Ref. [16], we introduce a tuning factor γ_i into the synthesis of the control input, i.e., the decomposition coefficients g_is in Eq. (19) are now replaced as $g_i + \gamma_i$ instead—We consider the current decomposition input $u_{\text{Tot}}(t)$ as in Eqs. (19) and (20) at any given time instant t between any two given decomposition instants, $t \in (t_{k,\text{dec}},\, t_{k+1,\text{dec}})$

$$u_{\text{Tot}}(t) = \sum_{j=1}^{N_\Psi} (g_j + \gamma_j)\mathbf{u}_j(t - j\Psi_{s_k}, s_k) \tag{23}$$

where the total number of input bases N_Ψ depends on the output bases employed (e.g., when the third-order uniform B-splines are used as the output bases, $N_\Psi = 4$), and the following cost function:

$$L_\Gamma = \int_{t_{k,\text{dec}}}^{t} \left(y(\tau) - \sum_{j=1}^{N_\Psi} (g_j + \gamma_j)\mathbf{b}_j(\tau - j\Psi_{s_k}, s_k)\right)^2 d\tau \tag{24}$$

where $y(t)$ is the measured output, and

$$\Gamma = [\gamma_1\ \gamma_2, \ldots,\ \gamma_{N_\Psi}]^{\text{T}}$$

Then, the optimal Γ^* that minimizes L_Γ can be readily obtained via the least-mean square minimization method.

3.6 Extension to MIMO Systems. Finally, we briefly discuss the extension of the above DD-DC approach to IC-ONTTS to MIMO systems. Mainly the extension requires minor modification in the offline dictionary construction—the online decomposition and synthesis can be conducted separately for each output channel and thereby reduced to the SISO case.

For MIMO systems, the dictionary $\mathbb{D}_B$ now becomes

$$\begin{aligned} &\mathbb{D}_B = \{(\mathbf{u}_i(\cdot, s_k),\ \mathbf{b}(\cdot, s_k)) : i = 1, \ldots, q, k = 1, \ldots, N_D\} \\ &\text{with} \\ &\mathbf{b}_i(\cdot, s_k) = [0\ \cdots \mathbf{b}(\cdot, s_k)\ \cdots\ 0]^{\text{T}}, \quad \text{for} \quad i = 1, 2, \ldots, q \end{aligned} \tag{25}$$

where in each output basis $\mathbf{b}_i(\cdot, s_k)$, only one output basis $\mathbf{b}(\cdot, s_k)$ appears in the ith output channel, and $\mathbf{u}_i(\cdot, s_k)$ is the corresponding input basis—in general $\mathbf{u}_i(\cdot, s_k) \in \Re^{q\times 1}$ contains nonzero input in multiple input channels due to cross-axis coupling.

The above dictionary $\mathbb{D}_B$ Eq. (25) can also be constructed by using the data-driven ILCs like the MIIC technique above by taking the cross-coupling between input–output channels into account. For example, the cross-coupling effect can be accounted for by using the MIIC technique to learn the input–output pair iteratively: (I) Learn the input of the ith channel for each ith output base $\mathbf{b}_i(\cdot, s_k)$ in Eq. (25), and record the coupling-caused output in other channels, $y^c_{\mathbf{b},j}(\cdot, s_k)$ for $j = 1, \ldots, q$ and $j \neq i$. (II) Learn the input for tracking the opposite coupling-caused output in other channels $-y^c_{\mathbf{b},j}(\cdot, s_k)$ for $j = 1, \ldots, q$ and $j \neq i$ by using the MIIC technique, and record the combined coupling-caused output in the ith channel, $y^c_{\text{tot},i}(\cdot, s_k)$. (III) Learn the ith-channel input for the updated output bases $\mathbf{b}_i(\cdot, s_k) - y^c_{\text{tot},i}(\cdot, s_k)$. Steps (II) and (III) can be iterated if needed to improve the accuracy (in practice, only few iterations are needed).

Online implementation of the proposed approach for MIMO systems is almost the same as that for SISO systems. The decomposition-based transition trajectory design, the desired trajectory decomposition, and the input synthesis are conducted separately for each input–output channel, and the synthesized inputs (of each channel) are then applied simultaneously in parallel (i.e., by following the same time order). Compensation for the cross-coupling effect is ensured by the superpositioning principle: The cross-coupling-caused outputs are eliminated when applying the input bases $\mathbf{u}_i(\cdot, s_k) \in \Re^{q\times 1}$ (to each input channel simultaneously) to precisely track the corresponding output bases $\mathbf{b}_i(\cdot, s_k)$ in Eq. (25) and thereby are also eliminated when the synthesized input is applied—by the superpositioning principle. Also, this decoupled decomposition and synthesis scheme to MIMO systems also implies that each subsystem can have different input-amplitude constraint and the same approach can be applied. Therefore, with minor changes, the proposed DD-DC approach is extended to MIMO systems.

4 Experiment Implementation

4.1 Experiment Setup. We demonstrated the proposed approach for output transition/tracking by implementing it to a piezoelectrical actuator for lateral scanning (x–y axes direction) on an atomic force microscope (FastScan, Bruker Nano, Inc., Santa Barbara, CA). The output tracking with nonperiodic tracking–transition switching studied in this experiment mimicked that appearing in applications such as nanomanipulation and nanomanufacturing. All the control algorithms were coded and implemented through MATLAB (Mathworks Inc.) software environment along with a data acquisition system (PCI6259, National Instruments Inc.), at a sampling frequency of 20 kHz.

4.2 Implementation of the Proposed Data-Driven Decomposition Control Approach. Implementation of the proposed approach consisted of offline dictionary construction and optimal bases selection, and online desired tracking trajectory decomposition, transition trajectory design, and input synthesis.

Offline B-spline-based dictionary construction: Uniform B-splines were chosen as the output bases in the dictionary $\mathbb{D}_B$, where the output basis $\hat{\mathbf{b}}(\cdot)$ was given by the third-degree B-spline obtained recursively via

$$\begin{aligned} &B_{k,j}(t) = \frac{t - t_k}{t_{k+j} - t_k} B_{k,j-1}(t) + \frac{t_{k+j+1} - t}{t_{k+j+1} - t_{k+1}} \\ &B_{k+1,j-1}(t), \quad (j = 1, 2, 3) \end{aligned} \tag{26}$$

$$\text{where} \quad t_k = k\Psi_0, \text{ and initially} \tag{27}$$

$$B_{k,0} = \begin{cases} 1, & t_k \le t < t_{k+1} \\ 0, & \text{otherwise} \end{cases}$$

where the base-speed output bases are given as

$$\hat{B}(t) \triangleq B_{0,3}(t), \quad t \in [-2\Psi_0,\ 2\Psi_0] \tag{28}$$

Knot period of $\Psi_0 = 5.0$ ms for the base output bases $\hat{B}(\cdot)$ was chosen based on the desired output in the tracking sessions. Then, by setting $s = 2, 2.5, 5$, and 10 in Eq. (9), output bases at four different speeds were also obtained via Eq. (9) and used to construct the dictionary $\mathbb{D}_B$. These five speeds of input–output pairs were chosen as they covered the dynamic range of the desired output considered in this experiment and provided a good balance between the optimization of the decomposition (i.e., reducing the number of bases used) and the time and efforts needed in dictionary construction via ILCs. The corresponding input bases $\hat{U}(\cdot, s_k)$ ($k = 1, 2, 3$, and 4) were obtained by using the following MIIC technique [11] offline a priori:

$$\begin{aligned} \hat{U}_1(j\omega, s_k) &= k_{dc} B(j\omega, s_k) \\ \hat{U}_{n+1}(j\omega, s_k) &= \frac{\hat{U}_n(j\omega, s_k)}{y_n(j\omega, s_k)} B(j\omega, s_k) \end{aligned} \tag{29}$$

where $B(j\omega, s_k)$ is the Fourier transform of the output basis $B(\cdot, s_k)$, $\hat{U}_n(j\omega, s_k)$ and $y_n(j\omega, s_k)$ are the control input and the corresponding output obtained in the kth iteration, and k_{dc} is a prechosen constant (e.g., the inverse of the DC-gain of the piezo actuator, respectively). The convergence was reached in three to five iterations when tracking each of the output bases. The five pairs of input–output bases obtained are shown in Fig. 3.

Offline optimal bases selection: To minimize the total transition time while accounting for the input threshold limit, the optimal input–output pair to be used was identified first, as described in

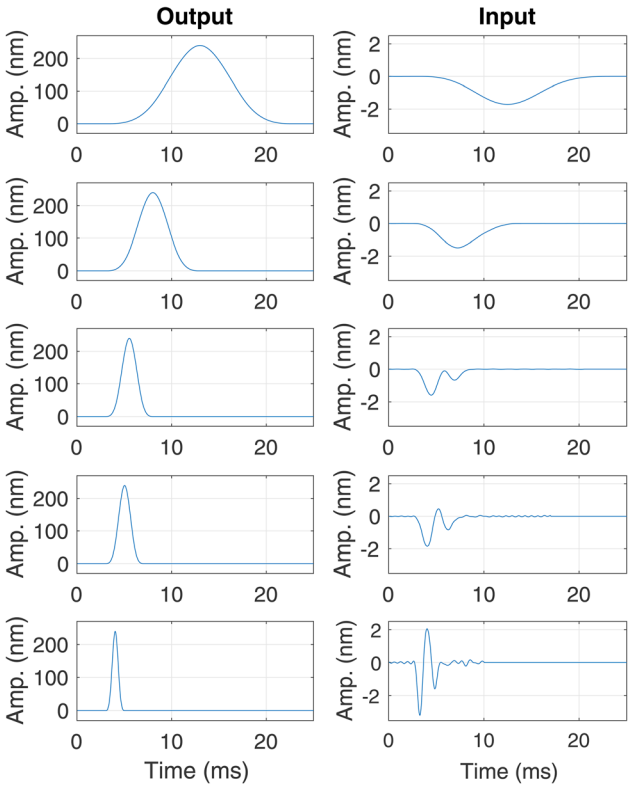

Fig. 3 Pairs of (left column) output elements and (right column) input elements at five different speeds used in the experiment, where the input elements were obtained experimentally via the MIIC technique [11] a priori

Table 1 The indicators of each element in the dictionary

Ψ_s (ms)	Ψ_s/Ψ_0	$T_{\text{set}}/(q\Psi_s)$	S_Ψ
5.0	1.0	1.74	1.37
2.5	0.5	1.09	0.80
2.0	0.25	1.48	0.87
1.0	0.2	2.83	1.52
0.5	0.1	7.6	3.9

Sec. 3.4. $\lambda_1=\lambda_2=0.5$ were selected in Eq. (22) to equally weight the "speed" of the bases and the corresponding input amplitude in the optimization. Then, the speed indicator S_Ψ was computed via Eq. (22). As shown in Table 1, among the five knot periods employed, the speed indicator S_Ψ was smallest at $\Psi_s=2.5$ ms. Thus, the optimal knot period of $\Psi^*=2.5$ ms was selected.

4.3 Transition Design and Trajectory Tracking Results. Next, the selected optimal input–output element (corresponding to the optimal knot period Ψ^*) was used in the online trajectory design and tracking with nonperiodic tracking–transition switching. The desired trajectory consisted of four tracking and three transition sessions in between to be designed, as shown in Fig. 4(*a*). With the support of the entire trajectory at ~200 ms, the preview time was set at 100 ms.

Two different speeds of tracking were tested and examined in the experiment, where two different speed factors at $s_k=0.25$ and $s_k=0.5$ were selected for the relatively slower and faster tracking (corresponding to the total tracking time of 300 ms and 200 ms), respectively. The decomposition of the tracking sessions and the design of the transition sessions at the speed factor $s_k=0.5$ are shown in Fig. 4, where the desired tracking sessions and the designed transition sessions are shown in Fig. 4(*a*), respectively,

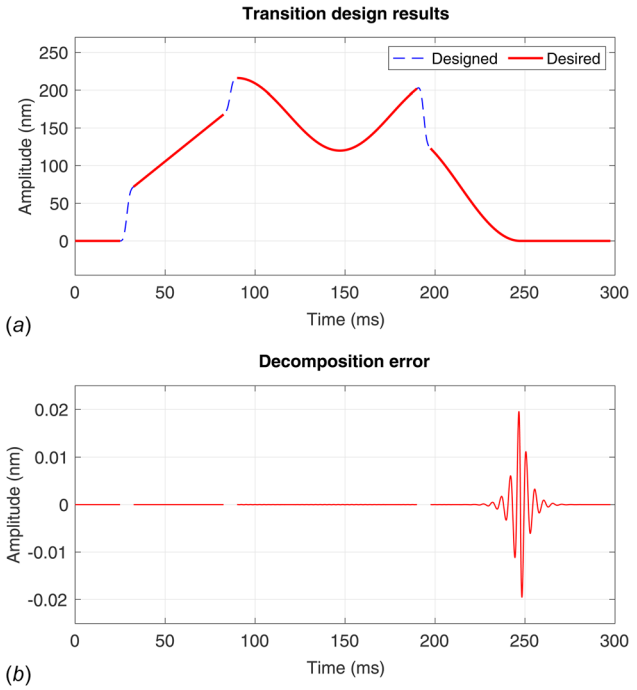

Fig. 4 (*a*) The decomposition of the tracking sessions and the design of the transition sessions and (*b*) the decomposition error in the tracking sessions

and the decomposition error is shown in Fig. 4(*b*). Then, the obtained desired trajectory was tracked by using the synthesized input. To compare the tracking performance, we also applied a conventional PI feedback controller to track the same designed trajectory, where the control parameters were well tuned experimentally at $K_p=0.42$ and $K_I=35$, respectively, resulting in a closed-loop bandwidth at ~120 Hz. The tracking results obtained are compared in Figs. 5(*a*) and 5(*b*) for $s_k=0.5$ and $s_k=0.25$, respectively.

The experimental results showed that the decomposition-caused error was extremely small (nearly 4 orders of magnitude smaller than the trajectory) and can be ignored (see Fig. 4(*b*)). The tracking results showed that by using the proposed approach, smooth transition trajectory design and tracking was achieved. The designed desired trajectory in the transition sessions was smooth, particularly cross the tracking–transition switching instants (Fig. 4(*a*)). The tracking results, as presented in Fig. 5, clearly showed that by using the proposed technique, precision tracking was maintained over the entire course. Whereas the tracking error was pronounced when using the well-tuned PI controller. At the relatively slow tracking at $s_k=0.5$, the relative RMS tracking error and the relative maximum error were at 0.90% and 3.5%, respectively, over 3.5 times and 2.5 times smaller than those obtained by the PI controller, respectively. Such a precision tracking was maintained as the speed increased (at $s_k=0.25$), with the relative RMS tracking error and the relative maximum error were at 1.60% and 5.9%, over 3.6 times and 2.6 times smaller than those obtained by the PI controller, respectively. Particularly, spikes-like transient tracking error was pronounced at the tracking–transition switching instants when using the PI controller, but became much subdued with the use of the proposed technique (see Fig. 6). As the same desired trajectory in the transition sessions was tracked when the PI controller was applied, the transient-caused spikes at the switching instants could have been larger otherwise.

Moreover, the experiment also demonstrated the efficiency and ease of use of the proposed technique in practical implementation: As a data-driven approach, the proposed technique removed the modeling process and instead exploited the high performance and

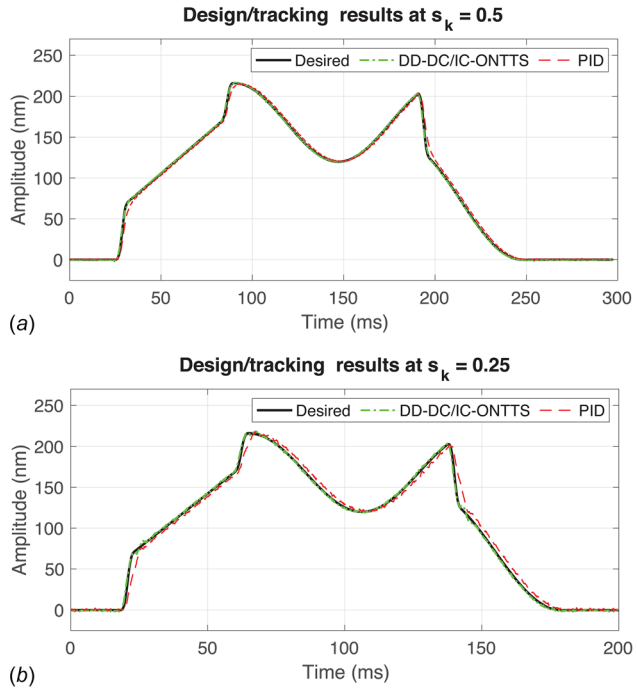

Fig. 5 Comparisons of the tracking result obtained in the experiment by using the proposed method (DSBOTT) with that by the PID control at (*a*) s_k=0.5 and (*b*) s_k=0.25, respectively

robustness against system dynamics changes of the data-driven ILC techniques for tracking. Through the proposed decomposition approach, these virtues of ILCs were extended to general nonrepetitive tracking and exploited for nonperiodic tracking–transition switching. The trajectory decomposition, transition trajectory design, and control input synthesis were only needed at a small number of instants—in this experiment, the decomposition instants (see Fig. 2) only accounted for less than 4% of the total sampling instants. Therefore, the experimental results clearly

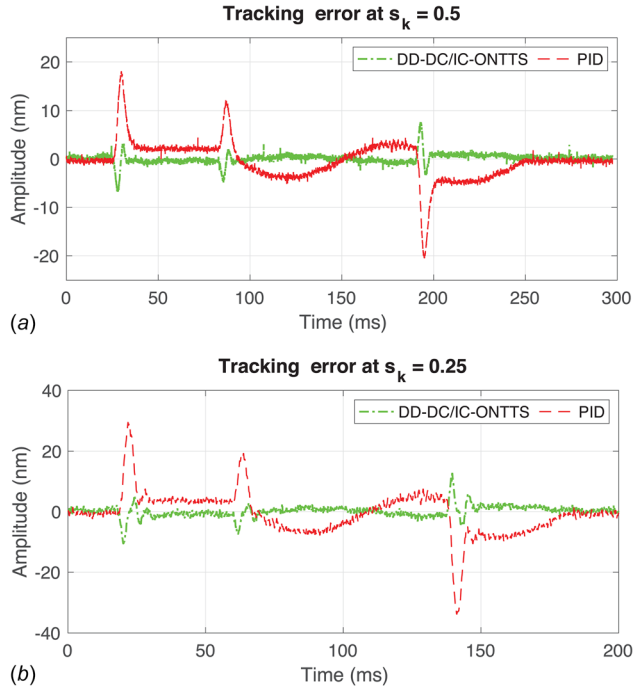

Fig. 6 Comparisons of the tracking error obtained in the experiment by using the proposed method (DSBOTT) with that by the PID control at (*a*) $s_k = xx$ and (*b*) $s_k = xx$, respectively

demonstrated the performance of the proposed method in output tracking with nonperiodic tracking–transition switching.

5 Conclusion

A data-driven decomposition control was proposed to optimal input-constrained output tracking with nonperiodic tracking–transition switching for linear systems. A dictionary of paired input–output bases was constructed a priori offline through data-driven iterative learning control technique. Then, the desired transition trajectory was designed as a linear combination of the output bases by taking the boundary condition into account, and the total transition time was minimized under the input-amplitude constraint through a one-dimensional searching algorithm. Finally, the corresponding control input was obtained as the linear combination of the corresponding input bases via the superpositioning principle, where the combination coefficients were online adjusted to compensate for the dynamics variations and external disturbance. The proposed method was implemented in experiment on a piezoelectric actuator to illustrate the proposed technique.

Funding Data

- NSF (Grant Nos. CMMI-1663055 and CMMI-1851097; Funder ID: 10.13039/100000001).

References

[1] Yue, W., Pano, V., Ouyang, P. R., and Hu, Y., 2017, "Model-Independent Position Domain Sliding Mode Control for Contour Tracking of Robotic Manipulator," Int. J. Syst. Sci., **48**(1), pp. 190–199.

[2] Salehian, S. S. M., and Billard, A., 2018, "A Dynamical-System-Based Approach for Controlling Robotic Manipulators During Noncontact/Contact Transitions," IEEE Rob. Autom. Lett., **3**(4), pp. 2738–2745.

[3] Sharifi, M., Behzadipour, S., and Salarieh, H., 2016, "Nonlinear Bilateral Adaptive Impedance Control With Applications in Telesurgery and Telerehabilitation," ASME J. Dyn. Syst., Meas., Control, **138**(11), p. 111010.

[4] Lewis, F. L., and Syrmos, V. L., 1995, *Optimal Control*, 2nd ed., Wiley, New York.

[5] Perez, H., and Devasia, S., 2003, "Optimal Output-Transitions for Linear System," Automatica, **39**(2), pp. 181–192.

[6] Lau, M. A., and Pao, L. Y., 2003, "Input Shaping and Time-Optimal Control of Flexible Structures," Automatica, **39**, pp. 893–900.

[7] Perez, H., Zou, Q., and Devasia, S., 2004, "Design and Control of Optimal Scan Trajectories: Scanning Tunneling Microscope Example," ASME J. Dyn. Syst., Meas., Control, **126**(1), pp. 187–197.

[8] Jetto, L., Orsini, V., and Romagnoli, R., 2015, "A Mixed Numerical–Analytical Stable Pseudo-Inversion Method Aimed at Attaining an Almost Exact Tracking," Int. J. Robust Nonlinear Control, **25**(6), pp. 809–823.

[9] Wang, H., Zou, Q., and Xu, H., 2012, "Inversion-Based Optimal Output Tracking–Transition Switching With Preview for Nonminimum-Phase Linear Systems," Automatica, **48**(7), pp. 1364–1371.

[10] Zou, Q., and Liu, J., 2016, "Multi-Objective Optimal Trajectory Design and Tracking With Non-Periodic Tracking-Transition Switching for Non-Minimum Phase Linear Systems," Int. J. Control, **89**(11), pp. 2371–2383.

[11] Kim, K., and Zou, Q., 2013, "A Modeling-Free Inversion-Based Iterative Feedforward Control for Precision Output Tracking of Linear Time-Invariant Systems," IEEE/ASME Trans. Mechatron., **18**(6), pp. 1767–1777.

[12] de Rozario, R., and Oomen, T., 2019, "Data-Driven Iterative Inversion-Based Control: Achieving Robustness Through Nonlinear Learning," Automatica, **107**, pp. 342–352.

[13] Jetto, L., Orsini, V., and Romagnoli, R., 2017, "B-Splines and Pseudo-Inversion as Tools for Handling Saturation Constraints in the Optimal Set-Point Regulation," 2017 American Control Conference (ACC), Seattle, WA, May 24-26 , IEEE, pp. 1041–1048.

[14] Tanaskovic, M., Fagiano, L., Novara, C., and Morari, M., 2017, "Data-Driven Control of Nonlinear Systems: An On-Line Direct Approach," Automatica, **75**, pp. 1–10.

[15] Hoelzle, D. J., Alleyne, A. G., and Johnson, A. J. W., 2009, "Iterative Learning Control Using a Basis Signal Library," 2009 American Control Conference, St. Louis, MI, June 10–12 , IEEE, pp. 925–930.

[16] Liu, J., and Zou, Q., 2018, "On Superposition of Hammerstein Systems: Application to Simultaneous Hysteresis-Dynamics Compensation," Int. J. Robust Nonlinear Control, **28**(14), pp. 4075–4092.

[17] Ramani, K. S., Duan, M., Okwudire, C. E., and Galip Ulsoy, A., 2017, "Tracking Control of Linear Time-Invariant Nonminimum Phase Systems Using Filtered Basis Functions," ASME J. Dyn. Syst., Meas., Control, **139**(1), p. 011001.

[18] Romagnoli, R., and Garone, E., 2019, "A General Framework for Approximated Model Stable Inversion," Automatica, **101**, pp. 182–189.
[19] Liu, J., and Zou, Q., 2018, "Non-Periodic Transition-Tracking Switching Via Learning-Based Decomposition: High-Speed Nano-Positioning Experiment Example," 2018 Annual American Control Conference (ACC), Milwaukee, WI, June 27–29, IEEE, pp. 6372–6377.
[20] Isidori, A., 1995, *Nonlinear Control Systems*, 3rd ed., Springer-Verlag, London.
[21] Devasia, S., Chen, D., and Paden, B., 1996, "Nonlinear Inversion-Based Output Tracking," IEEE Trans. Autom. Control, **41**(7), pp. 930–942.
[22] Hunt, L. R., and Meyer, G., 1997, "Stable Inversion for Nonlinear Systems," Automatica, **33**(8), pp. 1549–1554.
[23] Burden, R. L., and Faires, J. D., 2005, *Numerical Analysis*, 8th ed., Thomson Brooks/Cole.
[24] Minakais, M., Mishra, S., and Wen, J. T., 2019, "Database-Driven Iterative Learning for Building Temperature Control," IEEE Trans. Autom. Sci. Eng., **16**(4), pp. 1896–1906.
[25] Liu, J., and Zou, Q., 2016, "On Single-Basis Online Asymptotic Trajectory Decomposition for Control Applications," 2016 IEEE International Conference on Advanced Intelligent Mechatronics (AIM), Banff, AB, Canada, July 12–15, IEEE, pp. 1291–1296.
[26] Rudin, W., 1966, *Real and Complex Analysis*, 3rd ed., McGraw-Hill, New York.